

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Блинова Светлана Павловна  
Должность: Заместитель директора по учебно-воспитательной работе  
Дата подписания: 21.01.2025 07:54:25  
Уникальный программный ключ:  
1cafd4e102a27ce11a89a2a7ceb20237f3ab5c65

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Заполярье государственный университет им. Н.М. Федоровского»  
Политехнический колледж

## МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ПРОВЕДЕНИЮ ПРАКТИЧЕСКИХ РАБОТ ПО ДИСЦИПЛИНЕ

### «ИНФОРМАТИКА»

(1 курс)

для специальности:

40.02.04 Юриспруденция

Методические указания по проведению практических работ учебной дисциплины «Информатика» разработана на основе актуализированного Федерального государственного образовательного стандарта по специальности среднего профессионального образования

40.02.04 Юриспруденция

Организация-разработчик: Политехнический колледж ФГБОУ ВО «Заполярный государственный университет им. Н.М. Федоровского»

Разработчик:

Олейник М. В.

Рассмотрена на заседании предметной комиссии

Естественнонаучных дисциплин

Председатель комиссии




Олейник М.В.

Утверждена методическим советом ФГБОУ ВО «Заполярный государственный университет им. Н.М. Федоровского» Политехнический колледж

Протокол заседания методического совета № 3 от «22» 01 2025 г.

Зам. директора по УР



А.В. Петухова



## Слово к студентам

1 Что представляют собой методические указания?

Здесь вы найдете основной материал разделов: операционная система Windows, текстовый редактор Word, электронные таблицы Excel, разработка Web-сайтов, мультимедийные объекты Power Point и программированию.

В каждом разделе представлены образцы практических занятий, которые содержат: сведения из теории по конкретной теме, представленной именно в этом занятии; указан план занятия и непосредственно практическая часть. К каждому практическому занятию приведен список контрольных вопросов.

После всех практических занятий приведен пример обязательной контрольной работы.

В конце методических указаний приведены вопросы, ответы на которые необходимо знать для успешной сдачи зачета за II семестр первого года обучения в колледже.

Так же имеется достаточно полный список литературы.

2 Зачем нужны методические указания

Эта книжечка поможет Вам:

– найти нужную информацию по тому или иному разделу информатики I и II семестра первого года обучения в Политехническом колледже ЗГУ.

– повторить соответствующий материал при подготовке к обязательной контрольной работе, к зачету;

3 Как пользоваться методическими указаниями

Пособие состоит из 6 частей:

– пояснительная записка, где указано, какими знаниями и умениями Вы будете обладать при успешном изучении представленного материала;

– тематическое планирование, где указано, сколько часов отводится на изучение каждой темы;

– примеры практических занятий;

– примеры контрольной работы;

– литературы.

Я надеюсь, что данные методические указания станут Вам добрым помощником, и желаю Вам успехов!

М.В. Олейник

## 1. Пояснительная записка

Методические указания по дисциплине «Информатика» предназначена для студентов дневного отделения средних специальных учебных заведений технических и социально-экономических специальностей на базе девяти классов общеобразовательной школы.

Изучение основ работы с текстовым редактором WORD, электронными таблицами EXCEL, разработки Web-сайтов и объектов Power Point и программированием.

Предусмотрены практические занятия, которые носят репродуктивный, частично-поисковый и поисковый характер. По окончании каждого блока студенты выполняют контрольную работу на компьютере, которая включает все основные темы изученного материала.

Практические занятия рассчитаны на 70 ч.

**Использовать приобретенные знания и умения в практической деятельности и повседневной жизни:**

- эффективной организации индивидуального информационного пространства;
- автоматизации коммуникационной деятельности;
- эффективного применения информационных образовательных ресурсов в учебной деятельности.

**Общие и профессиональные компетенции для специальности:**

40.02.04 Юриспруденция

код	Общие компетенции
ОК 01	Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам.
ОК 02	Использовать современные средства поиска, анализа и интерпретации информации, и информационные технологии для выполнения задач профессиональной деятельности.
ОК 03	Планировать и реализовывать собственное профессиональное и личностное развитие, предпринимательскую деятельность в профессиональной сфере, использовать знания по правовой и финансовой грамотности в различных жизненных ситуациях.
ОК 04	Эффективно взаимодействовать и работать в коллективе и команде.
ОК 05	Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста.
ОК 06	Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей, применять стандарты антикоррупционного поведения.

## 2. Тематический план.

По специальности:

40.02.04 Юриспруденция

<b>Вид учебной работы</b>	<b>Количество часов</b>
<b>Обязательная аудиторная учебная нагрузка (всего)</b>	123
в том числе:	
лекционные занятия	53
практические занятия	70
<i>Итоговая аттестация в форме дифференцированного зачета</i>	

## **Практическое занятие №1**

**Тема работы:** Правовые нормы информационной деятельности. Стоимостные характеристики информационной деятельности.

**Цель работы:** научиться пользоваться образовательными информационными ресурсами, искать нужную информацию с их помощью, овладеть методами работы с программным обеспечением.

### **Содержание отчёта по практической работе:**

- тема и цель работы;
- тексты статей УК РФ для задания 1;
- заполненная таблица для задания 2;
- ответы на контрольные вопросы;
- вывод по практической работе.

## **Теоретическая часть**

### **1. Правовые нормы информационной деятельности**

Информация является объектом правового регулирования. Информация не является материальным объектом, но она фиксируется на материальных носителях. Первоначально информация находится в памяти человека, а затем она отчуждается и переносится на материальные носители: книги, диски и другие накопители, предназначенные для хранения информации.

Правовое регулирование — процесс целенаправленного воздействия государства на общественные отношения при помощи специальных юридических средств и методов, которые направлены на их стабилизацию и упорядочивание. Правовое регулирование является одним из составных элементов правового воздействия, которое по содержанию намного шире его и включает в себя не только целенаправленную деятельность по упорядочиванию общественных отношений, но и косвенное воздействие правовых средств и методов на субъектов, непосредственно не подпадающих под правовое регулирование.

Правовое регулирование в информационной сфере является новой и сложной задачей для государства. В Российской Федерации существует ряд законов в этой области. Решение проблемы защиты информации во многом определяется теми задачами, которые решает пользователь, как специалист в конкретной области. В настоящее время для защиты от несанкционированного доступа к информации все более часто используются биометрические системы идентификации. Используемые в этих системах характеристики являются неотъемлемыми качествами личности человека и поэтому не могут быть утерянными или поддельными.

Информационно-правовые нормы регулируют обособленную группу общественных отношений применительно к особенностям информационной сферы; задают содержание прав и обязанностей субъектов, участвующих в правоотношении. Информационно-правовые нормы регулируют взаимоотношения граждан, СМИ, организаций, фирм между собой, их взаимные права и обязанности.

Информационная безопасность Российской Федерации – состояние защищенности её национальных интересов в информационной сфере, определяющихся совокупностью сбалансированных интересов личности, общества и государства.

В связи с возрастающим значением информационных ресурсов предприняты



ряд правовых мер для их охраны и защиты. Многие черты информационного общества уже присутствуют в современной жизни развитых стран. Жизненно важной для общества становится проблема информационной безопасности действующих систем хранения, передачи и обработки информации.

О важности проблемы информационной безопасности свидетельствуют многочисленные факты. Более 80% компьютерных преступлений осуществляется через глобальную сеть Интернет, которая обеспечивает широкие возможности злоумышленникам для нарушений в глобальном масштабе.

В соответствии с действующим уголовным законодательством Российской Федерации под преступлениями в сфере компьютерной информации понимаются совершаемые в сфере информационных процессов и посягающие на информационную безопасность деяния, предметом которых являются информация и компьютерные средства. Неправомерный доступ к компьютерной информации (ст. 272 УК РФ), а также создание, использование и распространение вредоносных программ для ЭВМ (ст. 273 УК РФ) совершаются только путём действий, в то время как нарушение правил эксплуатации ЭВМ, системы ЭВМ или их сети (ст. 274 УК РФ) — путём как действий, так и бездействием.

## **2. Стоимостные характеристики информационной деятельности**

Современные информационные технологии обеспечиваются средствами компьютерной и коммуникационной техники. А их использование требует капитальных вложений. Поэтому, перед внедрением информационных технологий должна быть рассчитана эффективность их применения.

Эффективность автоматизированного преобразования информации – это целесообразность применения средств вычислительной и организационной техники при формировании, передаче и обработке данных.

Эффективность делится на расчетную и фактическую: первая определяется на стадии проектирования автоматизации информационных работ, вторая – по результатам внедрения автоматизированных ИТ.

Обобщенным критерием экономической эффективности является минимум затрат живого и овеществленного труда. При этом установлено, что чем больше участков прикладных работ автоматизировано, тем эффективнее используется техническое и программное обеспечение. Экономический эффект от внедрения вычислительной и организационной техники подразделяют на прямой и косвенный.

*Прямая экономическая эффективность информационных технологий* – экономии материально-трудовых ресурсов и денежных средств, от сокращения персонала, связанного с реализацией информационных задач (управленческий персонал, инженерно-технический персонал и т. д.), уменьшения фонда заработной платы, расхода основных и вспомогательных материалов вследствие автоматизации конкретных видов информационных работ.

*Косвенная эффективность* проявляется в конечных результатах деятельности организаций. При анализе косвенной эффективности основным показателем является повышение качества управления, которое, как и при прямой экономической эффективности, ведет к экономии живого и овеществленного труда.

Экономическую эффективность определяют с помощью трудовых и стоимостных показателей. Основным при расчётах является метод сопоставления дан-

ных базисного и отчётного периодов. В качестве базисного периода при переводе отдельных работ на автоматизацию принимают затраты на обработку информации до внедрения информационной технологии (при ручной обработке), а при совершенствовании действующей системы автоматизации информационных работ - затраты на обработку информации при достигнутом уровне автоматизации. При этом пользуются абсолютными и относительными показателями.

Наряду с трудовыми показателями, рассчитываются и стоимостные показатели, т.е. определяются затраты (в денежном выражении) на обработку информации при базисном и отчётном вариантах.

В таблицах 1 и 2 приведены обозначения переменных и формулы для расчёта необходимых показателей.

**Таблица 1 – Обозначение переменных**

Показатель	Переменная
Затраты на ручную обработку документов (чел/час)	$T_0$
Затраты при использовании информационных технологий (чел/час)	$T_1$
Затраты (в денежном выражении) на обработку информации при базисном варианте	$C_0$
Затраты (в денежном выражении) на обработку информации при отчётном варианте	$C_1$
Затраты на техническое обеспечение	30
Затраты на программное обеспечение	ПО,
Коэффициент эффективности	$K_{эф}$

**Таблица 2 – Формулы расчёта показателей эффективности**

Показатель	Формула
Абсолютный показатель экономической эффективности	$T_{ЭК} = T_0 - T_1$
Относительный индекс производительности труда	$J_{пт} = \frac{T_1}{T_0}$
Абсолютный показатель стоимости	$C_{ЭК} = C_1 - C_0$
Индекс стоимости затрат	$J_{сз} = \frac{C_1}{C_0}$
Срок окупаемости затрат	$T_{ок} = \frac{(30 + ПО) K_{эф}}{C_0 - C_1},$

Технологические стадии разработки автоматизированных информационных

технологий и систем регламентируются российскими и международными стандартами.

### Практическая часть

**Задание 1.** Найти в Интернете закон РФ «Уголовный Кодекс российской Федерации» и найти указанные в теоретическом материалы статьи, касающиеся преступлений в сфере компьютерной информации.

Скопируйте в отчёт тексты этих статей.

**Задание 2.** Рассчитайте необходимые показатели по данным таблицы 3 и заполните таблицу 4.

**Таблица 3 – Исходные данные для задания**

Показатель	Переменная	Значение
Затраты на ручную обработку документов (чел/час)	$T_0$	1500
Затраты при использовании информационных технологий (чел/час)	$T_1$	1020
Затраты (в денежном выражении) на обработку информации при базисном варианте, тыс. руб.	$C_0$	230
Затраты (в денежном выражении) на обработку информации при отчётном варианте, тыс. руб.	$C_1$	120

**Таблица 4 – Результаты расчётов**

Показатель	Формула	Результат расчёта
Абсолютный показатель экономической эффективности, чел-час	$T_{ЭК} = T_0 - T_1$	
Относительный индекс производительности труда	$J_{ПТ} = \frac{T_1}{T_0}$	
Абсолютный показатель стоимости, тыс.руб.	$C_{ЭК} = C_1 - C_0$	
Индекс стоимости затрат	$J_{СЗ} = \frac{C_1}{C_0}$	

### Контрольные вопросы

1. Что такое правовое регулирование?
2. Что такое информационная безопасность Российской Федерации?
3. Что такое прямая и косвенная эффективность информационных технологий?

## WORD

**Практическое занятие №2** «Знакомство. Панель инструментов. Структура окна. Правила ввода текста. Методы оформления текста»

Необходимые для выполнения работ знания:

Форматирование абзацев: шрифты, форматирование символов: изменение шрифта, размера, цвета. Выравнивание абзацев: междустрочный интервал, красная строка, расстояние между абзацами.

Цель работы: уметь привести данный текст к образцу, используя в работе окна Абзац и Шрифт

Сведения из теории:

Правила ввода текста: 1) клавиша Enter –нажимается только для перехода к новому абзацу или для подтверждения какого-либо действия; 2) для перехода на следующую строку внутри одного абзаца необходимо воспользоваться клавишами SHIFT+ENTER; 3) Формат-абзац: отступ для всего текста слева (справа); отступ только первой строки; междустрочный интервал; интервал между абзацами до и после; 4) Формат-Шрифт: Вкладка Шрифт: изменение шрифта, размера, различные виды подчеркивания, видоизменение; вкладка интервал: расстояние между буквами –интервал разреженный; вкладка Анимация –спецэффекты. 5) Пункт меню файл: сохранить, создать, открыть, параметры страницы.

Контрольные вопросы:

- 1) Какие команды находятся в пункте меню Абзац и как с ними работать.
- 2) Что можно изменить с помощью окна Шрифт.
- 3) Как создать документ и как его сохранить.

План работы:

### **1 Наберите текст:**

Появление и развитие информатики.

Термин информатика возник в 60-х гг. во Франции для названия области, занимающейся автоматизированной обработкой информации с помощью электронных вычислительных машин. Французский термин *informatique* (информатика) образован путем слияния слов *information* (информация) и *automatique* (автоматика) и означает «информационная автоматика или автоматизированная переработка информации». В англоязычных странах этому термину соответствует синоним *computer science* (наука о компьютерной технике). Выделение информатики как самостоятельной области человеческой деятельности в первую очередь связано с развитием компьютерной техники. Причем основная заслуга в этом принадлежит микропроцессорной технике, появление которой в середине 70-х гг. послужило началом второй электронной революции. С этого времени элементарной базой вычислительной машины становятся интегральные схемы и микропроцессоры, а область, связанная с созданием и использованием компьютеров, получила мощный импульс в своем развитии. Информатика трактовалась как «комплексная научная и инженерная дисциплина, изучающая все аспекты разработки, проектирования, создания, оценки, функционирования основанных на ЭВМ систем переработки информации, их применения и воздействия на различные области социальной практики.» Информатика- это об-



ласть человеческой деятельности, связанная с процессами преобразования информации с помощью компьютеров и их взаимодействием со средой применения.

**2 С помощью клавиши Enter разбейте текст на абзацы.**

**3 С помощью средств форматирования WORD сделайте 1 абзац:** шрифт Arial, размер 11, выравнивание по левому краю, отступ первой строки 2 см.; интервал после 12 пт.; **2 абзац:** шрифт Courier, размер 12, выравнивание по центру, интервал после 10пт.; **3 абзац** шрифт Arial, размер 13, выравнивание по ширине, отступ первой строки 1,5 см., интервал после 12пт.; **4 абзац:** шрифт Courier, размер 11, выравнивание по правому краю, интервал после 10 пт.; **5 абзац:** шрифт Arial, размер 13, выравнивание по ширине, отступ первой строки 1,5 см.

**4 Внимательно посмотрите и измените по образцу начертание некоторых слов (курсив, подчеркивание и т. д.)**

**5 После всех изменений должен получиться текст. Проверьте себя:**

#### **Появление и развитие информатики.**

Термин информатика возник в 60-х гг. во Франции для названия области, занимающейся автоматизированной обработкой информации с помощью электронных вычислительных машин.

Французский термин *informatique* (информатика) образован путем слияния слов *information* (информация) и *automatique* (автоматика) и означает «информационная автоматика или автоматизированная переработка информации». В англоязычных странах этому термину соответствует синоним computer science (наука о компьютерной технике).

Выделение *информатики* как самостоятельной области человеческой деятельности в первую очередь связано с развитием компьютерной техники. Причем основная заслуга в этом принадлежит микропроцессорной технике, появление которой в середине 70-х гг. послужило началом второй электронной революции. С этого времени **элементарной базой** вычислительной машины становятся интегральные схемы и микропроцессоры, а область, связанная с созданием и использованием компьютеров, получила мощный импульс в своем развитии.

Информатика трактовалась как «комплексная научная и инженерная дисциплина, изучающая все аспекты разработки, проектирования, создания, оценки, функциони-

рования основанных на ЭВМ систем переработки информации, их применения и воздействия на различные области социальной практики.»

***Информатика**- это область человеческой деятельности, связанная с процессами преобразования информации с помощью компьютеров и их взаимодействием со средой применения.*

### **Практическое занятие №3 «Размещение графики в документе. Вставка рисунка, изменение формата рисунка»**

Цель работы: уметь добавлять рисунки в документ из имеющейся библиотеки Worda, добавлять объект WordArt, накладывать объекты друг на друга, перемещать их; повторение команд Абзац, Шрифт.

Необходимые для выполнения работ знания:

Размещение графики в документе. Вставка рисунка, изменение формата рисунка, свободное перемещение рисунка. Рисование в самом документе (использование инструментов панели Рисование).

Сведения из теории:

Вставка –Рисунок –Картинки, выбрать категорию, выбрать рисунок: копировать, вставить. Окно вставки Рисунка свернуть. Переместить Рисунок: войти в формат рисунка, перейти на вкладку положение, выбрать за текстом.

Вставка –Рисунок –объект Word Art, выбрать образец, ввести текст. На экране появиться панель «WordArt», с помощью которой можно изменить заливку текста, цвет линий, установить дополнительный объем, тень.

Контрольные вопросы:

- 1) В каком пункте строки меню находится команда Рисунок.
- 2) Как добавить Рисунок в Документ.
- 3) Что можно изменить используя окно Формат рисунка...
- 4) Как добавить в документ фигурный текст
- 5) Какие можно изменить форматы у объекта Word Art

План работы:

- 1) привести текст к образцу, используя палитру цветов.
- 2) К тексту примените: основной текст шрифт Times New Roman, размер 13, отступ первой строки 2 см;
- 3) слова удобство, качество, цена: цвет по образцу, интервал перед 12 пт, отступ слева 3 см;
- 4) слова изделие №9: шрифт Courier, размер 12;
- 5) последний интервал между абзацами 50 пт;
- 6) рисунок и надпись по образцу, установите яркость рисунка 70%;
- 7) проверьте себя.

### Изделие №9-это №1 на рынке.

Сбылись мечты миллионов. Наконец-то. Уникальное по своим характеристикам изделие №9 поступило в продажу. Теперь вы можете выбросить свои старые швабры, соковыжималки и громкоговорители. Их заменит изделие №9- универсальное средство с бесплатным дополнительным комплектом.

### УДОБСТВО

Изделие №9 занимает не более 10 см<sup>2</sup> площади и приводится в действие нажатием всего одной кнопки. Встроенный Анализатор Функций сам определяет цель, для которой был включен прибор. Изделие не требует батареек и работает без электрошнура. Единственное, что необходимо сделать- раз в 7 лет производить получасовую зарядку блока питания, о необходимости которой напоминает красная лампочка.

### КАЧЕСТВО

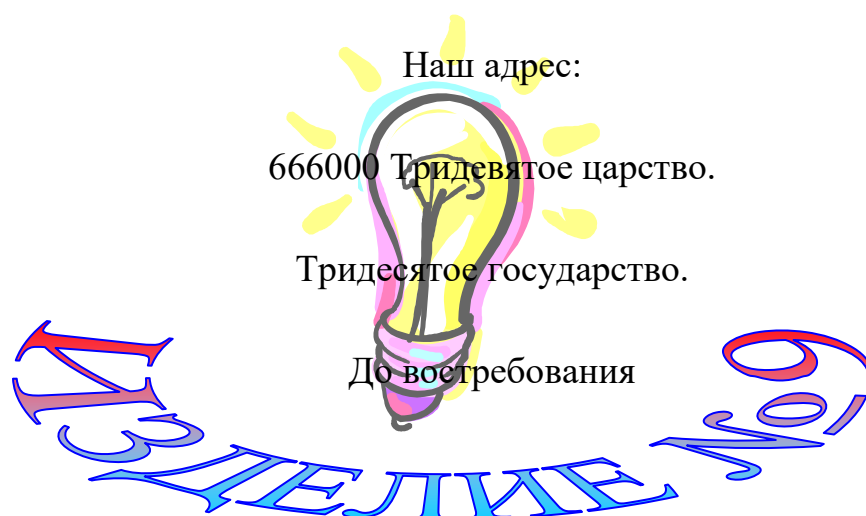
Ученые самого известного университета разрабатывали этот прибор 27 лет. Его невозможно сломать или испортить. Даже концентрированная серная кислота не оказывает никакого воздействия на его защитный кожух.

Попробуйте сами. Прибор выдерживает давление в 122 атмосферы и ударную волну на удалении 150 метров от эпицентра. 10-ти летняя гарантия обеспечивает максимум доверия потребителя.

### ЦЕНА

В это трудно поверить, но цена прибора всего \$99,99. Любая система изделия №9, вышедшая из строя заменяется бесплатно даже посмертно. При одновременной покупке трех изделий- четвертое поставляется бесплатно.

Все, что вы хотели до сих пор, не имеет никакого значения. Теперь у вас есть изделие №9





## **Практическое занятие №4 «Списки. Изменение формата списка»**

Необходимые для выполнения работ знания: в каком пункте меню находится команда список, как изменить формат списка.

Цель работы: уметь сделать список на основе уже набранного текста; изменить формат списка, например с нумерованного на маркированный, поменять табулятор и отступ всего списка слева.

Сведения из теории: формат -список, из предложенных вкладок выбрать маркированный, нумерованный или многоуровневый, при необходимости изменения вида номера или табулятора воспользоваться в этом окне кнопкой **Формат...**(Изменить...)

Контрольные вопросы:

- 1) Какие Можно организовать списки и как с ними работать.
- 2) Как можно изменить один список на другой.
- 3) Как можно сделать список на основе уже набранного текста.

План работы:

- 1) установите поля Левое, Правое –1см; Верхнее, Нижнее –3см.
- 2) Шрифт Times New Roman, размер 12, отступ первой строки 2см, интервал перед абзацем 66пт, после 12 пт.
- 3) Первый список «--»: расположение маркера 0,5 см, табуляция 1см
- 4) Второй список «⊗»: расположение маркера 3см
- 5) Третий список нумерованный: табуляция 1см, обратите внимание на расположение текста.
- 6) Рисунок по образцу

## ПОКРЫТОСЕМЕННОЕ РАСТЕНИЕ И ЕГО ОРГАНЫ



**Корень** – один из главных вегетативных органов у покрытосеменных. Он выполняет важные *функции*:

- закрепление растений в почве;
- активное поглощение из почвы воды и минеральных веществ;
- синтез некоторых органических веществ;
- запас питательных веществ;
- осуществление вегетативного размножения.

Из набухшего семени прежде всего показывается зародышевый корешок. Из него развивается *главный корень*. Он растет отвесно вниз и выделяется большей длиной и толщиной; похож на стержень. Затем на главном корне появляются *боковые корни*. Они развиваются в строгой последовательности: от основания до кончика (но не на самой верхушке). Многие растения имеют *придаточные корни*, отрастающие от стебля и других органов.

Совокупность всех корней растения составляет *корневую систему растения*.

Корневые системы бывают двух типов:

- ⊗ *стержневая* – хорошо развит главный корень (береза, петрушка);
- ⊗ *мочковатая* – главный корень незаметен среди других корней (злаки, лук).

Во внутреннем строении молодого корня можно выделить следующие три *зоны корня*.

1. *Зона деления*, прикрытая *корневым чехликом*. Здесь происходит непрерывное деление клеток. Эти клетки мелкие, с тонкими стенками.
2. *Зона роста и растяжения*. В ней происходит удлинение и рост клеток, в результате чего корень растет в длину.
3. *Зона всасывания*, в которой происходит формирование тканей. Здесь образуются *корневые волоски* – выросты наружных клеток корня длиной 0,1-1,5-8 мм, имеющие тонкие слизистые оболочки и крупные центральные вакуоли. На поперечном срезе под слоем клеток, образующих корневые волоски, располагаются слои клеток *коры корня*.
4. *Проводящая зона*. В этой зоне корневые волоски отмирают и разрушаются, а наружные стенки утолщаются. Живое содержимое их отмирает.

## «Создание многоуровневого списка»

### План работы:

- 1) Сделать рамку по образцу.
- 2) Установите поля Левое, Правое –1см; Верхнее, Нижнее –3см.
- 3) Шрифт Arial, размер 12
- 4) Список по образцу.
- 5) Рисунок и выравнивание по образцу.

## МНОГОУРОВНЕВЫЕ СПИСКИ

⊕ **Астрономия** (гр. Astronomia) – наука о небесных телах, системах небесных тел и о вселенной в целом.

### 1. Солнечная система

#### 1.1. Солнце

#### 1.2. Планеты Земной группы

1.2.1. Меркурий

1.2.2. Венера

1.2.3. Земля

1.2.4. Марс

1.2.5. Плутон

#### 1.3. Планеты гиганты

1.3.1. Юпитер

1.3.2. Сатурн

1.3.3. Уран

1.3.4. Нептун

#### 1.4. Малые тела

1.4.1. Метеориты

1.4.2. Кометы

1.4.3. Астероиды



⊕ **Астрофизика** – раздел астрономии, изучающий строение небесных тел и происходящие в них процессы.

### 2. Астрономия

#### 1.1. Галактики

#### 1.2. Планеты

1.2.1. Венера

1.2.2. Земля

1.2.3. Марс

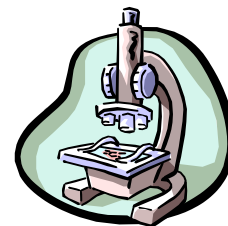
#### 1.3. Созвездия

1.3.1. Большая медведица

1.3.2. Лебедь

1.3.3. Орион

#### 1.4. Астероиды



⊕ **Космос** – мировое пространство, Вселенная.

### 3. Космос

#### 3.1. Солнечная система

##### 3.1.1. Планеты

3.1.1.1. Малые: Плутон, Венера, Меркурий

3.1.1.2. Большие: Юпитер, Сатурн

##### 3.1.2. Звезды

#### 3.2. Галактики

#### 3.3. Всякая всячина

3.3.1. Метеориты

3.3.2. Космический мусор

⊕ **Гороскоп** – предсказание по звездам.

### 4. Зодиакальный круг

#### 4.1. Стихии

4.1.1. Земля, Вода, Воздух, Огонь

#### 4.2. Знаки

4.2.1. Рак, Лев, Дева, Овен

4.2.2. Рыбы, Весы, ...

#### 4.3. Годы

4.3.1. Бык, Крыса, Тигр, Кот...





## **Практическое занятие №5 «Вставка автофигуры, нестандартная заливка автофигур.»**

Цель работы: уметь добавить в документ автофигуру, изменить ее формат.

Необходимые для выполнения работ знания: как добавить в документ автофигуру, как изменить ее формат, нестандартная заливка автофигуры.

Сведения из теории: Вставка –Рисунок –Автофигуры, обычно внизу экрана появляется панель Автофигуры, где необходимо выбрать нужную и нарисовать на рабочем листе. Далее в окне Формат автофигуры можно изменить ее рамку, цвет рамки, заливку и т.д.

Цель работы: используя уже изученные средства форматирования текстового редактора Word создать предложенные автофигуры по образцу.;  
закрепление Вставки Рисунка, объекта.

Контрольные вопросы:

- 1) Как добавить в рабочий документ автофигуру.
- 2) Как изменить ее формат
- 3) Какие возможности заливки автофигур находятся в окне Формат автофигуры

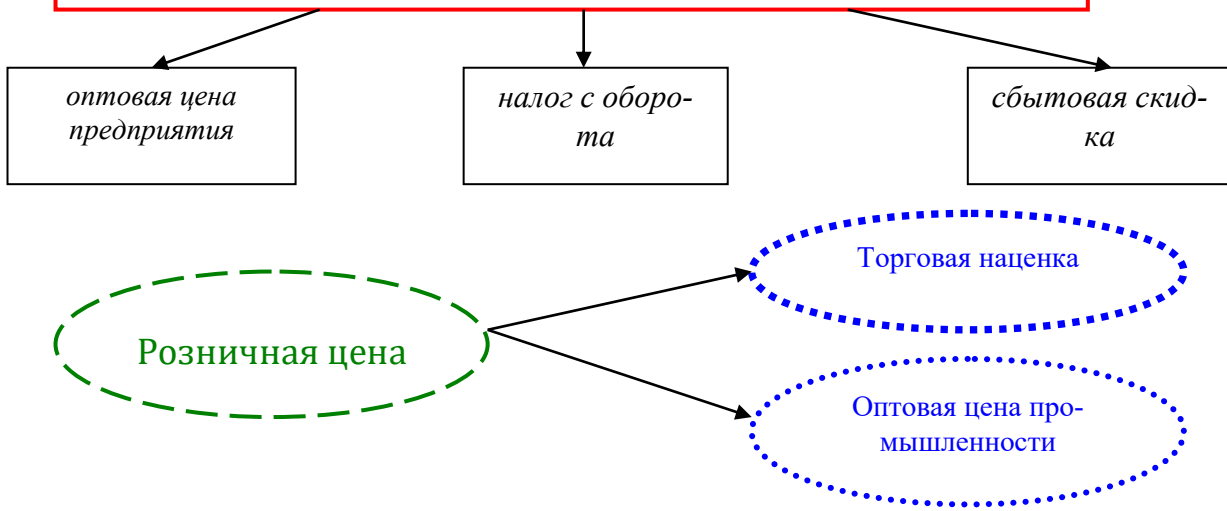
План работы:

- 1) установите поля Левое, Правое –1 см; Верхнее, Нижнее –3 см
- 2) вставьте автофигуры по образцу, измените в соответствии с образцом формат автофигур.
- 3) Шрифт Arial, размер 12, начертание по образцу.

# ПОНЯТИЕ ОБ ОПТОВЫХ И РОЗНИЧНЫХ ЦЕНАХ



## ОПТОВАЯ ЦЕНА ПРОМЫШЛЕННОСТИ



- 4) Сделать разрыв листа с помощью команды Раздел  
1 Составить блок – схему «Составные части компьютера»

■ главные части компьютера

● составные части главных частей

▲ периферийные устройства, подключающиеся к определенным частям.

- 2 В каждой фигуре вставить название части.
- 3 Стрелками обозначить подключение, отображающее ввод вывод информации.
- 4 Залить фигуры по образцу (используйте два цвета).
- 5 Лист сделать с альбомной ориентацией

## **Практическое занятие №6** «Создание таблицы, оформление таблицы, изменение формата отдельных ячеек.»

Цель работы: уметь добавить в документ таблицу, изменить ее формат (заливка ячеек, наклон текста, различные виды оформления, выравнивания текста).

Необходимые для выполнения работы знания: как добавить таблицу, как изменить формат элементов таблицы.

Сведения из теории: Таблица – Добавить – Таблица, в появившемся диалоговом окне установить количество строк и столбцов. Изменить формат элемента таблицы: его необходимо выделить и через контекстное меню выбрать соответствующую команду (например, для изменения заливки ячеек или границы необходимо воспользоваться командой Границы и заливка, перейти на соответствующую вкладку у выбрать необходимый формат)

Контрольные вопросы: 1) как добавить в документ Таблицу;  
2) Как изменить ширину столбца, высоту строки;  
3) Как возможно расположить текст в ячейках таблицы;  
4) Как изменить заливку ячеек; границы отдельно взятой ячейки, всей таблицы целиком.

План работы: 1) установите поля Левое, Правое –2см; Верхнее, Нижнее –1см  
2) Добавить таблицу с указанными строками и столбцами.  
3) Ввести предложенный текст: Шрифт Таhоmа, размер 11.  
4) Выравнивание, заливка и оформления по образцу.  
5) Для последних семи строк выполнить команды: выделить, Таблица, Автоподбор, Выровнять высоту строк.  
6) С помощью программки Калькулятор подсчитать и вставить данные в строку  
Весь мир

РЕГИОН	Для всех земельных ресурсов региона, %				
	Пашня, сады	Луга и пастбища	Леса	Земли занятые населенными пунктами, промышлен. объектами, транспортом и магистралями	Малопродуктивные и непродуктивные зоны
Европа	32	19	26	5	18
Азия	21	15	21	2	41
Африка	11	23	26	1	39
Северная Америка	12	18	33	3	34
Южная Америка	8	19	47	1	25
Австралия и Океания	5	51	8	1	335
<b>Весь мир</b>					

1) Подготовьте таблицу

Виды средств	По балансу на начало года	На конец года		Отклонение фактических запасов на конец года	
		Норматив	Фактически и по балансу	От норматива	От запасов на начало года
I. Норматируемые оборотные средства					
1. Товары в предприятиях розничной торговли и в пути по себестоимости					
2. Прочие товарно-материальные ценности					

3. Касса					
4. Денежные средства в пути и денежные документы					
ИТОГО					
II. Нормируемые оборотные средства					
1. Оборотные средства сезонного хранения					
2. Ненормируемые денежные средства					
3. Дебиторы					

8) Заполните таблицу произвольными числовыми данными, строку ИТОГО подсчитайте с помощью Калькулятора.

9) Снизу таблицы добавьте еще одну строку ИТОГО и подсчитайте для второй части таблицы

10) Установите шрифт Arial, размер 12.

11) Произвольным образом измените заливку ячеек оглавления таблицы и поменяйте рамки.

## **Практическое занятие №7 «Колонки. Многоколоночный текст. Форматирование колонок»**

Цель работы: уметь добавить в документ Буквицу, изменить ее формат. Разбить текст на колонки, закончить колонку на конкретном слове.

Необходимые для выполнения работы знания: как разбить уже набранный текст на колонки.

Сведения из теории: текст необходимо выделить, Формат –Колонки, выбрать необходимое число колонок, при необходимости воспользоваться командой Разделитель, изменить ширину колонок и расстояние между ними. Чтобы закончить колонку в выбранном месте: установить курсор там, где должна заканчиваться колонка и выполнить команды Вставка –Разрыв –Новая колонка

Контрольные вопросы:

- 1) как разбить текст на колонки;
- 2) что можно изменить в окне Колонки...;
- 3) можно ли закончить колонку в выбранном произвольном образом месте.

План работы:

- 1) установите поля Левое, Правое –1 см; Верхнее, Нижнее –3 см
- 2) Набрать текст: тип шрифта Таhоmа, размер шрифта 11.
- 3) Вставить буквицу в начале текста, изменить ее цвет на красный, шрифт ScriptS.
- 4) Разбить страницу на две колонки, в первой колонке ввести маркированный список, во второй маркированный список с помощью рисунка, весь текст выравнивать по ширине.
- 5) Сделать интервал между абзацами 12 пт.
- 6) Вставить рисунок на тему «учеба»

## **Открытие и сохранения документа**

**Д**ля того чтобы работать с документом, Вам нужно сначала открыть его и отобразить на экране. После этого Вы можете редактировать его, оформлять и печатать. Если хотите сохранить свою работу, Вы должны сохранить документ на диске.

### **® Как открыть документ**

Ψ После запуска системы Word автоматически открывается новый пустой документ с именем Документ 1. С ним можно сразу же работать, вводя текст

Ψ Вам необязательно начинать работу с новым документом с нуля. Вы можете использовать один из шаблонов, часто встречающихся типов, например, писем или отчетов.

Ψ Для того чтобы открыть существующий документ, используйте команду Открыть (меню Файл) или нажмите кнопку открыть на стандартной панели инструментов.

Ψ Вы можете открыть сразу несколько документов и перемещаться между ними в процессе работы.

Ψ Если Вы забыли имя, какого – то документа или место, куда его записали, воспользуйтесь командой Поиск файла, которой можно пользоваться и в том случае, когда необходимо открыть, скопировать, переместить, напечатать или удалить сразу несколько документов.

### **® Как сохранить документ**

● После того, как Вы открыли документ, он копируется и отображается в окне документов. Во время работы эта копия документа меняется. Для того чтобы сохранить работу, следует чаще сохранять документ на диске.

● Для этого можно нажать кнопку Сохранить стандартной панели инструментов, или воспользоваться возможностью автоматического сохранения.

● После сохранения документа он остается на экране, так что Вы имеете возможность продолжать работу с ним.

● Если документ Вам больше не нужен, закройте его, не забыв сохранить в нем все изменения.



## Практическое занятие №8 «Текстовые эффекты Word Art. Редактор формул Microsoft Equation»

Цель работы: уметь добавить в документ формулу.

Необходимые для выполнения работы знания: как добавить в документ формулу

Сведения из теории: Вставка –Объект –Microsoft Equation, на экране появиться панель Формула, где будут представлены образцы, выбрать необходимый, числовые и буквенные значения ввести с клавиатуры, щелкнуть мышкой в любом месте. Для редактирования уже готовой формулы надо щелкнуть левой кнопкой мыши по области формулы два раза –на экране снова появиться панель Формула, с помощью которой и производим редактирование.

Контрольные вопросы:

- 1) как добавить формулу в документ;
- 2) как внести в формулу изменения;
- 3) можно ли перемещать формулу по рабочему документу произвольным образом.

План работы:

- 1) установите поля Левое, Правое –2см; Верхнее, Нижнее –3см
- 2) Наберите формулы по образцу, используя встроенный редактор формул.
- 3) Для каждой формулы выберите положение за текстом и «раскидайте» их произвольным образом по листу.

$$\int_0^{\pi} \sin dx = 2$$

$$x = x_0 + V_0 t + a \frac{t^2}{2}$$

$$Q = \Delta U + \int_{v1}^{v2} p dV$$

$$\omega = \sqrt{\frac{1}{LC}} - \left( \frac{R}{2L} \right)^2$$

$$\vec{F} = m \vec{a}$$

$$\vec{E} = \sum_{i=1}^{i=k} \vec{E}_i$$

$$\left\{ y = \frac{x - Vt}{\sqrt{1 - \beta^2}} \right.$$

$$dz = y' + y''$$

- 4) Добавьте к листу заголовок, с помощью объекта WordArt «Некоторые физические законы»
- 5) Перейдите на второй лист, где выполните работу согласно образцу.
- 6) Шрифт Times New Roman, размер 13



**Т А Б Л И Ц А 7.1. Техничко-экономические параметры трансформаторов**

<i>Вар.</i>	$S_H$ , <i>МВА</i>	$\Delta P_x$ , <i>кВт</i>	$U_{KB-H}$ , <i>%</i>	$U_{KC-H}$ , <i>%</i>	$I_x$ , <i>%</i>	$K$ , <i>Тыс. руб.</i>
1	40,0	34	10,5	30	0,55	88
2	36,0	50,0	245,0	32,2	0,5	110,3

Напряжение может быть рассчитано по формуле Г.А. Илларионова:

$$U = \frac{1000}{\sqrt{\frac{500}{l} + \frac{2500}{P}}}, \quad (6.1)$$

где  $l$  –длина линии от РП-220 до ГПП (2,3 км)

$P$  –передаваемая мощность на одну цепь (44,225 МВт)

Подставляя значения получим

$$U = \frac{1000}{\sqrt{\frac{500}{2,3} + \frac{2500}{44,225}}} = 71 \text{ кВ}$$

## Практическое занятие № 9 «Диаграмма»

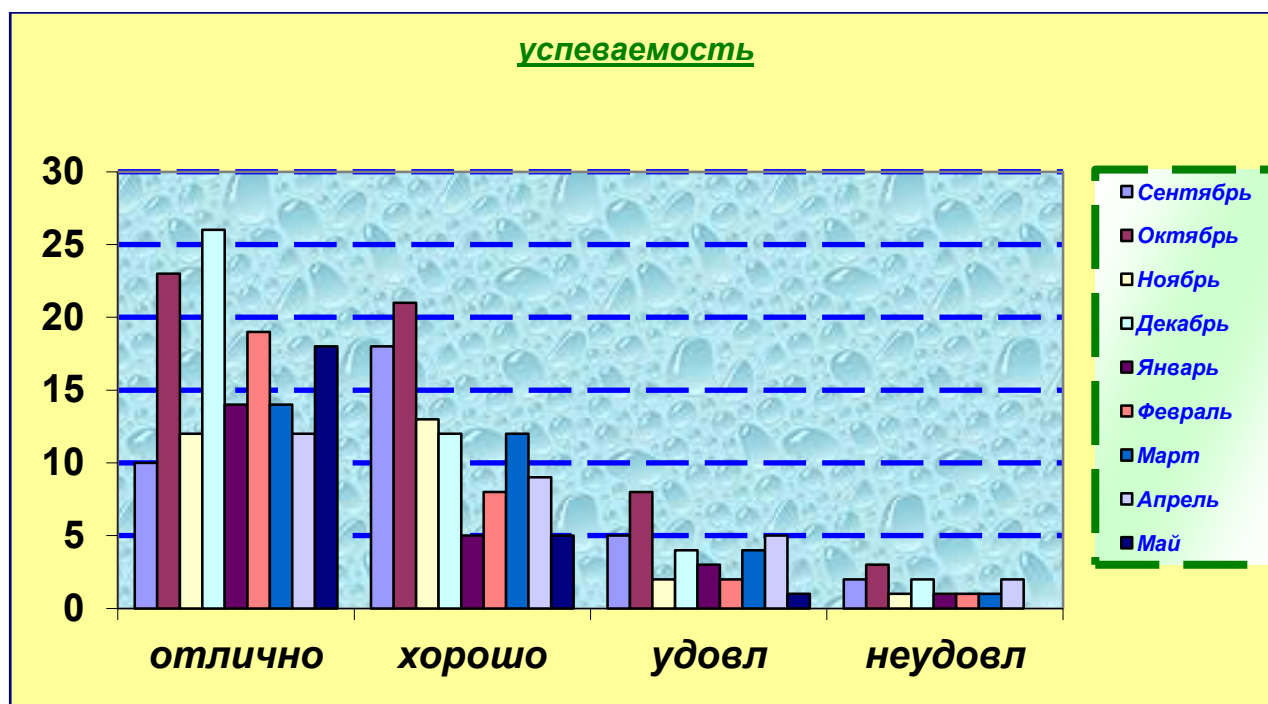
Цель работы: уметь добавить в документ Диаграмму, изменить ее формат согласно предложенному образцу.

План работы:

- 1) установите поля Левое, Правое –2,5 см; Верхнее, Нижнее –3см
- 2) Заполнить предложенную таблицу.

	<b>отлично</b>	<b>хорошо</b>	<b>удовл</b>	<b>неудовл</b>
Сентябрь	10	18	5	2
Октябрь	23	21	8	3
Ноябрь	12	13	2	1
Декабрь	26	12	4	2
Январь	14	5	3	1
Февраль	19	8	2	1
Март	14	12	4	1
Апрель	12	9	5	2
Май	18	5	1	0

- 3) выделите всю таблицу целиком и выполните команды Вставка –Диаграмма
- 4) выделяя отдельно каждый элемент, измените его формат предложенному образцу



- 5) наберите таблицу температуры за прошедшую неделю и постройте Линейную диаграмму, измените формат отдельных ее элементов по вашему усмотрению.

## **EXCEL**

### **Практическое занятие № 10**

**Тема:** Запуск. Базовые понятия электронных таблиц: ячейка, формулы, рабочая книга. Создание, открытие и сохранение рабочих книг. Операции с листами рабочей книги.

**Цель работы:** уметь запускать программу EXCEL, познакомить с механизмом работы: ячеек, формул, рабочих книг.

#### **Необходимые для выполнения работы знания:**

Уметь запускать программу с помощью ярлыка и дерева каталогов. Рабочая книга. Листы. Ячейка. Формула. Операции с листами. Ввод границ. Создание и сохранение документов под определенным именем.

#### **Сведения из теории:**

1) Для запуска выбрать с помощью мыши ярлык для Excel. 2) Документ состоит из рабочей книги и листов. 3) Название ячейки состоит из названия столбца и названия строки (A1). 4) Операции с листами: подвести курсор мыши к меню листов и названию листа вызвать контекстное меню, с помощью пунктов меню можно: переименовать, добавить, удалить листы, для подтверждения нажать ENTER. 5) Каждая формула начинается со знака (=). 6) Выделить определенные ячейки, вызвать Формат – Ячейки, вкладка Границы, выбрать определенный тип линии, цвет линии и тип обрамления. 7) при закрытии документа делается запрос для сохранения документа в определенный каталог, под определенным именем.

#### **План работы:**

- 1.) Запустите программу EXCEL с помощью ярлыка на рабочем столе.
- 2.) На листе 1, В ячейке A1 поставьте «X», в ячейке B1 поставьте «Y». По столбцу «X» введите интервал [-6;6], по столбцу «Y» напротив каждого значения введите формулу, вычисляющую данную функцию.
- 3.) Откройте Лист 2, сделайте подобное вычисление со второй функцией.
- 4.) Переименуйте Лист 1 – «Знакомство», Лист 2 – «Функция»
- 5.) Откройте Лист 3 – «Объединение», скопируйте содержимое листов «Знакомство» и «Функция» на Лист «Объединение».
- 6.) Поставьте Лист «Объединение» впереди Листов «Знакомство» и «Функция».
- 7.) Удалите Листы «Знакомство» и «Функция».
- 8.) Создайте Лист 4 – «Функция», впечатайте заголовок «Вычисление функций» в ячейку A1 объедините ячейки с A1 по D1. Скопируйте содержимое Листа «Объединение» на Лист «Функция» под заголовок, введите обрамление таблицы.
- 9.) Сохраните на диск А данную книгу с именем «Практическая работа № 10 и расширением xls.
- 10) Позовите преподавателя для проверки.

#### **1) Лист 1 «Знакомство»**

Построить таблицу значений функции  $y=3,5x-2$  на интервале [-6;6].

## 2) Лист 2 «Функция»

Построить таблицу значений функций  $y=(k-20)*x-k$  на интервале  $[-4;4]$  с шагом 0,5, где  $k$  – номер вашего компьютера.

### Практическое занятие № 11

Тема: Техника ввода данных. Заполнение смежных ячеек и создание рядов.

**Цель работы:** Научиться разными способами вводить значения в ячейки. Пользоваться маркером заполнения.

#### Необходимые для выполнения работы знания:

Ввод в ячейки значений с помощью клавиатуры (вручную) и с помощью маркера заполнения (автоматически - по шаблону, по заданному шагу, названия дней недели и месяцев).

#### Сведения из теории:

1) С помощью клавиатуры вводим значения в ячейки.

2) Для автоматического ввода по шаблону необходимо ввести одно значение, затем навести курсор мыши на нижний правый угол выделенной ячейки до ее трансформации в маленький крестик, нажать и удерживать левую клавишу мыши и провести в нужном направлении по ячейкам, обращая внимания на всплывающие подсказки.

3) Для того чтобы ввести данные по заданному шагу необходимо ввести два значения для задания компьютеру шага и воспользоваться маркером заполнения.

#### План работы:

1.) Запустите программу EXCEL.

2.) Переименуйте листы по образцу.

3.) Создайте первую таблицу с помощью маркера заполнения и сделайте вычисления. Обрамите таблицу.

4.) Создайте вторую таблицу и сделайте те же самые действия что, и на первом листе.

5.) Создайте третью таблицу и с помощью формул сделайте вычисления, сначала столбец « $a_n$ », потом столбец « $S_n$ », вставьте обрамление по образцу.

6.) Сохраните рабочую книгу на диск А с именем «Практическая работа № 11. xls».

7.) Позвать учителя для проверки.

#### 1) Лист 1 «Знакомство»

Построить таблицу значений функции  $y=2*x*x^2$  на интервале  $[-6;6]$ .с шагом 0,2

#### 2) Лист 2 «Функция»

Построить таблицу значений функций  $y=5*x-1-k$  с шагом 0,5, где  $k$  – номер вашего компьютера.

### 3) Лист 3 «Прогрессия»

Дано  $a_1$ ,  $d$ ,  $n$ , с помощью формул  $a_n = a_1 + d$ ,  $S_n = (a_1 + a_n) \cdot n / 2$  заполните таблицу.

<i>Вычисление n-ого члена и суммы арифметической прогрессии</i>			
$d$	$n$	$a_n$	$S_n$
0,725	1	-2	
0,725	2		
0,725	3		
0,725	4		
0,725	5		
0,725	6		
0,725	7		
0,725	8		
0,725	9		
0,725	10		

## Практическое занятие № 12

Тема: Создание собственных форматов Excel.

Цель работы: научиться создавать и вводить собственные форматы.

Необходимые для выполнения работы знания:

Создание форматов. Применение форматов для данных.

Сведения из теории:

1) Для того чтобы ввести формат ячейки необходимо выделить определенную ячейку либо ячейки, выбрать Формат – Ячейки, вкладку Число, выбрать тип числового формата и согласиться с выполнением работы. Для того чтобы создать собственный формат необходимо выделить ячейку либо ячейки, выбрать Формат – Ячейки, вкладку Число, Тип числового формата – Все форматы и в строке Тип набрать собственный формат с помощью знаков (#, 0, “, букв), согласиться с выполнением задачи. Буквы при создании формата с обеих сторон закрываются кавычками (например 3,58 кг - #,##”кг”), для каждой цифры применять #.

План работы:

Переименовать листы согласно образцу. Ввести обрамление получившихся таблиц.

Пересчет делать с помощью формул.

### Лист 1 «Товарный чек»

Заполните столбцы «наименование», «количество», чтобы были разные единицы измерения (шт., банки, кг). Поставьте стоимость за 1 единицу. Вычислите «сумму» по формуле = «количество»\* «цена». Вычислите «итого».

Магазин «1000 мелочей»  
адрес \_\_\_\_\_

### ТОВАРНЫЙ ЧЕК

Наименование	Количество	Цена	Сумма
Итого:			

Сумма прописью \_\_\_\_\_

« \_\_\_\_ » \_\_\_\_\_ 20 \_\_\_\_ г.

Продавец \_\_\_\_\_

## Лист 2 «Салат»

Для столбцов «цена», «сумма» установите денежный формат. Посчитайте «сумма». Снизу таблицы добавьте строку «общая стоимость салата» и вычислите ее. Сверху таблицы добавьте заголовок САЛАТ «ОЛИВЬЕ»

наименование	количество	цена	сумма
колбаса	200 гр.	180	
яйцо	3 шт.	28	
зеленый горошек	1 б.	30	
картофель	4 шт.	17	
майонез	1 п.	20	
огурец	1 шт.	48	

## Лист 3 «Расход»

Площадь одного подоконника 1 м<sup>2</sup>, одной двери 2 м<sup>2</sup>. Подсчитайте количество дверей и подоконников в вашей квартире и заполните столбцы «площадь». Посчитайте «расход» по формуле = «кг»/10\* «площадь». Столбец «расход» через формат ячеек сделайте в кг. Сверху таблицы введите заголовок:

### РАСХОД МАТЕРИАЛОВ ДЛЯ ОКРАСКИ

Материал	Поверхность					
	Двери			Подоконники		
	кг на 10 м <sup>2</sup>	Площадь	Расход	кг на 10 м <sup>2</sup>	Площадь	Расход
Олифа	7,6			6,6		
Белила тертые	6,0			6,5		
Пигмент	1,5			0,6		

## Практическое занятие № 13

Тема: Абсолютная, относительная и смешанная адресации.

Цель работы: научить вводить формулы с различной адресацией.

Необходимые для выполнения работы знания:

Абсолютная и относительная адресация. Смешанная адресация.

Сведения из теории:

1) Абсолютная адресация указывает на ячейку и имеет вид  $\$A\$1$ , зафиксированы как столбец, так и строка. 2) Смешанная адресация предназначена для фиксации одного из параметров либо столбца либо строки, например ( $\$A1$ ,  $A\$1$ ). 3) Относительная адресация имеет вид  $A1$ .

План работы:

Переименовать листы согласно образцу. Ввести оформление получившихся таблиц.

Пересчет делать с помощью ввода формул.

### Лист 1 «Таблица умножения»

Ввести формулу в ячейке на пересечении цифр 1 и 1, расставьте смешанную адресацию.

**ТАБЛИЦА УМНОЖЕНИЯ**

	1	2	3	4	5	6	7	8	9	10
1	1	2	3	4	5	6	7	8	9	10
2	2	4	6	8	10	12	14	16	18	20
3	3	6	9	12	15	18	21	24	27	30
4	4	8	12	16	20	24	28	32	36	40
5	5	10	15	20	25	30	35	40	45	50
6	6	12	18	24	30	36	42	48	54	60
7	7	14	21	28	35	42	49	56	63	70
8	8	16	24	32	40	48	56	64	72	80
9	9	18	27	36	45	54	63	72	81	90
10	10	20	30	40	50	60	70	80	90	100



### Лист 2 «Мороженное»

Дана стоимость одной порции мороженого, подсчитайте стоимость указанного числа порций. Установите в ячейках денежный формат.

#### ОПРЕДЕЛЕНИЕ СТОИМОСТИ НЕСКОЛЬКИХ ПОРЦИЙ МОРОЖЕНОГО

	1	2	3	4	5	6
Рожок	28					
Эскимо	22					
Семейное	120					
Батончик	28					
В стаканчике	47					
С вафлями	55					
Торт - мороженое	160					

### Лист 3 Курс»

Пересчитайте курс доллара относительно рубля, поставьте формат в рублях. По строке перечислено количество купюр. В выделенной ячейке курс рубля относительно доллара. К таблице добавьте заголовок «Курс». Поставьте границы в таблицу.

руб \ \$	1	2	3	4	5	6	7	8	9
10									
20									
30									
40									
50									
28,30									

## Практическое занятие № 14

Тема: работа с функциями *если, округл, слчис*.

Цель работы: научить пользоваться функциями, *если, округл, слчис*.

Необходимые для выполнения работы знания:

Создание формул с применением функций (*если, округл, слчис*). Принцип работы функций (*если, округл, слчис*).

Сведения из теории:

Для того чтобы воспользоваться функцией надо: вставка – функция, вызвать мастер функций, выбрать определённую категорию, нужную функцию, прочитать разъяснения к функции, согласится с выполнением задачи. Ввести определённые ячейки. Пользуясь разъяснением к функции ввести определенные значения

ЕСЛИ (лог. усл; истина; ложь)

ОКРУГЛ(число)

СЛЧИС()

План работы:

Переименовать листы согласно образцу. Ввести обрамление получившихся таблиц. Пересчёт делать с помощью ввода формул.

### Лист 1 «Математика»

Составить таблицу из следующих колонок: А – ФИО (список из 8 студентов), В – количество пропусков, С – оценка, D – запись в зачётке. Заполнить первые три. Если количество пропусков < 20, то в последней колонке ставится «зачёт», иначе – «незачёт». Столбик E назвать «Аттестация». Если оценка больше или равна 4 и если количество пропусков меньше или равно 20, то ставится зачёт. В противном случае – незачёт.

### Лист 2 «Прогноз погоды»

Составить таблицу для обработки параметров температуры в течении месяца, Столбец А назвать «Дата» и ввести числа от 1 до 31, В – «Температура» и вычисляем её по формуле  $= (30 + \text{СЛЧИС}()) * 15$ . Значения температуры округлить до целых и сделать их отрицательными. Столбец С назвать «Давление» и рассчитать его по формуле  $= 750 + \text{СЛЧИС}() * 20$ , округлив значение до целых. Столбец D назвать «Облачность» и задать условие: если давление > 760 – ясно, иначе – пасмурно. Перед столбцом А ввести столбец с названиями дней недели.



	A	B	C	D	E	F
1		1996	1997	1998	1999	2000
2	Доходы	\$ 97 000,00	\$ 102 000,00	\$ 107 000,00	\$ 119 000,00	\$ 128 000,00
3	Расходы	\$ 42 000,00	\$ 50 000,00	\$ 52 000,00	\$ 63 000,00	\$ 67 000,00
4	Прибыль	\$ 55 000,00	\$ 52 000,00	\$ 55 000,00	\$ 56 000,00	\$ 61 000,00
5	Прирост за год		\$ -3 000,00	\$ 3 000,00	\$ 1 000,00	\$ 5 000,00

### Лист 3 «Транспорт»

Подготовьте таблицу для расчета еженедельных трат на поездки в городском транспорте. В соответствующие ячейки вводится число поездок на каждом виде транспорта в определенный день недели. В отдельную ячейку вводится стоимость одной поездки на текущий момент (эта стоимость одинакова для любого вида транспорта).

	понедельник	вторник	среда	четверг	пятница	суббота	воскресенье
<i>Метро</i>							
<i>Автобус</i>							
<i>Троллейбус</i>							
<i>Трамвай</i>							
<i>Стоимость одной поездки:</i>				<i>Всего за неделю:</i>			

## Практическое занятие № 16

Тема: Условное форматирование. Работа с расширенным фильтром.

Цель работы: научить добавлять к ячейкам примечания и производить условное форматирование.

Необходимые для выполнения работы знания:

Применение условного форматирования для данных. Применять фильтр.

Сведения из теории:

Для того чтобы задать условное форматирование необходимо выделить определенные ячейки, вызвать Формат – Условное форматирование, набрать условие, и обозначение истинности форматирования. Для того чтобы задать автофильтр необходимо Выделить необходимые для фильтрации значения, выбрать Данные – Фильтр – Автофильтр, на ячейки появится обозначение меню, открыть меню ячейки, выбрать одно из пунктов (все, первые 10 или условие), если выбрали условие появляется диалоговое окно в котором надо выбрать выполнением задачи «ок» в документе скроются те строки которые будут отсеяны в процессе фильтрации, чтобы вернуть выбираем меню ячейки, пункт «все».

План работы:

Переименовать листы согласно образцу. Ввести обрамление получившихся таблиц. Пересчет делать с помощью ввода формул.

### Лист 1 «Бытовики»

Составить список работников с зарплатой. Названия столбцов: А – ФИО; В – должность; С – стаж; D – оклад. Вставить между первым и вторым столбиком столбец для возраста. вставить пустую строку вверху и добавить заголовок "Список сотрудников". К ячейке ОКЛАД добавить примечание «минимальная заработная плата».

ФИО	возраст	должность	стаж	оклад
Иванов	49	Сантехник	28	20 000
Петров	48	Сварщик	30	21 000
Сидоров	12	Вахтер	2	12 000
Ковалев	14	Слесарь	4	10 000

С помощью условного форматирования задать: для оклада >15000 р. - синий цвет, для возраста <16 и >45 лет -зеленый цвет, для стажа >25 лет - оранжевый цвет заливки.

### Лист 2 «Математика».

Составить таблицу из следующих колонок: А – ФИО (список из 8 студентов), В – количество пропусков, С – оценка, D – запись в зачетке. Заполнить пер-

вые три. Если количество пропусков  $<20$ , то в последней колонке ставится “зачет”, иначе - ”незачет”. Столбик Е назвать “Аттестация”. Если оценка больше или равна 4 и если количество пропусков меньше или равно 20, то ставится зачет. В противном случае – незачет. К столбцу «Аттестация» применить условное форматирование (Если зачет – ячейка заливается зеленым цветом, если незачет - ячейка заливается синим цветом). Добавить оглавление по названию листа.

### **Лист 3 «Прогноз погоды»**

Составить таблицу для обработки параметров температуры в течении месяца. Столбец А назвать “Дата” и ввести числа от 1 до 31, В – “Температура” и вычисляем ее по формуле  $= (30 + \text{СЛЧИС}()) * 15$ . Значения температуры округлить до целых и сделать их отрицательными. Столбец С назвать “Давление” и рассчитать его по формуле  $= 750 + \text{СЛЧИС}() * 20$ , округлив значение до целых. Столбец D назвать ”Облачность” и задать условие: если давление  $>760$  – ясно, иначе – пасмурно. Перед столбцом А ввести столбец с названиями дней недели. Переименовать лист в «Прогноз погоды». К столбцу «Облачность» применить условное форматирование (Если ясно – цвет текста красный, если пасмурно – цвет текста синий). Добавить оглавление по названию листа.

## Практическое занятие № 17

Тема: Работа с автофильтром. Работа с расширенным фильтром.

Цель работы: научить работать с автофильтром.

Необходимые для выполнения работы знания:  
применение автофильтра.

Сведения из теории:

Для того чтобы задать автофильтр необходимо выделить необходимые для фильтрации значения, выбрать данные – фильтр – автофильтр, на ячейки появится обозначение меню, открыть меню ячейки, выбрать одно из пунктов (все, первые 1- или условие), если выбрали условие появляется диалоговое окно в котором надо выбрать определенное условие по которому будет проходить фильтрация данных, согласится с выполнением задачи «ок» в документе скроются те строки которые будут отсеяны в процессе фильтрации, чтобы вернуть выбираем меню ячейки, пункт «все».

План работы:

Переименовать листы согласно образцу. Вести оформление получившихся таблиц. Пересчет делать с помощью ввода формул.

**Лист 1 «Гараж»**

Создать таблицу “Гараж”. Отфильтровать данные: имена должны начинаться с буквы А или К, оставить в таблице только водителей, а последний столбец отфильтровать по возрасту меньше 30 лет.

<b>Ф.И.О.</b>	<b>Должность</b>	<b>Возраст</b>
Кравченко В. Ю.	водитель	22
Антонов К. Б.	слесарь	41
Забусов К. Р.	водитель	28
Вакуленко А. Г.	электрик	25
Кулешов Р. Е.	электрик	22
Алексеев П. Д.	водитель	36
Чикин А. Н.	слесарь	21
Кузнецов В. В.	водитель	39
Дуров Е. К.	водитель	32
Крохалев С. М.	водитель	31
Знамцев М. Ю.	слесарь	35
Бессмертный С. Г.	электрик	21
Курилов И. Л.	электрик	25
Климов С. А.	водитель	30
Наговицын В. П.	водитель	25

Лист 2 «Студент»

<i>Студент</i>	Экзамен 1	Экзамен 2	Экзамен 3	Экзамен 4	<i>Среднее</i>
Азимов А.	87	90	79	96	
Булавин К.	92	94	94	97	
Гаврилов С.	96	95	95	80	
Пашков Ю.	85	87	87	88	
Рейн В.	81	88	88	85	

**Задание:** в данной таблице посчитать среднее значение баллов за четыре экзамена (результат округлить до целых значений), выполнить форматирование строки заголовков, столбца с фамилиями студентов и задать оформление таблицы. Справа от “Среднее”, задать бонус и, высчитать итог.

<i>Бонус</i>	<i>Итог</i>
2	
3	
4	
2	
3	



## Практическое занятие № 18

Тема: Мастер диаграмм.

Цель работы: составлять диаграмму с помощью мастера диаграмм. Форматирование диаграммы.

Необходимые для выполнения работы знания:

Переименовать листы согласно образцу. Ввести обрамление получившихся таблиц. Пересчет делать с помощью вводы формул.

Сведения из теории:

Для того чтобы составить диаграмму необходимо выделить необходимые данные, выбрать Вставка - Диаграмма, появится диалоговое окно мастера диаграмм, диаграмма создается в четыре этапа: на первом этапе выбирается тип и вид диаграммы, на втором создаются диапазоны данных и ряды для диаграммы, на третьем заголовки, легенда, подписи данных, на четвертом определяется в какое место должна встать диаграмма.

План работы:

Переименовать листы согласно образцу. Ввести обрамление получившихся таблиц. Пересчет делать с помощью ввода формул.

### Лист 1 «Успеваемость»

Фамилия студента	математика	физика	химия	геометрия	литература	история	ин. язык	БЖД	физкультура	Количество отличных оценок
Лосев Иван	4	3	4	5	5	3	3	4	5	
Коржаков Петр	4	3	5	5	5	3	5	5	5	
Юров Василий	3	4	3	5	4	3	4	5	4	
Минина Юлия	3	4	3	4	3	4	3	5	4	
Красина Ольга	3	5	5	3	3	4	5	3	4	
Викун Юрий	4	3	3	5	3	5	4	3	4	
Березин Михаил	5	4	4	5	4	4	5	4	4	
Серова Лидия	4	3	5	3	4	3	4	3	5	

❖ С помощью функции СЧЕТЕСЛИ подсчитать количество пятерок у каждого студента и выделить фамилии тех студентов, у которых число отличных оценок превышает 4.

❖ Для Лосева Ивана и Березина Михаила построить гистограмму для сравнения уровня успеваемости по всем предметам.

### Лист 2 «Экзамен»

экзамен студент	1	2	3	4	среднее
Азимов	87	67	65	78	
Булавин	92	45	89	98	
Перцев	96	56	97	89	
Гаврилов	76	99	68	99	

Вычислите «среднее» и по этому столбцу постройте круговую объемную диаграмму с подписями данных. С помощью условного форматирования выделите по среднему баллу лучшего студента.

## Практическое занятие № 19

Тема: Ссылки на ячейки другого листа.

Цель работы: научить работать со ссылками с листа на лист.

Необходимые для выполнения работы знания:

Создание ссылок с листа на лист. Адресации.

Сведения из теории:

Для того чтобы создать ссылку с листа на лист необходимо на одном из листов поставить знак равно для создания формулы перейти с помощью меню листов на другой лист, выделить на нем нужную ячейку поставить математический знак (/ \* - +), перейти с помощью меню листов на следующий лист выбрать следующую ячейку и нажать Enter. При выборе ячеек по необходимости ставит различные типы адресаций (относительная, абсолютная, смешанная) с помощью символа (\$).

План работы:

Переименовать листы согласно образцу. Ввести оформление получившихся таблиц. Пересчет делать с помощью ввода формул

**Лист1** «Справочная»

### Справочная информация

Районный коэффициент	1,8
Пенсионный налог	1%
Подходный налог	12%

### Повышающий коэффициент

Январь	Февраль	Март
1,2	1,3	1,4

По таблице «Повышающий коэффициент» постройте произвольную диаграмму с подписями данных (проценты).

**Лист2** «Январь»

Задание: построить таблицы расчета заработной платы для 3-х месяцев. Для каждого месяца расчет вести на отдельном листе (соответственно его переименовав) с учетом повышающего коэффициента. Значения районного коэффициента, подоходного налога и повышающего коэффициента взять из справочных таблиц, которые будут на отдельном листе (две). Для Леонова С. Ф. в феврале увеличить северный коэффициент до 0,5, а для Лисицыной С. Е. в марте до 0,6. Отформатировать таблицы.

### Расчет заработной платы за январь

ФИО	Оклад	Оклад с повышающим коэффициентом	Северный коэффициент	Всего начислено	Пенсионный налог (сумма)	Подходный налог (сумма)	На руки
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
Иванов И. И.	450 р.		0,8				
Петров И. П.	500 р.		0,7				
Сидоров А. А.	600 р.		0,8				
Мишин С. П.	450 р.		0,5				
Чумаков Г. А.	580 р.		0,3				
Леонов С. Ф.	470 р.		0,4				
Ушакова М. А.	400 р.		0,6				
Лисицына С. Е.	520 р.		0,5				
Андреева М. Т.	630 р.		0,7				
Алексеев И. А.	580 р.		0,8				

Заполните таблицу по формулам:

(3)=(2)\*повышающий коэффициент;

(5)=(3)+(2)\*(4)+(2)\*районный коэффициент;

(6)=(5)\*пенсионный налог;

(7)=(5)\*подходный налог;

(8)=(5)-(6)-(7).

## Практическое занятие № 20

Тема: Повторение Функция, Адресация.

Цель работы: повторить ранее изученные функции, смешанную адресацию.

Необходимые для выполнения работы знания:

Применение функций для решения задач. Применение адресаций для ячеек.

Сведения из теории:

Абсолютная адресация ставится для фиксации определенного значения с помощью значка (\$), например (\$A\$1) - зафиксированы как столбец, так и строка. Смешанная адресация предназначена для фиксации одного из параметров либо столбца, либо строки, например (\$A1, A\$1). Относительная адресация. Для того чтобы воспользоваться функцией надо: вставка-функция, выбрать нужную функцию, согласится с выполнением задачи. Пользуясь разъяснением к функции ввести определенные значения.

План работы:

Переименовать листы согласно образцу. Ввести обрамление получившихся таблиц. Пересчет делать с помощью ввода формул.

**Лист 1** «Плита»

Посчитайте «Расход кВт/ч» = февраль - январь:  
посчитайте «Сумма»=«Расход»\*«Стоимость». В столбце «Сумма» установить денежный формат. Снизу таблицы добавить строку «итого» и посчитайте, сколько денег было истрачено за год.

При наличии кухонной электроплиты 125 р/кВт ч  
При наличии кухонной газовой плиты 180 р/кВт ч

Месяц	Показание счетчика	Расход кВт/ч	Сумма	
			с электроплитой	с газовой плитой
	3750			
Январь	3840	90		
Февраль	3960			
Март	4070			
Апрель	4200			
Май	4312			
Июнь	7397			
Июль	4502			
Август	4607			
Сентябрь	4667			
Октябрь	4780			
Ноябрь	4899			
Декабрь	4972			

### *Лист 3 «Таблица»*

(«столбец»\*10+«строка»)\*(«столбец»\*10+«строка»); не забудьте про смешанную адресацию

Добавьте к таблице заголовок **ТАБЛИЦА КВАДРАТОВ**

	0	1	2	3	4	5	6	7	8	9
1	100	121	144	169	196	225	256	289	324	361
2	400	441	484	529	576	625	676	729	784	841
3	900	961	1024	1089	1156	1225	1296	1369	1444	1521
4	1600	1681	1764	1849	1936	2025	2116	2209	2304	2401
5	2500	2601	2704	2809	2916	3025	3136	3249	3364	3481
6	3600	3721	3844	3969	4096	4225	4356	4489	4624	4761
7	4900	5041	5184	5329	5476	5625	5776	5929	6084	6241
8	6400	6561	6724	6889	7056	7225	7396	7569	7744	7921
9	8100	8281	6464	8649	8836	9016	9216	9409	9604	9801

### **Практическое занятие № 21**

Тема: Создание презентации с использованием готовых шаблонов, подбор иллюстративного материала, создание текста слайда.

#### Цель работы:

- 1 Научиться создавать презентацию средствами PowerPoint;
- 2 Научиться форматировать и редактировать слайды.

#### Необходимые для выполнения работы знания:

Разработка презентации. Применение шаблонов дизайна.

В качестве темы первой презентации возьмем электронную иллюстрацию выступления, касающегося структуры построения курса лекций по изучению Microsoft Office.

Этот процесс подготовки презентации необходимо разбить на два этапа:

1. непосредственная разработка презентации, т. е. оформление каждого слайда;
2. демонстрация, т. е. процесс показа готовых слайдов, который может сопровождаться пояснениями лектора, некоторыми графическими пометками по ходу демонстрации.

Для начала нужно сформулировать тему будущей презентации, хотя бы ориентировочно определить количество слайдов и их структуру. Продумать варианты оформления слайдов.

#### План работы:

Подготовим шесть слайдов. На первом отразим название курса и его про-

должительность (титульный лист презентации).

На втором — графически отобразим структуру курса. На остальных — содержание занятий, соответственно по темам:

Microsoft Word;

Microsoft Excel;

Microsoft PowerPoint 2003;

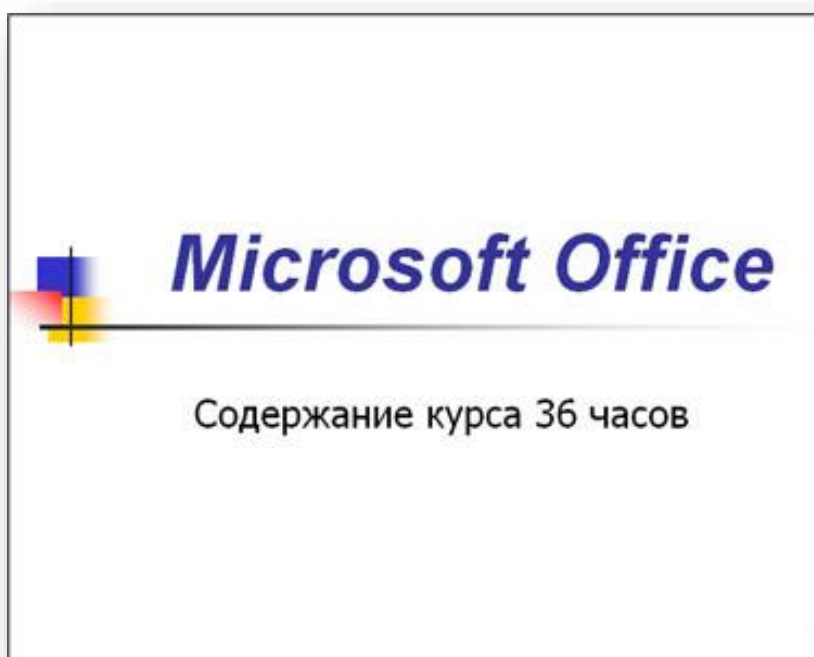
Организация работы с документацией.

Создание презентации

Слайд № 1

Запустите PowerPoint. Пуск-Программы - PowerPoint.

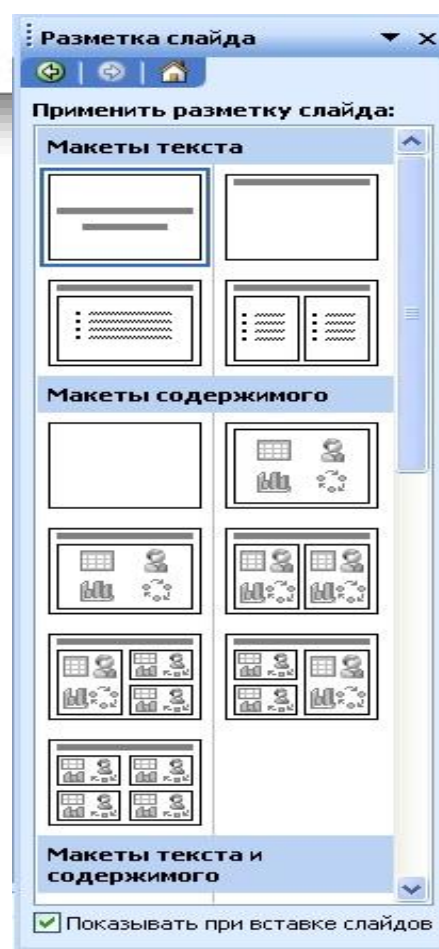
В группе полей выбора **создать новую презентацию** выберите **новая презентация**.



Следующим шагом окажется появление окна **Разметка слайда**, на котором представлены различные варианты разметки слайдов (рисунок 27.1). Выберите самый первый тип — **Титульный слайд**.

Наконец-то перед вами появится первый слайд с разметкой для ввода текста (метками-заполнителями).

**Метки-заполнители** — это рамки с пунктирным контуром, появляющиеся при создании нового слайда. Эти рамки служат метками-заполнителями для таких объектов, как заголовок слайда, текст, диаграммы, таблицы, организационные диаграммы и графика. Чтобы добавить текст в метку-



заполнитель, достаточно щелкнуть мышью, а чтобы добавить заданный объект, щелкнуть дважды. Однако белый фон не производит впечатления.

Начните свою работу с выбора цветового оформления слайда. PowerPoint предоставляет возможность воспользоваться шаблонами дизайна которые позволяют создавать презентации в определенном стиле.

Шаблон дизайна содержит цветовые схемы, образцы слайдов и заголовков с настраиваемыми форматами и стилизованные шрифты. После применения шаблона дизайна каждый вновь добавляемый слайд оформляется в едином стиле.

Рисунок 27.1 – Разметка слайда

В меню **Формат** выберите команду **Применить шаблон дизайна (Оформление слайда)** ... и дальше вас ждет очень приятный процесс - «просматривай и выбирай».

Когда разметка выбрана, остается ввести с клавиатуры текст заголовка и подзаголовка. Для этого достаточно щелкнуть мышью по метке-заполнителю, и ввести текст, который автоматически будет оформлен в соответствии с установками выбранного шаблона дизайна.

Первый слайд готов.

### Слайд № 2


Самый сложный по изготовлению и насыщенный слайд. К его подготовке мы приступим в самую последнюю очередь. Сразу же после разработки первого слайда, приступим к третьему.

### Слайд № 3

Для того чтобы вставить новый слайд, выполните команду **Вставка–Создать слайд...** Появляется уже знакомое окно **Создать слайд**. Выберите разметку слайда **Заголовок** и текст в две колонки.

- Щелчок мыши по метке-заполнителю заголовка позволяет ввести новый заголовок.
- Щелчок мыши в метке-заполнителе левой колонки дает возможность вводить текст.
- Переход к новому абзацу внутри колонки осуществляется при помощи клавиши (Enter).
- Перед вами знакомый по текстовому редактору маркированный список. Размер, цвет и вид маркера определяются параметрами выбранного шаблона дизайна.
- Когда первая колонка будет заполнена текстом, щелкните по метке-заполнителю второй колонки.





## Microsoft Word

- Форматирование абзацев
- Применение к абзацам отступов и интервалов
- Сортировка абзацев
- Форматирование нумерованных и маркированных списков
- Форматирование таблиц
- Применение таблиц к форматированию документов

3

### Слайд № 4

Разрабатывается точно так же, как предыдущий слайд. Выполните эту работу самостоятельно.

- вставьте новый слайд;
- выберите соответствующую разметку;
- введите текст;
- по необходимости располагайте текст в несколько строк;
- равномерно распределите текст по колонкам;
- в случае необходимости переместите метки-заполнители;
- выберите выравнивание текста по своему усмотрению.
- 



## Microsoft Excel

- Выравнивание данных в ячейке
- Заполнение ячеек при помощи маркера заполнения и прогрессии
- Относительные и абсолютные ссылки
- Форматы чисел
- Сортировка данных
- Заполнение таблицы в режиме формы
- Мастер функций
- Организация работы с листами рабочей книги

4

### Слайд № 5

Основным отличием от двух предыдущих слайдов является то, что в окне

**Создать слайд** нужно выбрать разметку **Заголовок и текст**.

Однако в этом варианте применен иерархический (или многоуровневый) список (два уровня абзацев - различные маркеры и отступы).

Для того чтобы "понизить" или "повысить" уровень абзаца примените кнопки панели инструментов. Можете сначала набрать весь текст в один уровень (обычный маркированный список), а затем выделить абзацы следующего уровня и нажать соответствующую кнопку панели инструментов. Маркер автоматически будет изменен при переводе абзаца на новый уровень.

Работая с маркированными списками, будьте особенно внимательны при выделении элементов списка. От этого во многом зависит и результат.

Так как в зависимости от длины строк вводимого текста у вас есть вероятность получить как "широкий", так и "узкий" список, после набора может возникнуть необходимость переместить список целиком, чтобы зрительно он располагался по центру слайда.



Слайд №6

Выполняется точно так же, как и предыдущий слайд.



## Слайд № 2

Теперь приступим к разработке второго слайда - самого сложного по выполнению и самого эффектного.

Так как этот слайд нужно вставить после первого, значит, следует перейти к первому слайду. Перемещение между слайдами осуществляется при помощи кнопок (стрелки), расположенных на вертикальной полосе прокрутки или клавиш **PageUp, PageDown**.

Когда на экране появится первый слайд, выполните команду **Вставка-Создать слайд....**

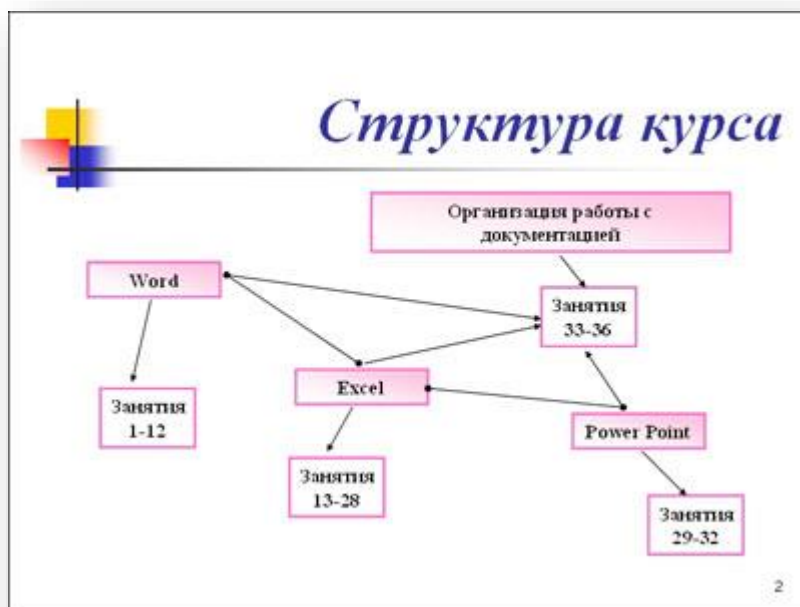
Выберите разметку **Только заголовок**.

Введите текст заголовка.

Далее оформите заголовки разделов курса, размещенные в рамках. Для этого потребуется воспользоваться панелью **Рисование**.

Если панель **Рисование** отсутствует на экране, активизировать ее можно, выполнив команду **Вид-Панели инструментов**. На экране появится диалоговое окно **Панели инструментов**. В списке панелей инструментов активизируйте переключатель **Рисование**.

Панель **Рисование** можно перемещать по экрану мышью, чтобы она не загромождала ту часть слайда, с которой вы собираетесь работать.



Итак, панель **Рисование** активна, выберите на ней инструмент **Надпись**. Указатель мыши принял форму текстового курсора, установите его в нужное место и введите название первого раздела, например, WORD. Вокруг текста возникает рамка (метка-заполнитель) (аналогично той ситуации, когда пользовались инструментом **Рамка текста** в редакторе Word).

Далее воспользуемся графическими возможностями оформления текста.

Выделив набранный заголовок раздела, воспользуйтесь командой **Формат-Шрифт** или соответствующими кнопками для того, чтобы подобрать шрифт, его цвет, размер шрифта и стили оформления (полужирный, курсив);

выровняйте набранный текст По центру рамки **Надпись** (выравнивание происходит по центру рамки, независимо от ее размера);

выполните команду **Формат-Надпись-Цвета и линии...** (рисунок 27.2). В появившемся диалоговом окне выберите опцию **Цвет**, выберите любой понравившейся цвет, вновь войдите в опцию **цвет**, выберите **Способы заливки – Градиентная**. Поэкспериментируйте и выберите понравившейся тип заливки. В меню **Формат-Надписи** в опции **Линии** выберите цвет, шаблон, тип и толщину.

Заголовок первого раздела готов. Три оставшихся оформлены таким же образом. Удобнее всего растиражировать имеющийся (переместить мышью с нажатой клавишей Ctrl, затем в новый заголовок устано

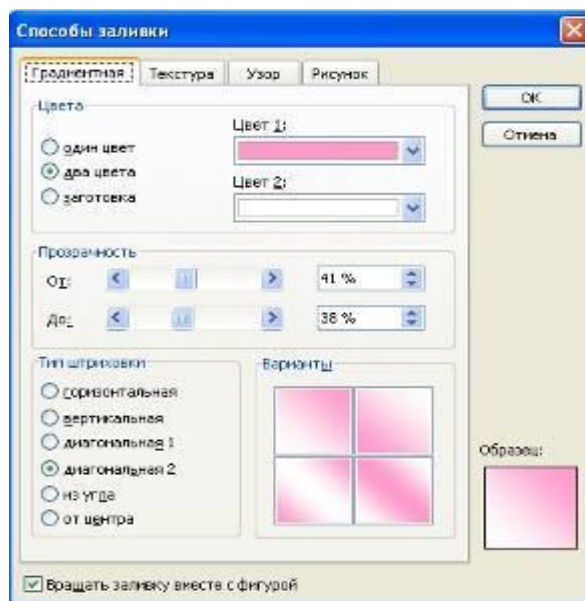


рисунок 2


Рисунок 27.2 – Способы заливки

вить текстовый курсор и изменить текст WORD на EXCEL и т.д. Таким образом,

за короткий промежуток времени можно подготовить заголовки всех разделов.

Перечень занятий подготовлен с применением тех же приемов, только выбрано более простое оформление (отсутствует заливка, выбран только цвет линии). Выполните самостоятельно. Для того чтобы текст внутри рамки располагался в две строки, набирайте его в два абзаца (после первой строки воспользуйтесь клавишей Enter) или в две строки (Shift+Enter).

Разместите на поверхности слайда все элементы таким образом, чтобы их можно было соединить линиями. Далее требуется нарисовать соединяющие линии. На схеме представлены линии двух типов: со стрелкой (соединяют заголовок раздела с перечнем занятий) и ограниченный с двух сторон кругами (соединяют заголовки разделов).

Для начертания линии выберите инструмент **Линия**, проведите линию от одной рамки к другой, затем для выделенной линии установите цвет и вид  стрелки. Нарисованную линию можно перемещать, "схватив" мышью за внутреннюю область. Если же указатель мыши подвести к краю линии, форма указателя изменится. В этом случае линию можно растягивать или сокращать, можно изменить ее наклон.

Пролистайте все имеющиеся слайды.

## Практическое занятие № 22

Тема: Демонстрация слайд-фильма и настройка анимации.

### Цель работы:

1 Научиться демонстрировать слайд-фильм и присваивать эффекты анимации объектов и переходов слайдов.

### Необходимые для выполнения работы знания:


В качестве темы первой презентации возьмем электронную иллюстрацию выступления, касающегося структуры построения курса лекций по изучению Microsoft Office.

Этот процесс подготовки презентации придется разбить на два этапа:

1. непосредственная разработка презентации, т. е. оформление каждого слайда;
2. демонстрация, т. е. процесс показа готовых слайдов, который может сопровождаться пояснениями лектора, некоторыми графическими пометками по ходу демонстрации.

### План работы:

1. Откройте созданный вами ранее файл
2. Демонстрация:

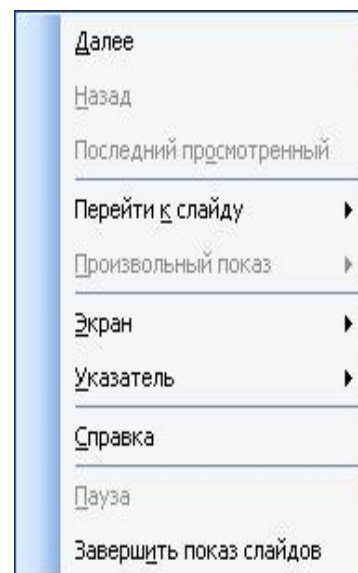
Для того чтобы начать демонстрацию, перейдите к первому слайду и воспользуйтесь кнопкой , расположенной в левой нижней части экрана или выполните команду **Вид-Демонстрация...** (в этом случае нет необходимости переходить к первому слайду).

Первый слайд должен появиться перед вами в режиме просмотра (занимает весь экран).

Переход к следующему слайду в режиме демонстрации осуществляется щелчком мыши, нажатием клавиш **Enter**, **Пробел**, **PageDown**, при помощи клавиш управления курсором **"Вниз"** или **"Вправо"**.

По ходу демонстрации вы можете делать любые устные пояснения, переходя к новому слайду через такой промежуток времени, который потребуется.

Провести демонстрацию наиболее эффективно, вам поможет использование **контекстного меню**, которое в процессе демонстрации можно вызвать **нажатием правой клавиши мыши**.



Как пользоваться командами контекстного меню?

– Вы хотите, чтобы указатель мыши не был виден во время презентации, но собираетесь пользоваться мышью. Выберите команду **Указатель - Параметры стрелки - Скрывать всегда**. Вы сможете перемещать указатель мыши по экрану,

щелкать для перехода к новому слайду, вызывать контекстное меню, при этом указатель мыши останется невидимкой. Для того чтобы вернуть стрелку указателя мыши, вызовите **контекстное меню** и выберите команду **Указатель - Параметры стрелки - Автоматически**.

– Вы хотите на какое-то время отвлечь внимание слушателей от экрана и привлечь к себе, чтобы сообщить что-то особенно важное. Выполните команду **Экран - Черный экран (Белый экран)**. Щелчок мыши выведет на экран следующий слайд.

– Вы хотите во время демонстрации графически отметить некоторые моменты: **обвести какие-то записи, провести указующую линию, чтобы акцентировать на чем-либо внимание слушателей**. При этом вам нужно, чтобы все пометки не сохранялись до следующей демонстрации. Выберите команду **Ручка** указатель мыши изменил форму, теперь им можно рисовать, делать пометки по ходу демонстрации. Цвет ручки можно выбрать при помощи команды **Указатель - Цвет**. Кроме указателя в виде ручки есть еще и Фломастер, и Выделение (маркер).

– По каким-то причинам, вы хотите завершить демонстрацию прежде, чем достигли последнего слайда. Выберите команду **Завершить демонстрацию**.

**Обратите внимание!!!** Обязательно первую демонстрацию проведите только для себя (прорепетируйте), а затем уже выходите выступать перед аудиторией.

Если провести демонстрацию вы хотите не сразу после создания презентации, а через какое-то время (например, на следующий день) нет необходимости запускать PowerPoint. Найдите свой файл при помощи проводника Windows (или заранее создайте для него ярлык на рабочем столе), выделите и вызовите контекстное меню нажатием правой клавиши мыши. В контекстном меню будет присутствовать команда **Демонстрация (Показ)**.

### 3. Настройка анимации.

Откройте презентацию, созданную на прошлом занятии, например, PP\_Иванов.

Щелкните по слайду № 1 правой кнопкой мыши и в контекстном меню выберите опцию **Смена слайдов**. Исследуйте эту панель в правой части окна. Также можно изменить следующие параметры **Смены слайдов** – **Скорость перехода**, а также возможность **Добавить звук**. Возможно задать Смену слайдов по щелчку, либо автоматически, через выбранный интервал времени.

**Обратите внимание!!!** Часто сложно рассчитать время, через которое должна происходить смена слайдов, так как в ходе демонстрации могут возникать различные ситуации, на которые заложить время не представляется возможным. Например, вопросы, которые может задать аудитория во время показа. Поэтому чаще используют смену слайдов по щелчку.

**К вашему сведению!** Слайды, оформленные в одном стиле (цвет, шрифт, переходы, анимация в одном выдержанные направлении), т.е. единого дизайна, легче воспринимаются аудиторией.



Задайте смену слайдов В форме **круга**, **скорость** – **Средне**, **звук** – **Бластер**. И щелкните по кнопке **Применить ко всем**. Если же вам необходимо задать смену слайдов различного вида, то нужно щелкнуть по кнопке **Применить**, тогда выбранный тип смены слайдов будет применен к текущему слайду.


Предварительный просмотр осуществляется при помощи кнопки **Просмотр**. Просмотр всего слайд-фильма производится щелчком по кнопке **Показ слайдов**.

Далее произведем анимацию объектов на слайде. Рассмотрим по шагам процесс анимации слайда.

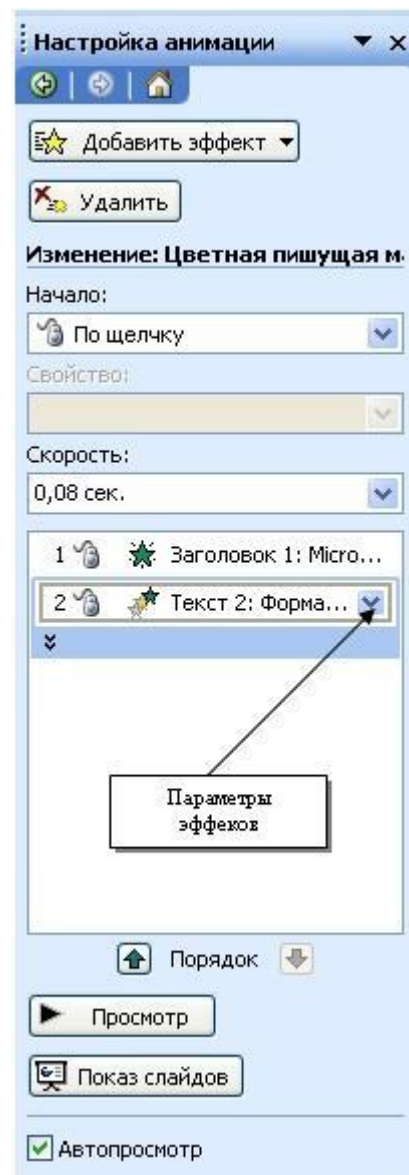
Слайд № 1. щелкните правой кнопкой мыши по заголовку на слайде и в контекстном меню выберем опцию **Настройка анимации**. В правой части экрана появится панель Настройка анимации. На данной панели есть возможность **Добавить эффект: Вход, выделение, выход, пути перемещения**. **Вход** – это объект, у которого анимационный эффект присутствует в момент его появления на слайде. **Выход** – в момент исчезновения, **Выделение** – объект изменяет свои параметры (цвет, размер, и др.). Пути перемещения - присвоение сложной траектории объекту на слайде.

Выберем эффект **Вход – поворот**, **скорость** – **средне**. Подзаголовок эффект **Выделение** – **Изменение цвета текста**, **скорость** – **быстро**.

Слайд № 3 – слайд № 4 – выполните самостоятельно. Обратите внимание на то, что при настройке анимации есть возможность корректировать **параметры эффектов**, а также изменять **порядок** следования объектов анимации.

Слайд № 2 – Здесь необходимо продумать порядок следования объектов при анимации, если вы будете анимировать каждый блок отдельно. Если же вы хотите, чтобы схема появлялась или уходила целиком, как единый объект, то необходимо вспомнить навыки работы с рисованными объектами в Word. На панели инструментов Рисование выберите кнопку **Выбор объектов**  и обрисуйте вокруг всей схемы прямоугольник. **Обратите внимание!!!** Все объекты должны попасть в зону выделения. Далее на панели Рисование щелчок по кнопке **Действия – Группировать**. Теперь схема – это один объект, настройте для него анимацию.

4 Сохраните результат своей работы в папке своей группы (класса) под тем же именем, что и ранее.





## Практическое занятие № 23

Тема: Запись изображений и звука с использованием различных устройств.

### Цель работы:

1. Научиться создавать в слайде графический объект, редактировать и форматировать его;
2. Уметь размещать графические объекты на слайде;
3. Сформировать навыки использования анимации при показе слайдов.

### Необходимые для выполнения работы знания:


Добавление графических объектов на слайд.

### План работы:

#### **Технология выполнения упражнения:**

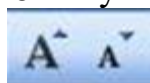
1. Откройте созданный ранее вами файл презентации **Распределение финансов.**

2. Откройте титульный слайд. Создайте на нем блок с датой, для этого используйте на панели **Рисование** кнопку

**Надпись** , присвойте ей следующие параметры (предварительно проверьте, что блок надпись активна) **цвет линии - нет линий, цвет заливки - нет заливки.** В текстовый блок вставим дату:

**Вставка-Дата и время.** Выберите любой доступный формат и поставьте флажок у опции **Обновлять автоматически.** Данная опция позволяет не менять ручную дату и время, а приложение PowerPoint будет производить эту функцию автоматически. Выделите блок с **Датой** и уменьшите размер шрифта до 18 пт. Сделать

это можно с помощью кнопок



Переместите блок с датой в верхний левый угол изменив предварительно его размер.

3. Выполните команду Вставка – Рисунок – Картинки, если вы хотите взять рисунок из коллекции Microsoft или **Вставка – Рисунок – Из файла**, если ваш рисунок находится в какой то из папок компьютера.



4. Выберите подходящую картинку и вставьте её на слайд. Картинка будет располагаться точно по центру слайда. Переместите картинку в другое место слайда, при необходимости уменьшите или увеличьте её размеры, используя маркеры окаймления. Кроме статичных рисунков в коллекции файлов есть и анимационные, они обозначены звездочкой внизу картинки.
5. Примените фон для **Титульного слайда**. По известному вам способу.



Анимация объектов при демонстрации слайдов

**Технология выполнения упражнения:**

1. Настройте анимацию для титульного слайда презентации.
  2. Перейдите к титульному слайду.
  3. Присвойте всем объектам вашей презентации анимацию объектов. Выполните команду **Показ слайдов – Настройка анимации**. В левой части можно проконтролировать – тот ли объект был выбран.
  4. Затем необходимо произвести отдельную настройку анимации каждого выбранного объекта: эффекта появления объекта, времени его появления и способа его появления (по щелчку мыши или по времени). Для просмотра результатов настройки нажимайте кнопку **Просмотр** окна Настройки анимации. Результат будет виден на образце слайда.
  5. После настройки всех объектов щелкните по кнопке ОК.
  6. К каждому слайду можно добавить звук (текст автора, записанный ранее; звуки, музыку). Используя команду **Вставка - Фильмы и звук – Звук из коллекции картинок (Стандартные)**.
  7. Прослушайте свой результат.
  8. Создайте **слайд № 4. Разметка слайда – Пустой**. Создайте надпись с помощью функции **Добавить объект Word Art – СПАСИБО ЗА ВНИМАНИЕ!** Присвойте эффект анимации **Выход – Центрифуга**.
  9. Сохраните презентацию на жестком диске вашего компьютера в папке своей группы командой **Файл – Сохранить** под тем же именем **Распределение финансов**. Обратите внимание, что расширение файла созданного в PowerPoint - **.ppt**
- Секреты настройки воспроизведения звукового файла**
- Часто мы сталкиваемся с проблемой воспроизведения звукового файла на протяжении нескольких слайдов или демонстрации всей презентации. Как присвоить звук на все слайды презентации PowerPoint?

1. Вставляете звук.
2. По правому клику мыши по грамофону заходите в область задач "Настройка анимации".
3. По правому клику мыши на эффекте проигрывания звука выбираете пункт "Параметры эффектов".

4. блоке "Закончить" выбираете "Закончить после X слайда" и выставляете нужный слайд.

А еще можно в свойствах звука поставить непрерывное воспроизведение.

## **Практическое занятие № 24**

Тема: Творческое задание.

### Цель работы:

создавать и продемонстрировать презентацию.

### Необходимые для выполнения работы знания:

При помощи приложения PowerPoint создать домашнюю картинную галерею. Для удобства перемещения по галерее добавить управляющие кнопки и включить звуковое сопровождение демонстрации.

Для удобства работы некоторые виды презентаций можно снабдить управляющими кнопками. В процессе демонстрации они позволят осуществить возврат к предыдущему слайду, переход к следующему слайду, вызов дополнительных функций.

### План работы:

#### **Творческое задание.**

#### **Темы проектов**

- 1 Роль компьютерных технологий в развитии средств мировых коммуникаций.
- 2 История внедрения и перспективы применения компьютерных технологий в современной медицинской науке и практике.
- 3 Искусственный интеллект: его возможности и потенциал.
- 4 Влияние ПК на здоровье человека.
- 5 История развития вычислительной техники.
- 6 Какое влияние оказывает работа с ПК на костно-мышечный аппарат учащихся.
- 7 Этические нормы поведения в информационной сети.
- 8 Преимущества и недостатки работы с ноутбуком, нетбуком, карманным компьютером.
- 9 Негативное воздействие компьютера на здоровье человека и способы защиты.
- 10 Значение компьютерных технологий в жизни современного человека.
- 11 Информационные технологии в системе современного образования.
- 12 Современные технологии и их возможности.
- 13 Сканирование и системы, обеспечивающие распознавание символов.
- 14 Всемирная сеть Интернет: доступы к сети и основные каналы связи.
- 15 Основные принципы функционирования сети Интернет.
- 16 Разновидности поисковых систем в Интернете.
- 17 Программы, разработанные для работы с электронной почтой.
- 18 Беспроводной Интернет: особенности его функционирования.

- 19 Система защиты информации в Интернете.
  - 20 Мертвые языки программирования.
  - 21 Метод (алгоритм) шинглов.
  - 22 Моделирование гармонических колебаний в среде табличного процессора MS Excel.
  - 23 Нейрокомпьютеры и их применение
  - 24 Обработка информации с применением генетических алгоритмов, муравьиных алгоритмов, нейронных сетей, ориентированных и неориентированных графов.
  - 25 Определение числового кода символа и ввод символа по числовому коду в текстовых редакторах.
- Применение информационных технологий в различных сферах деятельности (образовании, горной промышленности, нефтепереработке и пр.).
- 26 Применение современных моделей автоматизации (математическое моделирование, процессное моделирование, нейронные сети, метод графов и пр.).
  - 27 Проектирование с применением диаграмм процессов
  - 28 Развитие операционных систем для локальных сетей.
  - 29 Развитие технологий соединения компьютеров в локальные сети.
  - 30 Разработка и внедрение on-line игр в образовательный процесс.
  - 31 Растровые и векторные редакторы. обработка фотографий в Adobe Photoshop.
  - 32 Создание изображений в векторном редакторе Corel Draw.
  - 33 Создание изображений в векторном редакторе, входящем в состав текстового редактора Word.
  - 34 Создание тематического Web-сайта.
  - 35 Сортировка массивов. Разработка нового метода сортировки.
  - 36 Таксономия (Классификация) Флинна.
  - 37 Шифрование с использованием закрытого ключа.
  - 38 Эпоха «Smart». Проблемы, особенности, перспективы развития.

Ваша работа должна состоять из 8 слайдов. Создайте эти слайды, применив шаблон разметки для 1 слайда – Титульный, для последующих – Пустой слайд. Для каждого из этих объектов задаются: **фон, текст, рисунки (изображения), кнопки и переход слайда.**

Продемонстрируйте преподавателю вашу презентацию.

## Практическое занятие № 25

Тема: Основные тэги HTML. Форматирование шрифта.

Цель работы: научиться использовать основные теги и форматировать шрифт.

Необходимые для выполнения работы знания:

- 1 различные виды технологий web -программирования
- 2 основные правила работы браузера Internet Explorer
- 3 основные понятия web-программирования (тэги, контейнера, атрибут)
- 4 основные этапы разработки web-сайта
- 5 правила тестирования и публикации web-сайта

Сведения из теории:

**Структура Web-страницы.** Большая часть тэгов образует контейнер, состоящий из открывающего и закрывающего тэгов. Тэги можно набирать как заглавными, так и строчными буквами.

Web-страница помещается в контейнер `<HTML></HTML>` и состоит из двух частей: заголовка и отображаемого в браузере содержания.

Заголовок страницы помещается в контейнер `<HEAD></HEAD>`. Заголовок содержит название страницы, которое помещается в контейнер `<TITLE></TITLE>` и при просмотре отображается в верхней строке окна браузера.

Также в заголовок помещаются не отображаемые при просмотре мета-тэги, задающие кодировку страницы для ее правильного отображения в браузере, а также содержащие описание и ключевые слова страницы, которые в первую очередь просматривают роботы поисковых систем.

Отображаемое в браузере содержание страницы помещается в контейнер `<BODY></BODY>`.

План работы:

Создать Web-страницу, знакомящую с основными тэгами HTML.

1. Запустить текстовый редактор Блокнот командой [Пуск - Программы - Стандартные - Блокнот].

2. Ввести HTML-код, задающий структуру Web-страницы:

```
<HTML>
<HEAD>
<TITLE>Первое знакомство с тэгами HTML</TITLE>
</HEAD>
<BODY>
  <BODY>
</HTML>
```

3. Ввести команду [Файл - Сохранить]. Файлу Web-страницы присвоить имя `ваша_фамилия.htm`

4. Запустить браузер и открыть созданный файл командой [Файл - Открыть]. В заголовке окна браузера высвечивается название Web-страницы *Первое знакомство с тэгами HTML*.

5. **Заголовки.** Внести в текст страницы после <BODY> в пустую строку тэги заголовков различных уровней (размеров).

Заголовки различных уровней:

<H1>Заголовок первого уровня</H1>

<H2>Заголовок второго уровня</H2>

<H3>Заголовок третьего уровня</H3>

<H4>Заголовок четвертого уровня</H4>

<H5>Заголовок пятого уровня</H5>

<H6>Заголовок шестого уровня</H6>

6. **Внесение изменений и дополнений в Web-страницу.** В процессе создания Web-страницы приходится добавлять новые тэги и просматривать получаемый результат.

7. **Активизировать Блокнот** с открытой в нем редактируемой Web-страницей. Внести в содержимое страницы необходимые изменения и сохранить новый вариант страницы, выполнив команду меню [Файл - Сохранить].

8. **Активизировать браузер** с открытым в нем предыдущим вариантом страницы. Щелкнуть по кнопке *Обновить*. В окне браузера отобразится обновленная Web-страница

### Форматирование шрифта.

Внести в текст страницы тэги, определяющие начертание шрифта и горизонтальных разделительных линий. Отделить этот фрагмент от остального текста с помощью горизонтальных разделительных линий. Разделительная линия:

1 Запустить текстовый редактор Блокнот командой [Пуск - Программы - Стандартные - Блокнот].

2 Открыть файл ваша\_фамилия.htm

3 Внести в текст страницы тэги, определяющие начертание шрифта и горизонтальных разделительных линий. Отделить этот фрагмент от остального текста с помощью горизонтальных разделительных линий. Разделительная линия:

<HR>

Форматирование шрифта:

<B>Жирный</B>

<I>Курсив</I>

<U>Подчеркнутый</U>

<B><I><U>Жирный подчеркнутый курсив</B></I></U>

<TT>Равноширинный</TT>

Выделение:

<EM>Выделение</EM>

<STRONG>Усиленное выделение</STRONG>

<HR>

4 **Списки.** Внести в текст страницы тэги, задающие списки нумерованные и ненумерованные, а также списки определений.

Нумерованный список:

**<OL>**

**<LI>Первый элемент списка</LI>**  
**<LI>Второй элемент списка</LI>**  
**<LI>Третий элемент списка</LI>**  
**</OL>**

Ненумерованный список:

**<UL>**

**<LI>Первый элемент списка</LI>**  
**<LI>Второй элемент списка</LI>**  
**<LI>Третий элемент списка</LI>**  
**</UL>**

Список определений:

**<DL>**

**<DT>ТЕРМИН 1</DT>**  
**<DD>Пояснение к термину 1</DD>**  
**<DT>ТЕРМИН 2</DT>**  
**<DD>Пояснение к термину 2</DD>**  
**<DT>ТЕРМИН 3</DT>**  
**<DD>Пояснение к термину 3</DD>**  
**</DL>**

5 **Внесение изменений и дополнений в Web-страницу.** В процессе создания Web-страницы приходится добавлять новые тэги и просматривать получаемый результат.

6 **Активизировать Блокнот** с открытой в нем редактируемой Web-страницей. Внести в содержимое страницы необходимые изменения и сохранить новый вариант страницы, выполнив команду меню [Файл - Сохранить].

7 **Активизировать браузер** с открытым в нем предыдущим вариантом страницы. Щелкнуть по кнопке *Обновить*. В окне браузера отобразится обновленная Web-страница.

В итоге в файле имя.htm у вас должно получиться

**<HTML>**

**<HEAD>**

**<TITLE> Первое знакомство с тэгами HTML</TITLE>**

**</HEAD>**

**<BODY>**

**<H1>Заголовок первого уровня</H1>**

**<H2>Заголовок второго уровня</H2>**

**<H3>Заголовок третьего уровня</H3>**

**<H4>Заголовок четвертого уровня</H4>**

**<H5>Заголовок пятого уровня</H5>**

**<H6>Заголовок шестого уровня</H6>**

**<HR>**

**<B>Жирный</B>**

*<I>Курсив</I>*

<U>Подчеркнутый</U>

***Жирный подчеркнутый курсив*****</B></I></U>**

**<TT>Равноширинный</TT>**

**<EM>Выделение</EM>**

**<STRONG>Усиленное выделение</STRONG>**

**<HR>**

**<OL>**

**<LI>Первый элемент списка</LI>**

**<LI>Второй элемент списка</LI>**

**<LI>Третий элемент списка</LI>**

**</OL>**

**<UL>**

**<LI>Первый элемент списка</LI>**

**<LI>Второй элемент списка</LI>**

**<LI>Третий элемент списка</LI>**

**</UL>**

**<DL>**

**<DT>ТЕРМИН 1</DT>**

**<DD>Пояснение к термину 1</DD>**

**<DT>ТЕРМИН 2</DT>**

**<DD>Пояснение к термину 2</DD>**

**<DT>ТЕРМИН 3</DT>**

**<DD>Пояснение к термину 3</DD>**

**</DL>**

**</BODY>**

**</HTML>**



## Практическое занятие № 26

Тема: Цветовые схемы. Шрифты. Вставка изображений.

Цель работы: научиться использовать цветовые схемы и вставлять изображения.

Необходимые для выполнения работы знания:

- 1 различные виды технологий web -программирования
- 2 основные правила работы браузера Internet Explorer
- 3 основные понятия web-программирования (тэги, контейнера, атрибут)
- 4 основные этапы разработки web-сайта
- 5 правила тестирования и публикации web-сайта

Сведения из теории:

**Структура Web-страницы.** Большая часть тэгов образует контейнер, состоящий из открывающего и закрывающего тэгов. Тэги можно набирать как заглавными, так и строчными буквами.

Web-страница помещается в контейнер `<HTML></HTML>` и состоит из двух частей: заголовка и отображаемого в браузере содержания.

Заголовок страницы помещается в контейнер `<HEAD></HEAD>`. Заголовок содержит название страницы, которое помещается в контейнер `<TITLE></TITLE>` и при просмотре отображается в верхней строке окна браузера.

Также в заголовок помещаются не отображаемые при просмотре мета-тэги, задающие кодировку страницы для ее правильного отображения в браузере, а также содержащие описание и ключевые слова страницы, которые в первую очередь просматривают роботы поисковых систем.

Отображаемое в браузере содержание страницы помещается в контейнер `<BODY></BODY>`.

План работы:

- 1 Запустить текстовый редактор Блокнот командой [Пуск - Программы - Стандартные - Блокнот].
- 2 Создать файл **index.htm**

```
<HTML>
<HEAD>
<TITLE>Моя первая страница</TITLE>
</HEAD>
<BODY>...
</BODY>
</HEAD>
</HTML> (Запусти файл в браузере)
```

- 3 Атрибуты задания цветовой схемы (цвета фона, текста и гиперссылок). Цвет на Web-странице задают либо его названием, либо числовым шести разрядным шестна-

дцатеричным кодом #RRGGBB (первые два разряда задают интенсивность красного цвета, вторые – зеленого и третьи – синего).

Значение яркости цвета может меняться от минимальной 00 до максимальной FF. В таблице приведены примеры некоторых цветов:

Цвет	Код	Название		Цвет	Код	Название	
черный	#000000	black	ч	фиолетовый	#FF00FF	magenta	ф
белый	#FFFFFF	white	б	бирюзовый	#00FFFF	cyan	б
красный	#FF0000	red	к	желтый	#FFFF00	yellow	ж
зеленый	#00FF00	lime	з	золотой	#FFD800	gold	з
синий	#0000FF	blue	с	оранжевый	#FFA500	orange	о
серый	#808080	gray	с	коричневый	#A82828	brown	к

Основную цветовую схему Web-страницы можно задать в тэге <BODY> с помощью атрибутов:

Цвет фона	BGCOLOR="#RRGGBB"
Текстура фона	BACKGROUND="file_name"
Цвет текста	TEXT="#RRGGBB"
Цвет текста ссылки	LINK="#RRGGBB"
Цвет текста активной ссылки	ALINK="#RRGGBB"
Цвет текста просмотренной ссылки	VLINK="#RRGGBB"

При использовании текстуры, закрывающей собой всю площадь страницы, применение однотонного фона кажется излишним. Однако рисунки загружаются несколько медленнее, чем текст. Все это время посетители страницы будут видеть цвет фона, заданный атрибутом BGCOLOR. Поэтому для фона указывают цвет, совпадающий с основным тоном фонового рисунка.

Чтобы текст хорошо читался, цвета на странице подбирают контрастирующие по яркости: пастельный фон - темный текст, или темный фон - светлый текст. Нежелательны буквы белого цвета - они могут оказаться невидимыми при печати страницы на принтере.

Для оформления страницы можно использовать следующую цветовую схему: (после <BODY>

```
<BODY      BGCOLOR="#FFFFFF"      BACKGROUND="fon.png"
TEXT="#993300" LINK="#00FF00" ALINK="#FF0000" VLINK="#00FF00">
```

(Сохранить файл и Обновить страничку)

## Вставка изображений.

1 Запустить текстовый редактор Блокнот командой [Пуск - Программы - Стандартные - Блокнот].

2 Открыть файл **index.htm**

3 **Вставка изображений.** Для размещения на Web-страницах используются графические файлы форматов GIF, JPEG и PNG.

4 Изображения помещаются на Web-страницу тэгом IMG с атрибутом SRC, сообщаящим браузеру имя и местоположение графического файла.

```
<IMG SRC="image_name">
```

Если рядом с изображением не должно быть текста, его размещают внутри отдельного абзаца. Выравнивание рисунка по горизонтали в этом случае задают в тэге <P>. Размеры изображения (в пикселах) можно задать с помощью атрибутов WIDTH и HEIGHT. К изображению атрибутом ALT можно добавить название, которое появляется на экран или вместо иллюстрации (если по какой - либо причине графика не выводится в окне браузера), или в качестве всплывающий строки, при указании на рисунок мышью.

Открыть

Вставить в начале страницы картинку.

```
<P ALIGN="center"><IMG SRC="c:\Мои документы\bscap0.jpg"
WIDTH="515" HEIGHT="240" ALT="Просто картинка"></P>
```

Атрибут ALIGN выравнивает текст по верхнему краю, середине или нижнему краю изображения, справа или слева от него с помощью значений: top, bottom, middle, left или right.

**5 Форматирование текста.** Для выделения фрагментов текста используется тэг <FONT>. Атрибут FACE определяет гарнитуру шрифта, атрибут COLOR – цвет и атрибут SIZE - размер символов.

Выравнивание текста по горизонтали задает атрибут ALIGN. Его возможные значения: left - выравнивание по левому краю center – по центру и right - по правому краю.

Выравнивание можно добавлять к тэгам, задающим заголовки, абзацы, или размещающим информацию в таблице. Выровнять по центру заголовков страницы. HSPACE, VSPACE - горизонтальный и вертикальный отступы между картинкой и текстом.

Введем заголовки различных уровней.

```
<H1 ALIGN="center"><FONT COLOR="#FF6600">А это - заголовок
нашей странички.</FONT></H1>
```

```
<H2 ALIGN="center"><FONT COLOR="#FF6622">А это - подзаголовок
нашей странички.</FONT></H2> (Сохранить файл и Обновить страничку)
```

Выделим часть текста более крупным шрифтом и цветом и определим выравнивание

```
<P ALIGN="left"><FONT SIZE="6" COLOR="#FF0066" FACE="Arial" >
Давайте, просто напечатаем простой текст для создания нашей странички,
выровненный по левому краю.</FONT></P>
```

**<P ALIGN="right"><FONT SIZE="5" COLOR="#FF0000" FACE="Times New Roman" > Давайте, просто напечатаем простой текст для создания нашей странички, выровненный по правому краю.</FONT></P>**

**<P ALIGN="center"><FONT SIZE="4" COLOR="#FF0011" FACE="Arial"> Давайте, просто напечатаем простой текст для создания нашей странички, выровненный по центру.</FONT></P> (Сохранить файл и Обновить страничку)**

## Практическое занятие № 27

Тема: Изображение гиперссылок.

Цель работы: научиться оформлять гиперссылки.

Необходимые для выполнения работы знания:

- 1 различные виды технологий web -программирования
- 2 основные правила работы браузера Internet Explorer
- 3 основные понятия web-программирования (тэги, контейнера, атрибут)
- 4 основные этапы разработки web-сайта
- 5 правила тестирования и публикации web-сайта

Сведения из теории:

**Структура Web-страницы.** Большая часть тэгов образует контейнер, состоящий из открывающего и закрывающего тэгов. Тэги можно набирать как заглавными, так и строчными буквами.

Web-страница помещается в контейнер `<HTML></HTML>` и состоит из двух частей: заголовок и отображаемого в браузере содержания.

Заголовок страницы помещается в контейнер `<HEAD></HEAD>`. Заголовок содержит название страницы, которое помещается в контейнер `<TITLE></TITLE>` и при просмотре отображается в верхней строке окна браузера.

Также в заголовок помещаются не отображаемые при просмотре мета-тэги, задающие кодировку страницы для ее правильного отображения в браузере, а также содержащие описание и ключевые слова страницы, которые в первую очередь просматривают роботы поисковых систем.

Отображаемое в браузере содержание страницы помещается в контейнер `<BODY></BODY>`.

Связать Web-страницу с другими документами можно с помощью универсального тэга `<A>`, и его атрибута `HREF`, указывающим в каком файле хранится вызываемый ресурс.

План работы:

- 1 Запустить текстовый редактор Блокнот командой [Пуск - Программы - Стандартные - Блокнот].
- 2 Открыть файл `index.htm`
- 3 Различные виды гиперссылок. Связать Web-страницу с другими документами можно с помощью универсального тэга `<A>`, и его атрибута `HREF`, указывающим в каком файле хранится вызываемый ресурс.

`<A HREF="file_name">Указатель ссылки</A>`

`file_name` – путь к файлу или его URL-адрес в Интернете. Если вызываемый документ размещается в той же папке, что и Web-страница, то можно указывать только имя файла.

Указатель ссылки в окне браузера выделяется подчеркиванием и особым цветом. При указании на него мышью, ее курсор превращается в значок «рука». Щелчок мыши по указателю, вызывает переход на документ, указанный в гиперссылке.

Используем различные значения атрибута HREF для реализации различных реакций браузера:

**<A HREF="index1.htm">Моя вторая страничка</A> (Сохранить файл и Обновить страничку)**

А теперь с оформлением выравнивания и шрифтов Исправляем предыдущую запись.

**<A HREF="index1.htm"><P ALIGN="center"><FONT SIZE="4" COLOR="#FFaa11" FACE="Arial"> Моя вторая страничка</FONT></P></A> (Сохранить файл и Обновить страничку)**

4 Создадим вторую страничку. Для этого откроем **Блокнот** еще раз. И наберем в нем следующий текст.

```
<HTML>
<HEAD>
<TITLE>Моя вторая страница</TITLE>
</HEAD>
<BODY>
<BODY BGCOLOR="#FF00CC" BACKGROUND="fon.png"
TEXT="#993300" LINK="#00FF00" ALINK="#FF0000" VLINK="#00FF00">
```

```
<A HREF="index.htm"><P ALIGN="center"><FONT SIZE="4" COLOR="#FFaa11" FACE="Arial">Моя первая страничка</A>
</BODY>
</HTML>
```

5 Сохраним файл под именем **index1.htm**.

6 Запустить первый файл **index.htm**. Запустим гиперссылку **Моя вторая страница**.

В качестве указателя ссылки можно использовать не только текст, но и изображение. В этом случае указатель ссылки задается с помощью тэга **<IMG>**.

Пусть в нашем проекте при щелчке мышью по маленькому рисунку, вызывается ее крупная фотография. Такой прием часто используют, для снижения времени загрузки страницы. Посетитель увидит уменьшенные копии рисунков, а при желании, сможет загрузить полномасштабное изображение. Дополним нашу страничку, введем текст в пустую строку в файл **index1.htm**. **< A HREF="carsk.jpg"><IMG SRC="carsk.gif" WIDTH="30" HEIGHT="75"></A>**

7 (Сохранить файл и обновить странички index.htm, index1.htm..)

## Практическое занятие № 22 - 30

Тема: Творческое задание.

### Цель работы:

- 1 уметь создавать свой сайт;
- 2 уметь составлять план разработки web-сайта и реализовывать его;
- 3 уметь пользоваться инструментальными средствами создания web-страниц;
- 4 уметь создавать гиперсвязи;
- 5 научиться размещать на сайте различные интерактивные формы;
- 6 научиться проводить отладку и тестирование web-сайта.

### Необходимые для выполнения работы знания:

- 1 различные виды технологий web -программирования
- 2 основные правила работы браузера Internet Explorer
- 3 основные понятия web-программирования (тэги, контейнера, атрибут)
- 4 основные этапы разработки web-сайта
- 5 правила тестирования и публикации web-сайта

### План работы:

**Творческое задание. «Мой сайт».** Создать Web-сайт, рассказывающий о вас.

Сайт рекомендуется хранить в отдельной, специально созданной папке Сайт, вложенной в папку Ваша\_фамилия. Здесь будут находиться как Web-страницы, так и графические файлы. Начальная страница сайта обычно имеет имя index.htm, в этом случае для начала просмотра сайта, в адресе достаточно указать путь к папке, где он хранится.

Начальная страница «Мой сайт» будет содержать информацию о вас, вашей семье и ваших друзьях. Страницы связаны между собой гиперссылками.

Создадим начальную Web-страницу, в процессе работы над которой, познакомимся с применением атрибутов тэгов для вставки изображений, гиперссылок, таблиц и карт-изображений.

### Дополнительный справочный материал для самостоятельного изучения.

#### **Работа с таблицами.**

Общий вид таблицы:

```
<TABLE      BORDER=3      CELLSPACING=2      CELLPADDING=2
WIDTH="80%">
  <CAPTION> ... заголовок таблицы ... </CAPTION>
  <TR><TD> первая ячейка <TD> вторая ячейка
  <TR> ...
  ...
</TABLE>
```

Атрибуты метки TABLE не обязательны. По умолчанию, таблица выводится без рамки. Как правило, размер ячеек таблицы устанавливается автоматически,

чтобы наилучшим образом разместить содержимое, однако можно установить ширину таблицы с помощью атрибута WIDTH. Атрибуты BORDER, CELLSPACING и CELLPADDING предоставляют дополнительные возможности для контроля за внешним видом таблицы. Заголовок размещается над или под таблицей в зависимости от значения атрибута ALIGN.

Горизонтальный ряд ячеек определяется элементом TR, закрывающая метка не обязательна. Ячейки таблицы определяются элементами TD (для данных) и TH (для заголовков). Как и TR, эти элементы могут не иметь закрывающей метки. TH и TD могут включать несколько атрибутов: ALIGN и VALIGN для выравнивания содержимого ячейки, ROWSPAN и COLSPAN для указания того, что ячейка занимает более одного горизонтального ряда или колонки. Ячейка таблицы может содержать другие элементы уровня блока и текста, включая формы и другие таблицы.

Для элемента TABLE обязательны открывающая и закрывающая метки. Допустимые атрибуты:

### **align**

Допустимые значения: LEFT, CENTER и RIGHT. Определяет положение таблицы по отношению к полям документа. По умолчанию установлено выравнивание по левому краю, но это можно изменить путем включения в документ элемента DIV или CENTER.

### **DIV и CENTER**

Элементы DIV можно использовать для разбиения HTML-документа на несколько разделов (англ. *division* — раздел). Для указания горизонтального выравнивания внутри элемента DIV можно использовать атрибут ALIGN с допустимыми значениями LEFT, CENTER и RIGHT (аналогично принятому для абзацев <P>).

Обратите внимание: элемент DIV по определению закрывает открытый абзац P. За исключением этого случая, браузеры **не** показывают элемент DIV на экране. Элемент <CENTER> полностью равнозначен <DIV ALIGN=CENTER>. В элементах DIV и CENTER обязательны открывающие и закрывающие метки.

### **width**

При отсутствии этого атрибута ширина таблицы определяется автоматически в зависимости от содержимого. Атрибут WIDTH можно использовать для установки фиксированной ширины в пикселах (например, WIDTH=212) или в процентах от пространства между левым и правым полем (например, WIDTH="80%").

### **border**

Используется для указания ширины внешней окантовки таблицы в пикселах (например, BORDER=4). Значение можно установить равным нулю, в результате чего окантовка не будет видна. В отсутствие этого атрибута окантовка также не должна показываться на экране. Обратите внимание: некоторые браузеры воспринимают метку <TABLE BORDER> точно так же, как BORDER=1.

### **cellspacing**

В традиционном издательском программном обеспечении расположенные рядом ячейки таблицы имеют общую окантовку. В HTML это не так. Каждая



ячейка имеет собственную окантовку. Ширина окантовки ячейки в пикселах устанавливается атрибутом `CELLSPACING` (например, `CELLSPACING=10`). Этот атрибут также устанавливает расстояние между окантовкой таблицы и окантовками крайних клеток таблицы.

### **cellpadding**

Устанавливает расстояние между окантовкой ячейки таблицы и содержимым ячейки.

Элемент `CAPTION` может иметь один атрибут `ALIGN` со значением `ALIGN=TOP` или `ALIGN=BOTTOM`. Соответственно, заголовок таблицы показывается либо над, либо под таблицей. Большинство браузеров по умолчанию показывают заголовок над таблицей. Открывающая и закрывающая метки обязательны. В заголовках таблиц не разрешаются элементы уровня блока.

Элемент `TR` открывает описание строки таблицы. Закрывающая метка не обязательна. Имеет два атрибута:

### **align**

Устанавливает горизонтальное выравнивание в ячейках строки. Допустимые значения: `LEFT`, `CENTER` и `RIGHT`. Действие аналогично атрибуту `ALIGN` у абзацев.

### **valign**

Устанавливает вертикальное выравнивание в ячейках строки. Допустимые значения: `TOP`, `MIDDLE` и `BOTTOM`; содержимое ячеек при этом выравнивается по верхнему краю, по центру или по нижнему краю.

Существует два элемента, определяющих ячейки таблицы. `TH` используется для ячеек-заголовков, а `TD` - для ячеек с данными. Открывающие метки обязательны, закрывающие - нет. Ячейки могут иметь следующие атрибуты:

### **nowrap**

Этот атрибут запрещает автоматический перевод текста со строки на строку внутри ячейки таблицы (например, `<TD NOWRAP>`). Эффект применения этого атрибута аналогичен использованию объекта `&nbsp;`; вместо пробела по всему содержанию ячейки.

### **rowspan**

Используется с положительным целым значением числа строк таблицы, занятых ячейкой. По умолчанию принято равным единице.

### **colspan**

Используется с положительным целым значением числа колонок таблицы, занятых ячейкой. По умолчанию принято равным единице.

### **align**

Указывает принятое по умолчанию выравнивание содержимого ячейки. Имеет приоритет над значением, установленным атрибутом `ALIGN` строки таблицы. Допустимые значения те же самые: `LEFT`, `CENTER` и `RIGHT`. Если атрибут `ALIGN` не указан, принятое по умолчанию выравнивание - по левому краю для `<td>` и по центру для `<th>`, однако это можно изменить установкой атрибута `ALIGN` элемента `TR`.

### **valign**

Указывает принятое по умолчанию выравнивание содержимого ячейки.

Имеет приоритет над значением, установленным атрибутом ALIGN строки таблицы. Допустимые значения те же самые: TOP, MIDDLE и BOTTOM. Если атрибут VALIGN не установлен, по умолчанию принято выравнивание по центру, однако это можно изменить установкой атрибута VALIGN элемента TR.

**width**

Указывает рекомендуемую ширину содержимого ячейки в пикселах. Значение используется только в случаях, когда оно не противоречит требованиям к ширине других ячеек в колонке.

**height**

Указывает рекомендуемую высоту содержимого ячейки в пикселах. Значение используется только в случаях, когда оно не противоречит требованиям к высоте других ячеек в строке.

Таблицы, как правило, показываются на экране "приподнятыми" над поверхностью страницы, а ячейки - "вдавленными" в тело таблицы. Ячейки выделяются окантовкой только если в них есть содержимое. Если содержимое ячейки состоит только из пробелов, ячейка считается пустой, за исключением случаев, когда в ней есть хотя бы один объект &nbsp;.

## Практическое занятие № 31

Тема: Введение в язык программирования Python.

### Цель работы:

1. Познакомиться со средой разработки Python.
2. Изучить основные типы данных, команды ввода и вывода данных

### Теоретические сведения:

Python – это объектно-ориентированный, интерпретируемый, переносимый язык сверхвысокого уровня. Программирование на Python позволяет получать быстро и качественно необходимые программные модули.

В комплекте вместе с интерпретатором Python идет IDLE (интегрированная среда разработки). По своей сути она подобна интерпретатору, запущенному в интерактивном режиме с расширенным набором возможностей (подсветка синтаксиса, просмотр объектов, отладка и т.п.).

Для запуска IDLE в Windows необходимо перейти в папку Python в меню “Пуск” и найти там ярлык с именем “IDLE (Python 3.X XX-bit)”.

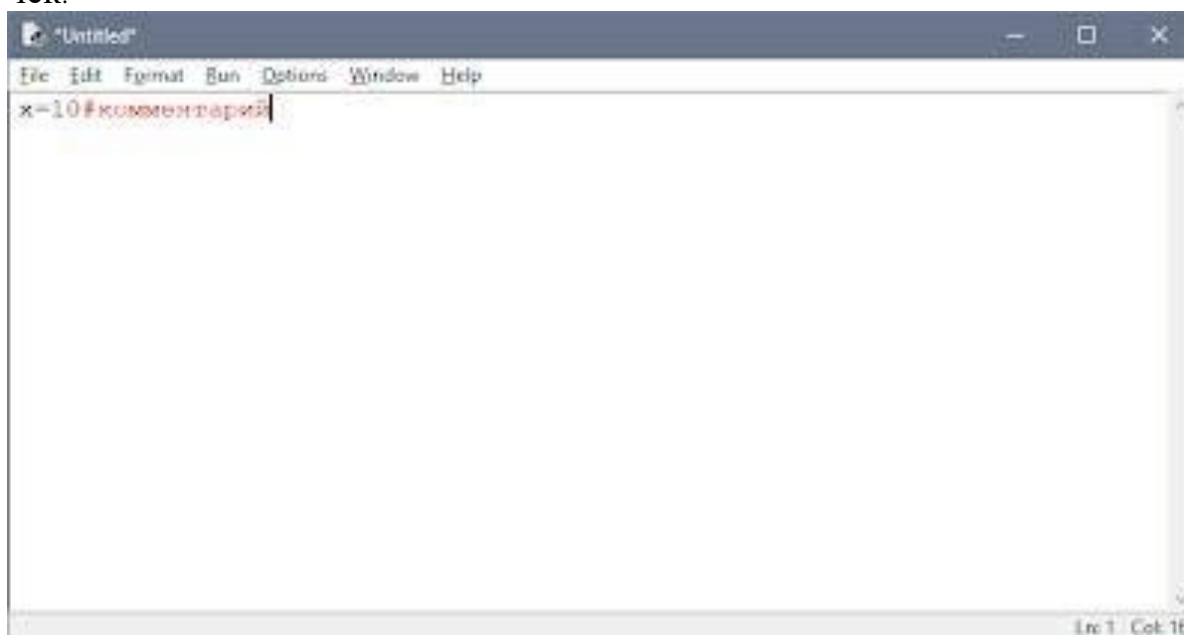
Для запуска редактора программы (кода) следует выполнить команду File->NewFile или сочетание клавиш Ctrl+N.

Любая Python-программа состоит из последовательности допустимых символов, записанных в определенном порядке и по определенным правилам.

Программа включает в себя:

- комментарии;
- команды;
- знаки пунктуации;
- идентификаторы;
- ключевые слова.

Комментарии в Python обозначаются предваряющим их символом # и продолжаются до конца строки (т.е. в Python все комментарии являются однострочными), при этом не допускается использование перед символом # кавычек:



## Знаки пунктуации

В алфавит Python входит достаточное количество знаков пунктуации, которые используются для различных целей. Например, знаки "+" или "\*" могут использоваться для сложения и умножения, а знак запятой "," - для разделения параметров функций.

## Идентификаторы

Идентификаторы в Python это имена используемые для обозначения переменной, функции, класса, модуля или другого объекта.

## Ключевые слова

Некоторые слова имеют в Python специальное назначение и представляют собой управляющие конструкции языка.

### Ключевые слова в Python:

```
['False', 'None', 'True', 'and', 'as', 'assert', 'break', 'class', 'continue', 'def', 'del', 'elif', 'else', 'except', 'finally', 'for', 'from', 'global', 'if', 'import', 'in', 'is', 'lambda', 'non-local', 'not', 'or', 'pass', 'raise', 'return', 'try', 'while', 'with', 'yield']
```

## Типы данных

1. None (неопределенное значение переменной)
2. Логические переменные (Boolean Type)
3. Числа (Numeric Type)
  - 1) int – целое число
  - 2) float – число с плавающей точкой
  - 3) complex – комплексное число
4. Списки (Sequence Type)
  - 1) list – список
  - 2) tuple – кортеж
  - 3) range – диапазон
5. Строки (Text Sequence Type)
  1. Str

## Ввод и вывод данных

Ввод данных осуществляется при помощи команды **input**(список ввода):

```
a = input()
print(a)
```

В скобках функции можно указать сообщение - комментарий к вводимым данным:

```
a = input ("Введите количество: ")
```

Команда `input()` по умолчанию воспринимает входные данные как строку символов. Поэтому, чтобы ввести целочисленное значение, следует указать тип данных `int()`:

```
a = int (input())
```

Для ввода вещественных чисел применяется команда `a=float(input())`

Вывод данных осуществляется при помощи команды **print**(список вывода):

```
a = 1  
b = 2  
print(a) print(a + b)  
print('сумма = ', a + b)
```

Существует возможность записи команд в одну строку, разделяя их через **;**. Однако не следует часто использовать такой способ, это снижает удобочитаемость:

```
a = 1; b = 2; print(a)print (a + b)  
print ('сумма = ', a + b)
```

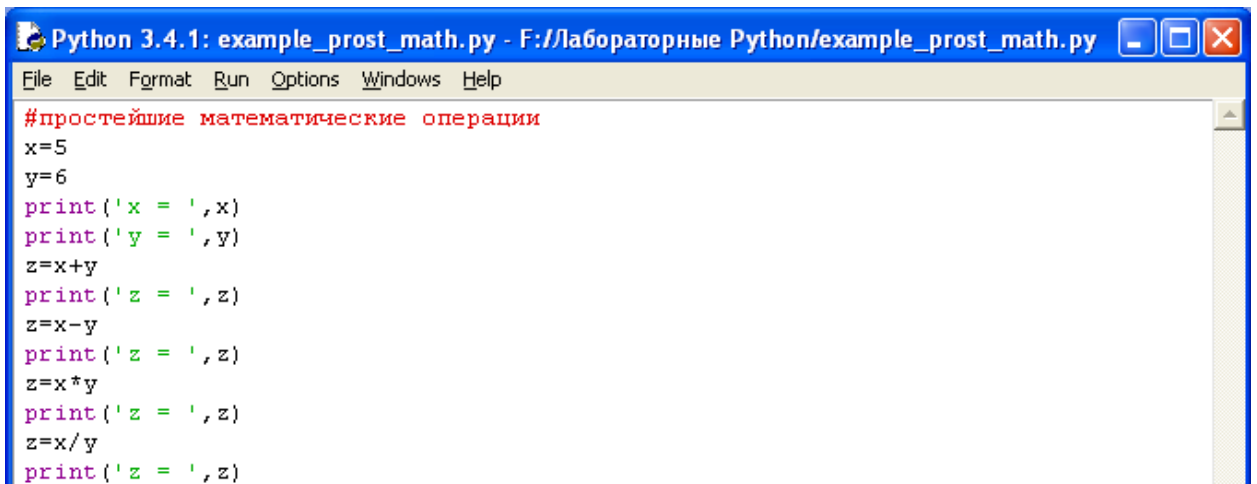
Для команды **print** может задаваться так называемый сепаратор — разделитель между элементами вывода:

```
x=2y=5  
print ( x, "+", y, "=", x+y, sep = " " )
```

Результат отобразится с пробелами между элементами:  $2 + 5 = 7$

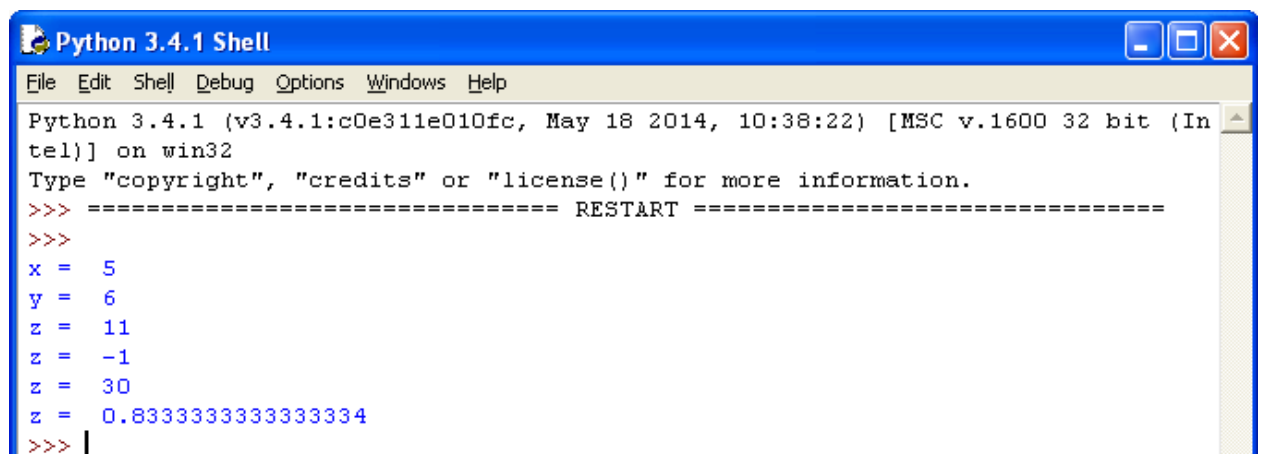
### Простые арифметические операции над числами

$x + y$	Сложение
$x - y$	Вычитание
$x * y$	Умножение
$x / y$	Деление



```
Python 3.4.1: example_prost_math.py - F://Лабораторные Python/example_prost_math.py
File Edit Format Run Options Windows Help
#простейшие математические операции
x=5
y=6
print('x = ',x)
print('y = ',y)
z=x+y
print('z = ',z)
z=x-y
print('z = ',z)
z=x*y
print('z = ',z)
z=x/y
print('z = ',z)
```

### Пример программы на Python



```
Python 3.4.1 Shell
File Edit Shell Debug Options Windows Help
Python 3.4.1 (v3.4.1:c0e311e010fc, May 18 2014, 10:38:22) [MSC v.1600 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ===== RESTART =====
>>>
x = 5
y = 6
z = 11
z = -1
z = 30
z = 0.8333333333333334
>>> |
```

Результат выполнения программы с применением простых арифметических операций

Для форматированного вывода используется **format**:

Строковый метод `format()` возвращает отформатированную версию строки, заменяя идентификаторы в фигурных скобках `{}`. Идентификаторы могут быть позиционными, числовыми индексами, ключами словарей, именами переменных.

Синтаксис команды **format**:

поле замены := "{" [имя поля] ["!" преобразование] [":" спецификация] "}"

имя поля := `arg_name` ( "." имя атрибута | "[" индекс "]" )\*

преобразование := "r" (внутреннее представление) | "s" (человеческое представление)

спецификация := см. ниже

Аргументов в `format()` может быть больше, чем идентификаторов в строке. В таком случае оставшиеся игнорируются.

Идентификаторы могут быть либо индексами аргументов, либо ключами:

```
example_format.py - K:\Лабораторные Python\example_format.py (3.7.1)
File Edit Format Run Options Window Help
x=11
print(x) #вывод без форматирования
print ("{:4}".format(x)) #перед значением переменной x будет присутствовать 2 пробела,
                        #так как число 11 занимает 2 знакоместа
```

```
Python 3.7.1 Shell
File Edit Shell Debug Options Window Help
Python 3.7.1 (v3.7.1:260ec2c36a, Oct 20 2018, 14:05:16) [MSC v.1915 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: K:\Лабораторные Python\example_format.py =====
11
   11
>>> |
```

В результате выведется число 11, а перед ним два пробела, так как указано использовать для вывода четыре знакоместа.

Или с несколькими аргументами:

```
example_format1.py - K:\Лабораторные Python\example_format1.py (3.7.1)
File Edit Format Run Options Window Help
x=2
print ("{:4d}{:4d}{:4d}".format(x,x,x))

Python 3.7.1 (v3.7.1:260ec2c36a, Oct 20 2018, 14:05:16) [MSC v.1915 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: K:\Лабораторные Python\example_format1.py =====
   2   2   2
>>>
```

В итоге каждое из значений выводится из расчета 4 знакоместа.

Спецификация формата:

спецификация	:= <code>[[fill]align][sign][#][0][width][,][.precision][type]</code>
заполнитель	:= символ кроме <code>'</code> или <code>'</code>
выравнивание	:= <code>"&lt;"   "&gt;"   "="   "^"</code>
знак	:= <code>"+"   "-"   ""</code>
ширина	:= integer

точность	:= integer
тип	:= "b"   "c"   "d"   "e"   "E"   "f"   "F"   "g"   "G"   "n"   "o"   "s"   "x"   "X"   "%"

Тип	Значение
'd', 'i', 'u'	Десятичное число.
'o'	Число в восьмеричной системе счисления.
'x'	Число в шестнадцатеричной системе счисления (буквы в нижнем регистре).
'X'	Число в шестнадцатеричной системе счисления (буквы в верхнем регистре).
'e'	Число с плавающей точкой с экспонентой (экспонента в нижнем регистре).
'E'	Число с плавающей точкой с экспонентой (экспонента в верхнем регистре).
'f', 'F'	Число с плавающей точкой (обычный формат).
'g'	Число с плавающей точкой. с экспонентой (экспонента в нижнем регистре), если она меньше, чем -4 или точности, иначе обычный формат.
'G'	Число с плавающей точкой. с экспонентой (экспонента в верхнем регистре), если она меньше, чем -4 или точности, иначе обычный формат.
'c'	Символ (строка из одного символа или число - код символа).
's'	Строка.
'%'	Число умножается на 100, отображается число с плавающей точкой, а за ним знак %.

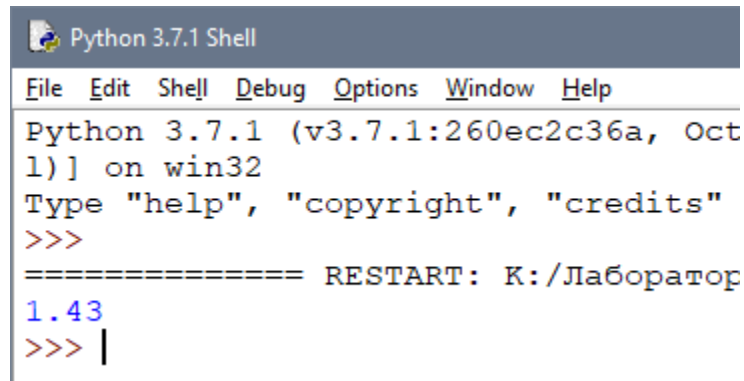
Для форматирования вещественных чисел с плавающей точкой используется следующая команда:

```
print('{0:.2f}'.format(вещественное число))
```

```
format_chisla.py - K://Лабораторные Python/format_chisla.py (3.7.1)
File Edit Format Run Options Window Help
x=10
y=7
print("{0:.2f}".format(x/y))
```



В результате выведется число с двумя знаками после запятой.



```
Python 3.7.1 Shell
File Edit Shell Debug Options Window Help
Python 3.7.1 (v3.7.1:260ec2c36a, Oct
1)] on win32
Type "help", "copyright", "credits"
>>>
===== RESTART: К:/Лаборатор
1.43
>>> |
```

Пример

Напишите программу, которая запрашивала бы у пользователя:

- ФИО ("Ваши фамилия, имя, отчество?")
  - возраст ("Сколько Вам лет?")
  - место жительства ("Где вы живете?")
- После этого выводила бы три строки: "Ваше имя"  
"Ваш возраст""Вы живете в"

**Решение**

```
a=input('Введите ваши фамилию, имя, отчество ')
b=input('Сколько вам лет? ')
c=input('Где вы живёте? ')
print('Ваше имя ',a)
print('Ваш возраст ',b)
print('Вы живете в ',c)|
```

```
Введите ваши фамилию, имя, отчество Иванов Иван Иванович
Сколько вам лет? 15
Где вы живёте? Уссурийск
Ваше имя Иванов Иван Иванович
Ваш возраст 15
Вы живете в Уссурийск
```

Задания для самостоятельной работы

- 1) Включите Python
  - 2) Напишите программу, которая запрашивала бы у пользователя:  
Имя, Фамилия, Возраст, Место жительства
- фамилия, имя ("Ваши фамилия, имя?")
  - возраст ("Сколько Вам лет?")

- место жительства ("Где вы живете?") После этого выводила бы три строки: "Ваши фамилия, имя"  
"Ваш возраст" "Вы живете в"

## Практическое занятие № 32

Тема: Математические операции в Python

Цель работы:

Познакомиться с основными математическими операциями в Python.

Теоретические сведения:

Язык Python, благодаря наличию огромного количества библиотек для решения разного рода вычислительных задач, сегодня является конкурентом таким пакетам как Matlab и Octave. Запущенный в интерактивном режиме, он, фактически, превращается в мощный калькулятор. В этом уроке речь пойдет об арифметических операциях, доступных в данном языке. Арифметические операции изучим применительно к числам.

Если в качестве операндов некоторого арифметического выражения используются только целые числа, то результат тоже будет целое число. Исключением является операция деления, результатом которой является вещественное число. При совместном использовании целочисленных и вещественных переменных, результат будет вещественным.

Речь пойдет об арифметических операциях, доступных в данном языке.

Если в качестве операндов некоторого арифметического выражения используются только целые числа, то результат тоже будет целое число. Исключением является операция деления, результатом которой является вещественное число. При совместном использовании целочисленных и вещественных переменных, результат будет вещественным.

### Целые числа (int)

Числа в Python 3 поддерживают набор самых обычных математических операций:

$x + y$	Сложение
$x - y$	Вычитание
$x * y$	Умножение
$x / y$	Деление
$x // y$	Получение целой части от деления
$x \% y$	Остаток от деления
$-x$	Смена знака числа
$abs(x)$	Модуль числа

<b>divmod(x, y)</b>	Пара (x // y, x % y)
<b>x ** y</b>	Возведение в степень
<b>pow(x, y[, z])</b>	<p>x : Число, которое требуется возвести в степень.  y : Число, являющееся степенью, в которую нужно возвести первый аргумент. Если число отрицательное или одно из чисел "x" или "y" нецелые, то аргумент "z" не принимается.  z : Число, на которое требуется произвести деление по модулю. Если число указано, ожидается, что "x" и "y" положительны и имеют тип int.</p>

### Пример применения вышеописанных операций над целыми числами

```
x = 5
y = 2
z = 3
x+y = 7
x-y = 3
x*y = 10
x/y = 2.5
x//y = 2
x%y = 1
-x = -5
abs(-x) = 5
divmod(x, y) = (2, 1)
x**y = 25
pow(x, y, z) = 1
```

### Вещественные числа (float)

Вещественные числа поддерживают те же операции, что и целые. Однако (из-за представления чисел в компьютере) вещественные числа неточны, и это может привести к ошибкам.

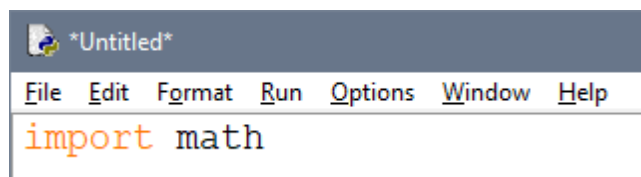
Пример применения вышеописанных операций над вещественными числами

```
x = 5.5
y = 2.3
x+y = 7.8
x-y = 3.2
x*y = 12.649999999999999
x/y = 2.3913043478260874
x//y = 2.0
x%y = 0.9000000000000004
-x = -5.5
abs(-x) = 5.5
divmod(x, y) = (2.0, 0.9000000000000004)
x**y = 50.44686540422945
```

## Библиотека (модуль) `math`

В стандартную поставку Python входит библиотека `math`, в которой содержится большое количество часто используемых математических функций.

Для работы с данным модулем его предварительно нужно импортировать.



```
*Untitled*
File Edit Format Run Options Window Help
import math
```

Рассмотрим наиболее часто используемые функции модуля `math`

<code>math.ceil(x)</code>	Возвращает ближайшее целое число большее, чем $x$
<code>math.fabs(x)</code>	Возвращает абсолютное значение числа $x$
<code>math.factorial(x)</code>	Вычисляет факториал $x$
<code>math.floor(x)</code>	Возвращает ближайшее целое число меньшее, чем $x$
<code>math.exp(x)</code>	Вычисляет $e^{**}x$
<code>math.log2(x)</code>	Логарифм по основанию 2
<code>math.log10(x)</code>	Логарифм по основанию 10
<code>math.log(x, base)</code>	По умолчанию вычисляет логарифм по основанию $e$ , дополнительно можно указать основание логарифма
<code>math.pow(x, y)</code>	Вычисляет значение $x$ в степени $y$
<code>math.sqrt(x)</code>	Корень квадратный от $x$

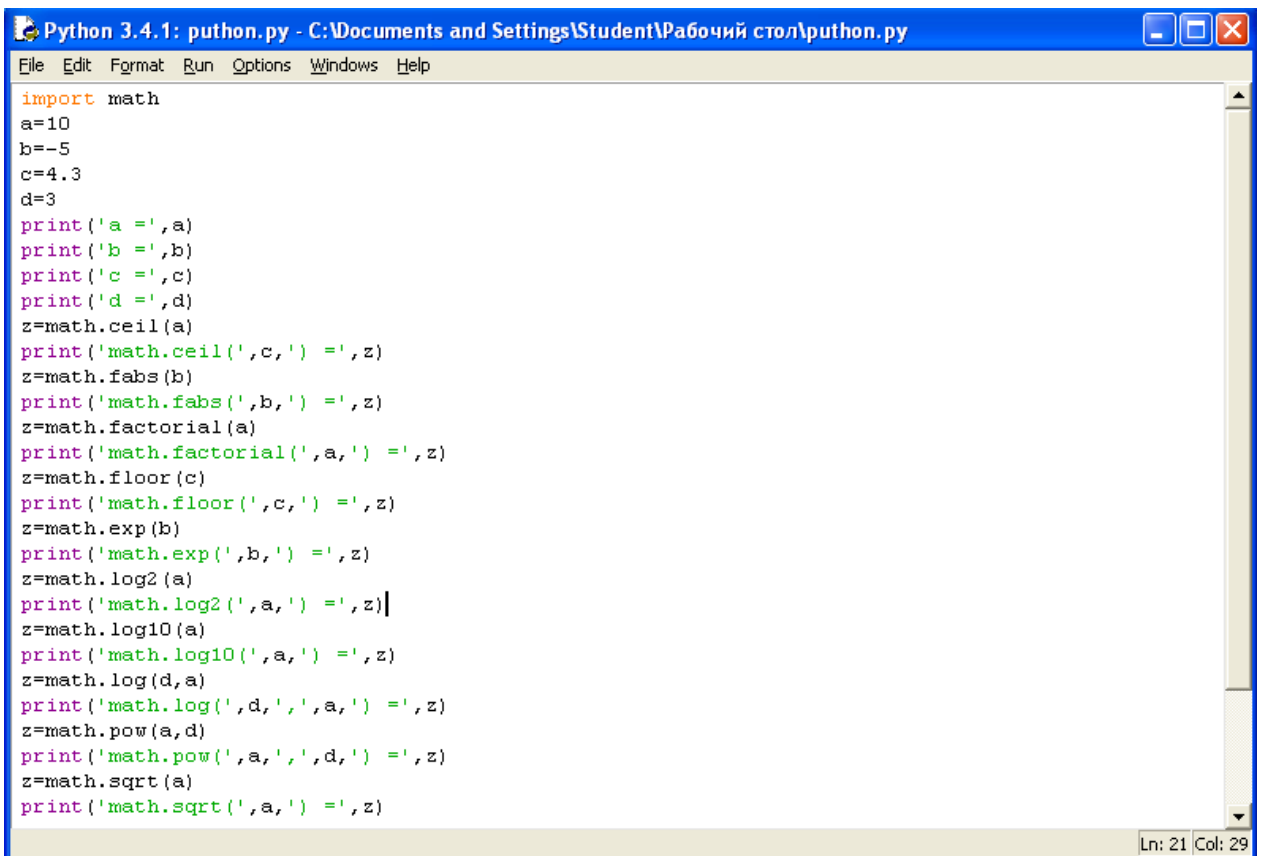
### Пример применения вышеописанных функций над числами

В программе определены 4 переменные -  $a$ ,  $b$ ,  $c$ ,  $d$ , каждая из которых является либо целым числом, либо вещественным, либо отрицательным.

Командой `print()` выводится значение каждой переменной на экран при выполнении программы.

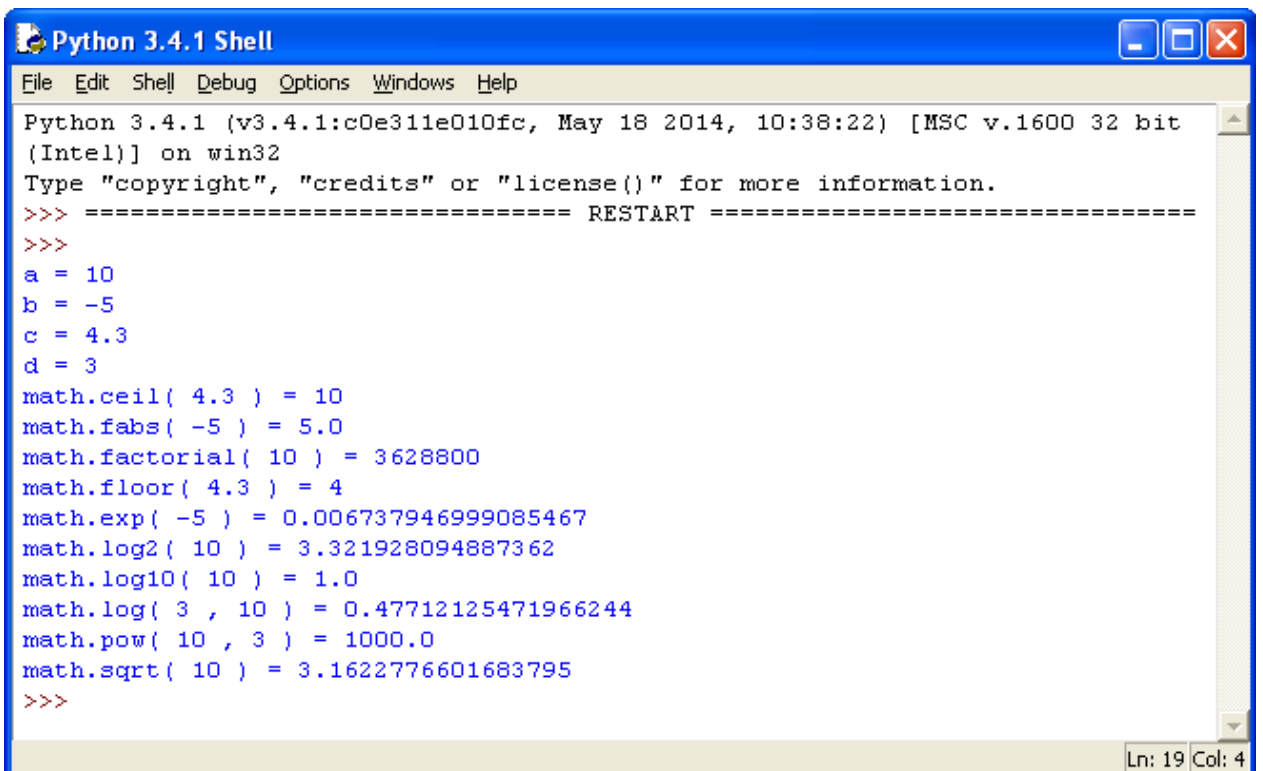
В переменную  $z$  помещается результат выполнения функции модуля `math`.

Затем командой `print()` выводится сообщение в виде используемой функции и её аргумента и результат её выполнения.



```
Python 3.4.1: puthon.py - C:\Documents and Settings\Student\Рабочий стол\puthon.py
File Edit Format Run Options Windows Help
import math
a=10
b=-5
c=4.3
d=3
print('a =',a)
print('b =',b)
print('c =',c)
print('d =',d)
z=math.ceil(a)
print('math.ceil(',c,') =',z)
z=math.fabs(b)
print('math.fabs(',b,') =',z)
z=math.factorial(a)
print('math.factorial(',a,') =',z)
z=math.floor(c)
print('math.floor(',c,') =',z)
z=math.exp(b)
print('math.exp(',b,') =',z)
z=math.log2(a)
print('math.log2(',a,') =',z)
z=math.log10(a)
print('math.log10(',a,') =',z)
z=math.log(d,a)
print('math.log(',d,',',a,') =',z)
z=math.pow(a,d)
print('math.pow(',a,',',d,') =',z)
z=math.sqrt(a)
print('math.sqrt(',a,') =',z
Ln: 21 Col: 29
```

### Пример программы на Python



```
Python 3.4.1 Shell
File Edit Shell Debug Options Windows Help
Python 3.4.1 (v3.4.1:c0e311e010fc, May 18 2014, 10:38:22) [MSC v.1600 32 bit
(Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ===== RESTART =====
>>>
a = 10
b = -5
c = 4.3
d = 3
math.ceil( 4.3 ) = 5
math.fabs( -5 ) = 5.0
math.factorial( 10 ) = 3628800
math.floor( 4.3 ) = 4
math.exp( -5 ) = 0.006737946999085467
math.log2( 10 ) = 3.321928094887362
math.log10( 10 ) = 1.0
math.log( 3 , 10 ) = 0.47712125471966244
math.pow( 10 , 3 ) = 1000.0
math.sqrt( 10 ) = 3.1622776601683795
>>>
Ln: 19 Col: 4
```

Результат выполнения программы с применением функций модуля math

## Тригонометрические функции модуля math

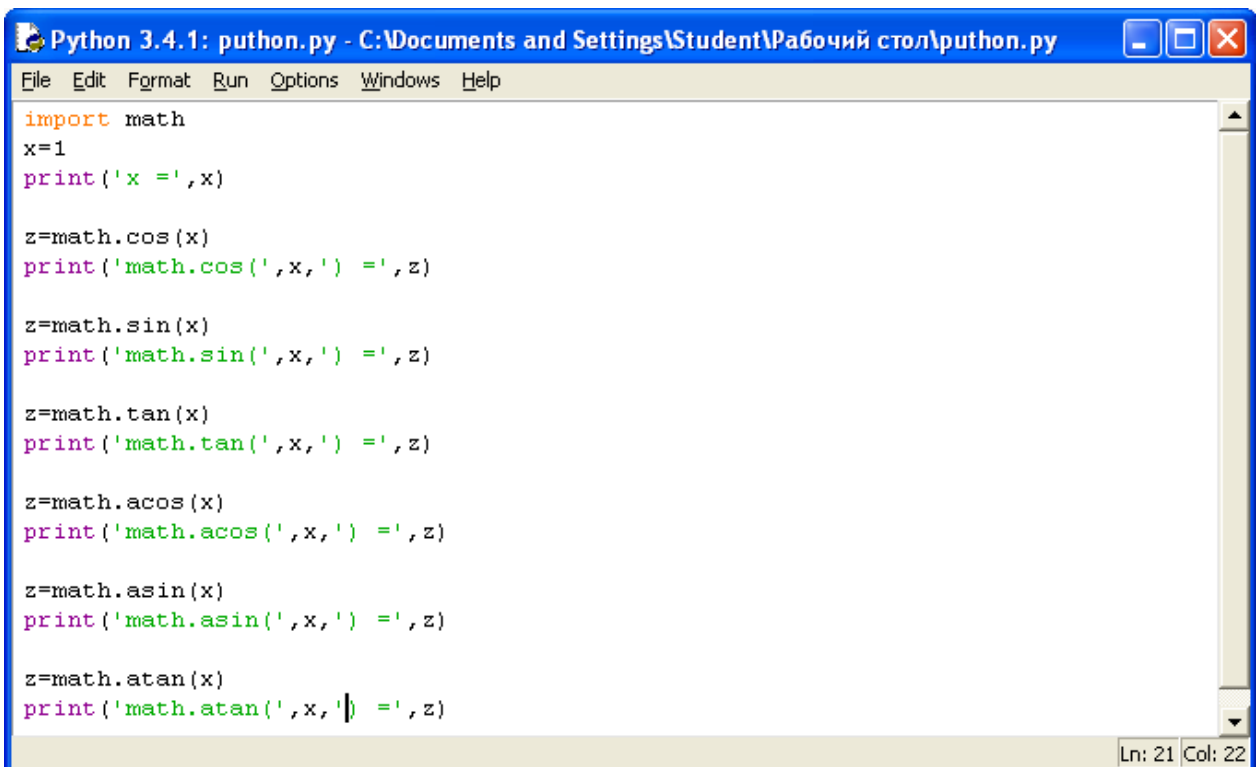
<b>math.cos(x)</b>	Возвращает cos числа X
<b>math.sin(x)</b>	Возвращает sin числа X
<b>math.tan(x)</b>	Возвращает tan числа X
<b>math.acos(x)</b>	Возвращает acos числа X
<b>math.asin(x)</b>	Возвращает asin числа X
<b>math.atan(x)</b>	Возвращает atan числа X

### Пример применения вышеописанных функций над числами

В программе определена переменная x, содержащая целое число. Значение переменной выводится командой print() на экран.

В переменную z помещается результат выполнения тригонометрической функции модуля math.

Затем командой print() выводится сообщение в виде используемой функции и её аргумента, и результат её выполнения.



```
Python 3.4.1: puthon.py - C:\Documents and Settings\Student\Рабочий стол\puthon.py
File Edit Format Run Options Windows Help
import math
x=1
print('x =', x)

z=math.cos(x)
print('math.cos(', x, ') =', z)

z=math.sin(x)
print('math.sin(', x, ') =', z)

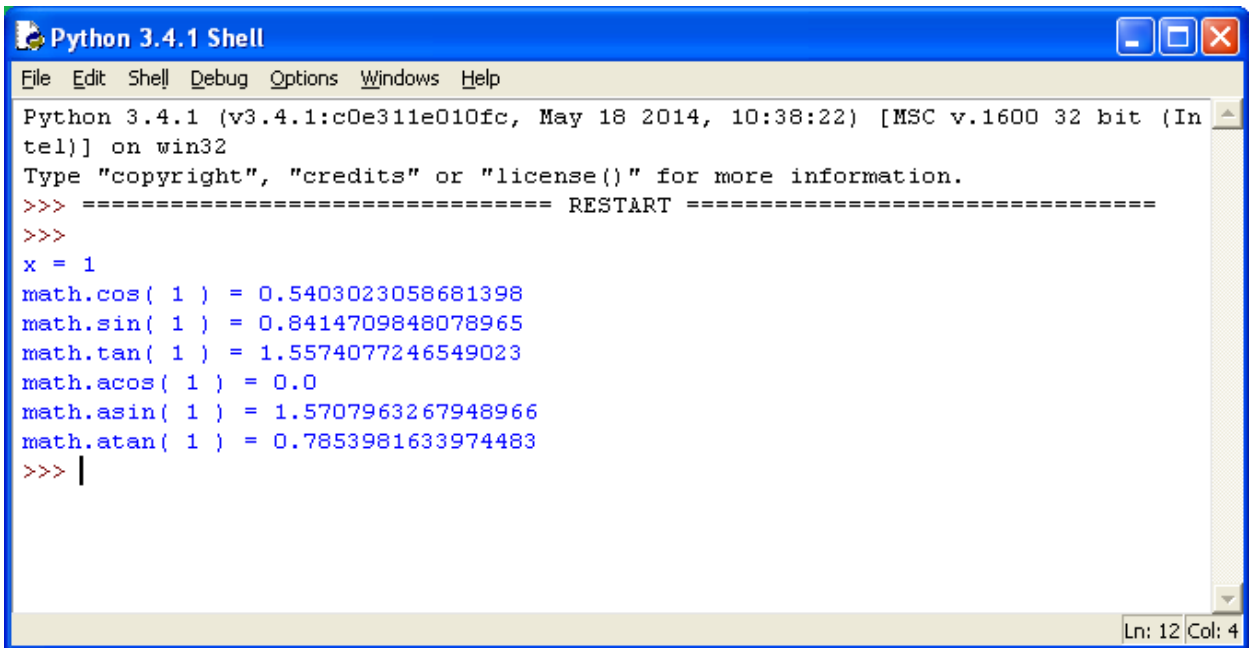
z=math.tan(x)
print('math.tan(', x, ') =', z)

z=math.acos(x)
print('math.acos(', x, ') =', z)

z=math.asin(x)
print('math.asin(', x, ') =', z)

z=math.atan(x)
print('math.atan(', x, ') =', z)
Ln: 21 Col: 22
```

Пример программы с использованием тригонометрических функций модуля math



```
Python 3.4.1 Shell
File Edit Shell Debug Options Windows Help
Python 3.4.1 (v3.4.1:c0e311e010fc, May 18 2014, 10:38:22) [MSC v.1600 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ===== RESTART =====
>>>
x = 1
math.cos( 1 ) = 0.5403023058681398
math.sin( 1 ) = 0.8414709848078965
math.tan( 1 ) = 1.5574077246549023
math.acos( 1 ) = 0.0
math.asin( 1 ) = 1.5707963267948966
math.atan( 1 ) = 0.7853981633974483
>>> |
```

Результат выполнения программы с применением тригонометрических функций модуля math

### Константы:

- **math.pi** - число Pi.
- **math.e** - число e (экспонента).

### Пример

Напишите программу, которая бы вычисляла заданное арифметическое выражение при заданных переменных. Ввод переменных осуществляется с клавиатуры. Вывести результат с 2-мя знаками после запятой.

### Задание

$$Z = \frac{9\pi t + 10 \cos(x)}{\sqrt{t} - |\sin(t)|} * e^x$$

x=10; t=1

### Решение

Сначала импортируем модуль math. Для этого воспользуемся командой `import math`.

Затем следует ввести значения двух переменных целого типа x и t.

Для ввода данных используется команда `input`, но так как в условии даны целые числа, то нужно сначала определить тип переменных: `x=int()`, `t=int()`.

Определив тип переменных, следует их ввести, для этого в скобках команды `int()` нужно написать команду `input()`.

Для переменной x это выглядит так: `x=int(input("сообщение при вводе`



значения"))).

Для переменной  $t$  аналогично: `t=int(input("сообщение при вводе значения"))`. Следующий шаг - это составление арифметического выражения, результат которого поместим в переменную  $z$ .

Сначала составим числитель. Выглядеть он будет так:  
`9*math.pi*t+10*math.cos(x)`.

Затем нужно составить знаменатель, при этом обратим внимание на то, что числитель делится на знаменатель, поэтому и числитель и знаменатель нужно поместить в скобки `()`, а между ними написать знак деления `/`.

Выглядеть это будет так: `(9*math.pi*t+10*math.cos(x))/(math.sqrt(t)-math.fabs(math.sin(t)))`.

Последним шагом является умножение дроби на экспоненту в степени  $x$ .

Так как умножается вся дробь, то следует составленное выражение поместить в скобки `()`, а уже потом написать функцию `math.pow(math.e,x)`.

В результате выражение будет иметь вид:

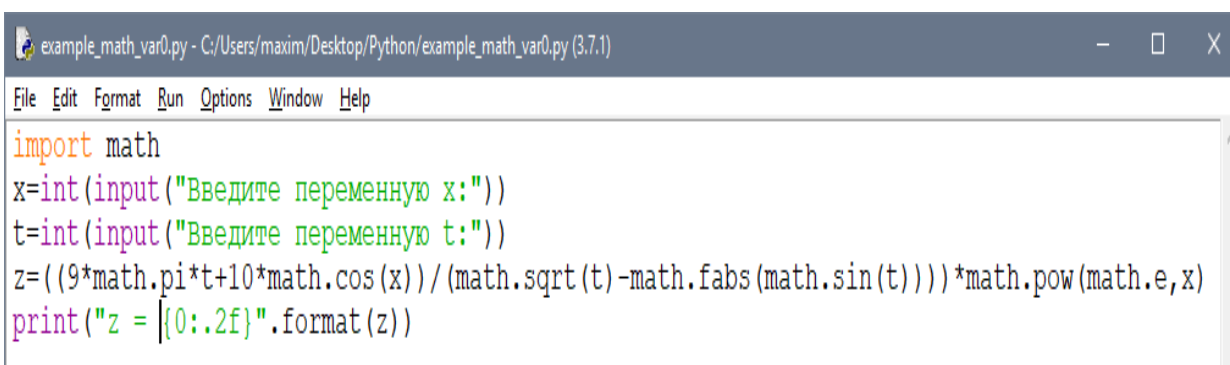
`z=((9*math.pi*t+10*math.cos(x))/(math.sqrt(t)-math.fabs(math.sin(t))))*math.pow(math.e,x)`.

При составлении данного выражения следует обратить внимание на количество открывающихся и закрывающихся скобок.

Командой `print()` выведем значение переменной, отформатировав его командой `format`.

Сам формат записывается в апострофах в фигурных скобках `{ }`.

В задаче требуется вывести число с двумя знаками после запятой, значит вид формата будет выглядеть следующим образом: `{0:.2f}`, где `2` - это количество знаков после запятой, а `f` указывает на то, что форматируется вещественное число. При этом перед `2` нужно поставить точку, указав тем самым на то, что форматируем именно дробную часть числа.



```
example_math_var0.py - C:/Users/maxim/Desktop/Python/example_math_var0.py (3.7.1)
File Edit Format Run Options Window Help
import math
x=int(input("Введите переменную x:"))
t=int(input("Введите переменную t:"))
z=((9*math.pi*t+10*math.cos(x))/(math.sqrt(t)-math.fabs(math.sin(t))))*math.pow(math.e,x)
print("z = {0:.2f}".format(z))
```

## Результат

```
Python 3.7.1 Shell
File Edit Shell Debug Options Window Help
Python 3.7.1 (v3.7.1:260ec2c36a,
1)] on win32
Type "help", "copyright", "credit
>>>
===== RESTART: C:/Users/maxim/
Введите переменную x:10
Введите переменную t:1
z = 2762685.71
>>> |
```

Задания для самостоятельной работы

Воспроизвести задание из примера. Сделать скриншоты кода и результата.

## Практическое занятие № 33

Тема: Структура ветвление в Python.

### Цель работы:

1. Познакомиться со структурой ветвление (if, if-else, if-elif-else).
2. Научиться работать с числами и строками используя данную структуру.

### Теоретические сведения:

Условный оператор ветвления if, if-else, if-elif-else

Оператор ветвления if позволяет выполнить определенный набор инструкций в зависимости от некоторого условия. Возможны следующие варианты использования.

#### 1. Конструкция if

Синтаксис оператора if выглядит так:

**if логическое выражение:**

команда\_1 команда\_2

...

команда\_n

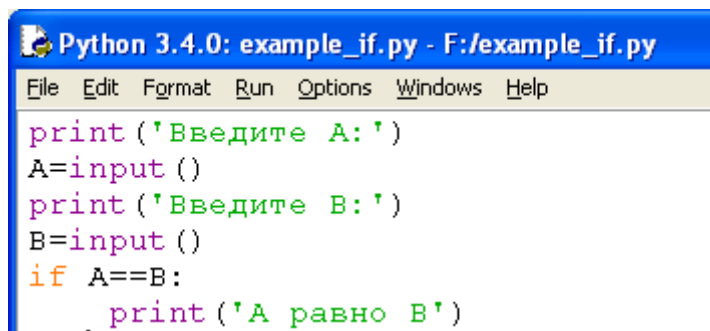
После оператора if записывается логическое выражение.

**Логическое выражение** — конструкция [языка программирования](#), результатом вычисления которой является «истина» или «ложь».

Если это выражение истинно, то выполняются инструкции, определяемые данным оператором. Выражение является истинным, если его результатом является число не равное нулю, непустой объект, либо логическое True. После выражения нужно поставить двоеточие “:”.

**ВАЖНО:** блок кода, который необходимо выполнить, в случае истинности выражения, отделяется **четырьмя** пробелами слева!

Программа запрашивает у пользователя два числа, затем сравнивает их и если числа равны, то есть логическое выражение  $A==B$  истинно, то выводится соответствующее сообщение.



```
Python 3.4.0: example_if.py - F:/example_if.py
File Edit Format Run Options Windows Help
print ('Введите A:')
A=input ()
print ('Введите B:')
B=input ()
if A==B:
    print ('A равно B')
```

Пример программы на Python

```
Python 3.4.0 Shell
File Edit Shell Debug Options Windows Help
Python 3.4.0 (v3.4.0:04f714765c13, Mar 16 201
tel)] on win32
Type "copyright", "credits" or "license()" fo
>>> ===== RESTART
>>>
>>>
Введите А:
10
Введите В:
10
А равно В
>>> |
```

Результат выполнения программы с использованием условного оператора if

## 2. Конструкция if – else

**Бывают случаи, когда необходимо предусмотреть**

альтернативный вариант выполнения программы. Т.е. при истинном условии нужно выполнить один набор

**инструкций, при ложном – другой. Для этого используется конструкция if – else.**

Синтаксис оператора if – else выглядит так:

**if** логическое выражение:

команда\_1

команда\_2

...

команда\_n

**else:**

команда\_1команда\_2

...

команда\_n

Программа запрашивает у пользователя два числа, затем сравнивает их и если числа равны, то есть логическое выражение  $A==B$  истинно, то выводится соответствующее сообщение. В противном случае выводится сообщение, что числа не равны.

### Пример программы на Python

```
Python 3.4.0: example_if.py - F:/example_if.py
File Edit Format Run Options Windows Help
print ('Введите A:')
A=input ()
print ('Введите B:')
B=input ()
if A==B:
    print ('A равно B')
else:
    print ('A не равно B')
```

Пример программы на Python

```
>>> ===== RESTART
>>>
Введите A:
10
Введите B:
5
A не равно B
>>> |
```

Результат выполнения программы с использованием условного оператора if-else

### 3. Конструкция if – elif – else

Для реализации выбора из нескольких альтернатив можно использовать конструкцию if – elif – else.

Синтаксис оператора if – elif – else выглядит так:

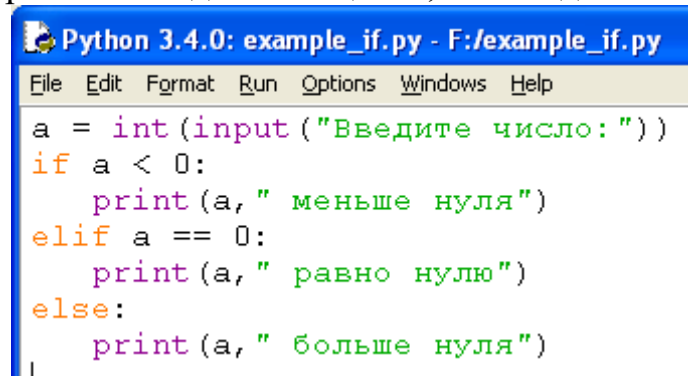
```
if логическое выражение_1:
    команда_1команда_2
...
команда_n
elif логическое выражение_2:
    команда_1команда_2
...
команда_n
elif логическое выражение_3:
    команда_1команда_2
...
команда_n
else:
    команда_1команда_2
```

...

команда\_n

Программа запрашивает число у пользователя и сравнивает его с нулём  $a < 0$ .

Если оно меньше нуля, то выводится сообщение об этом. Если первое логическое выражение не истинно, то программа переходит ко второму -  $a == 0$ . Если оно истинно, то программа выведет сообщение, что число равно нулю, в противном случае, если оба вышеуказанных логических выражения оказались ложными, то программа выведет сообщение, что введённое число больше нуля.



```
Python 3.4.0: example_if.py - F:/example_if.py
File Edit Format Run Options Windows Help
a = int(input("Введите число:"))
if a < 0:
    print(a, " меньше нуля")
elif a == 0:
    print(a, " равно нулю")
else:
    print(a, " больше нуля")
```

Пример программы на Python

```
Введите число:41
41 больше нуля
>>> ===== RESTART =
>>>
Введите число:-5
-5 меньше нуля
>>> ===== RESTART =
>>>
Введите число:0
0 равно нулю
>>>
```

Результат выполнения программы с использованием условного оператора if-elif-else

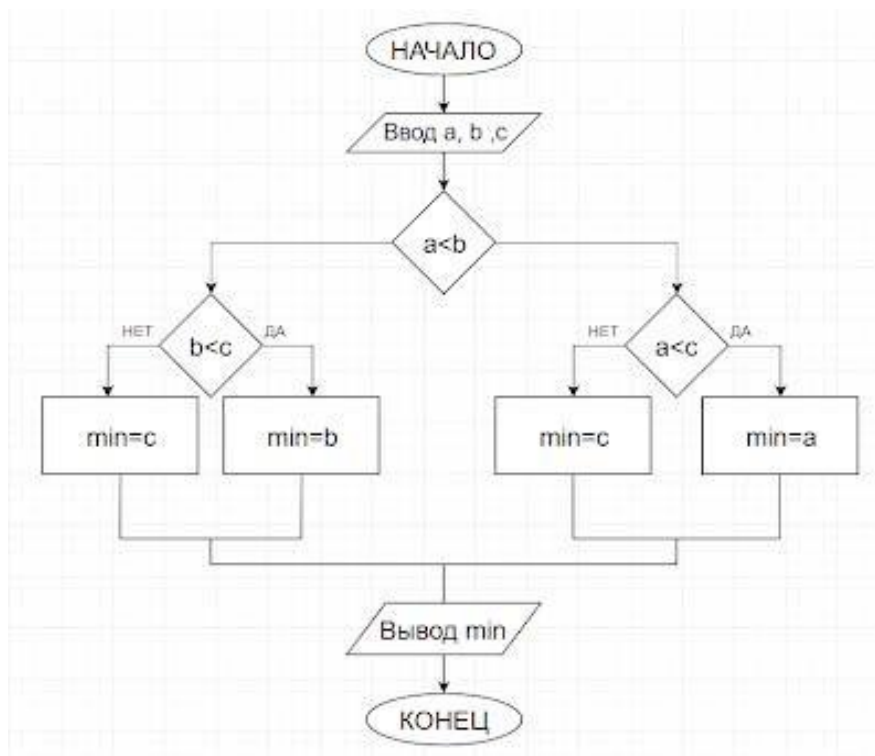
Пример

**Задание**

Дано 3 числа. Найти минимальное среди них и вывести на экран.

**Решение**

Для простоты построим блок-схему задачи.



Командами

`a=input()`

`b=input()`

`c=input()`

введём три числа, присвоив значения переменным a, b, c.

Условной конструкцией if-else проверим на истинность логическое выражение  $a < b$ . Если оно истинно, то переходим на проверку логического выражения  $a < c$ . Если оно истинно, то переменной "y" присвоим значение переменной "a", т.е. "a" будет минимальным, а иначе "y" присвоится значение переменной "c".

Если в начале логическое выражение  $a < b$  оказалось ложным, то переходим на проверку другого логического выражения  $b < c$ .

Если оно истинно, то "y" присвоится значение переменной "b", иначе "c". Командой `print()` выводим минимальное значение.

```

#нахождение минимального из 3-х чисел
a=input('Введите целое число \n')
b=input('Введите целое число \n')
c=input('Введите целое число \n')
if a<b:
    if a<c:
        y=a
    else:
        y=c
else:
    if b<c:
        y=b
    else:
        y=c
print('Минимальное:', y)
  
```

## Пример программы

```
Введите целое число  
2  
Введите целое число  
5  
Введите целое число  
1  
Минимальное: 1
```

Результат выполнения программы

## Задания для самостоятельной работы

### Задание

Даны три целых числа. Выбрать из них те, которые принадлежат интервалу[1,3].



## Практическое занятие № 34

Тема: Работа с циклами в Python.

Цель работы:

1. Познакомиться с циклическими конструкциями в Python.

Теоретические сведения:

В Python существуют два типа циклических выражений:

- Цикл while
- Цикл for

### 1. Цикл while в Python

Инструкция while в Python повторяет указанный блок кода до тех пор, пока указанное в цикле логическое выражение будет оставаться истинным.

Синтаксис цикла while:

**while** логическое выражение:

```
команда 1
команда 2
...
команда n
```

После ключевого слова while указывается условное выражение, и пока это выражение возвращает значение True, будет выполняться блок инструкций, который идет далее.

Все инструкции, которые относятся к циклу while, располагаются на последующих строках и должны иметь отступ от начала строки (4 пробела).

```
#!/ Программу по вычислению факториала
number = int(input("Введите число: "))
i = 1
factorial = 1
while i <= number:
    factorial *= i
    i += 1
print("Факториал числа", number, "равен", factorial)
```

### Пример программы на Python

```
Python 3.4.1 (v3.4.1:c0e311e010fc, May 18 2014, 10:38:22) [MSC v.1600 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ===== RESTART =====
>>>
Введите число: 5
факториал числа 5 равен 120
>>> |
```

Результат выполнения программы с использованием циклического оператора while

## 2. Цикл for в Python:

Цикл for в Python обладает способностью перебирать элементы любого комплексного типа данных (например, строки или списка).

Синтаксис цикла for:

```
for int in range():
```

```
    команда 1
```

```
    команда 2
```

```
    ...
```

```
    команда n
```

Переменной int присваивается значение первого элемента функции range(), после чего выполняются команды. Затем переменной int присваивается следующее по порядку значение и так далее до тех пор, пока не будут перебраны все элементы функции range().

Функция range() является универсальной функцией Python для создания списков (list) содержащих арифметическую прогрессию. Чаще всего она используется в циклах for.

range(старт, стоп, шаг) - так выглядит стандартный вызов функции range() в Python. По умолчанию старт равняется нулю, шаг единице.

### Пример. 1

Найти сумму n элементов следующего ряда чисел: 1 -0.5 0.25 -0.125 ... n. Количество элементов (n) вводится с клавиатуры. Вывести на экран каждый член ряда и его сумму. Решить задачу используя циклическую конструкцию for.

#### Решение:

В данном случае ряд чисел состоит из элементов, где каждый следующий меньше предыдущего в два раза по модулю и имеет обратный знак. Значит, чтобы получить следующий элемент, надо предыдущий разделить на -2.

Какой-либо переменной надо присвоить значение первого элемента ряда (в данном случае это 1). Далее в цикле добавлять ее значение к переменной, в которой накапливается сумма, после чего присваивать ей значение следующего элемента ряда, разделив текущее значение на -2. Цикл должен выполняться n раз.

```
n=int(input('Введите количество элементов последовательности: '))
x=1
s=0
print(x)
for i in range(n):
    s+=x
    x/=-2
    print(x)
print('Сумма ряда:',s)
```

Пример программы с циклом for

```
Введите количество элементов последовательности: 5
1
-0.5
0.25
-0.125
0.0625
-0.03125
Сумма ряда: 0.6875
```

### Результат выполнения программы

#### Задача 3

Дано целое число, не меньшее 2. Выведите его наименьший натуральный делитель, отличный от 1.

#### Решение:

Для начала введём целое число командой `int(input(текст сообщения))`. Затем зададим переменной `i` значение 2. Переменная `i` выполняет роль счётчика. Если задать ей значение 1, то условие задачи не будет выполнено, а результатом всегда будет 1.

В цикле `while` в качестве логического выражения используется команда `n%i` сравниваемая с нулём. Таким образом, если остаток от деления введённого числа на текущее значение `i` не равно нулю, то счётчик увеличивается на 1, а если равно нулю цикл заканчивается и командой `print()` выводится сообщение и значение `i`.

```
n = int(input('Введите целое число не меньше 2\n'))
i = 2
while n%i != 0:
    i+=1
print('наименьший натуральный делитель:', i)
```

### Пример программы с циклом while

```
Введите целое число не меньше 2
49
наименьший натуральный делитель: 7
```

### Результат выполнения программы

#### Задание для самостоятельной работы:

1. Дано вещественное число – цена 1 кг конфет. Вывести стоимость 1, 2, ...10 кг конфет. Решить задачу используя циклическую конструкцию `for`.

2. Дана непустая последовательность целых чисел, оканчивающаяся нулем. Найти: а) сумму всех чисел последовательности; б) количество всех чисел последовательности

Решить задачу используя циклическую конструкцию `while`.

## Практическое занятие № 35

Тема: Работа со строками в Python.

Цель работы:

1. Познакомиться методами работы со строками.

Теоретические сведения:

Строка — базовый тип представляющий из себя неизменяемую последовательность символов; str от «string» — «строка».

### Функции и методы работы со строками

Функция или метод	Назначение
<b>S1 + S2</b>	Конкатенация (сложение строк)
<b>S1 * 3</b>	Повторение строки
<b>S[i]</b>	Обращение по индексу
<b>S[i:j:step]</b>	Извлечение среза
<b>len(S)</b>	Длина строки
<b>S.join(список)</b>	Соединение строк из последовательности str через разделитель, заданный строкой
<b>S1.count(S[, i, j])</b>	количество вхождений подстроки s в строку s1. Результатом является число. Можно указать позицию начала поиска i и окончания поиска j
<b>S.find(str, [start],[end])</b>	Поиск подстроки в строке. Возвращает номер первого вхождения или -1
<b>S.index(str, [start],[end])</b>	Поиск подстроки в строке. Возвращает номер первого вхождения или вызывает ValueError
<b>S.rindex(str, [start],[end])</b>	Поиск подстроки в строке. Возвращает номер последнего вхождения или вызывает ValueError
<b>S.replace(шаблон, замена)</b>	Замена шаблона
<b>S.split(символ)</b>	Разбиение строки по разделителю
<b>S.upper()</b>	Преобразование строки к верхнему регистру
<b>S.lower()</b>	Преобразование строки к нижнему регистру

Ниже приведена программа, демонстрирующая использование функций и методов работы со строками.

```

example_string.py - K:\Лабораторные Python\example_string.py (3.7.1)
File Edit Format Run Options Window Help
s1="Пропаганда"
s2="Сенсация"
s3="Сенсация*Сенсация*Сенсация*Сенсация"
s4='ОхОхОхАх'
print('s1 = ',s1)
print('s2 = ',s2)
print('s3 = ',s3)
print('s4 = ',s4)
print('s1+s2 = ',s1+s2) #сложение двух строк
print('s1*3 = ',s1*3) #умножение строки на 3, т.е.строка выведется 3 раза
print('s1[2] = ',s1[2]) #вывод элемента строки s1 с индексом 2
print('s1[2,4] = ',s1[2:4]) #извлечение среза строки s1 начиная с индекса 2
#и заканчивая индексом 4
print('s3.count = ',s3.count(s2)) #количество вхождений подстроки s2 в S3,
#в результате выведется число
print('s1.find('a') = ',s1.find('a')) #поиск подстроки 'a' в строке s1
#результатом будет номер первого вхождения
print('s1.index('п') = ',s1.index('п'))#поиск подстроки 'п' в строке s1
#результатом будет номер первого вхождения
print('s1.rindex('д') = ',s1.rindex('д'))#поиск подстроки 'а' в строке s1
#возвращает номер последнего вхождения
print('s4.replace('Ох','Ах',2) = ',s4.replace('Ох','Ах',2))#замена шаблона. Строка 'Ох' - это шаблон
#строка 'Ах' - это замена
#в строке 4 последовательность 'Ох' будет заменена
#на 'Ах' с шагом 2
print('s3.split('*') = ',s3.split('*'))#разбиение по разделителю *
print('s1.upper = ',s1.upper())#перевод символов в верхний регистр
print('s1.lower = ',s1.lower())#перевод символов в нижний регистр
Ln: 20 Col: 40

```

Пример программы на Python

```

Python 3.7.1 Shell
File Edit Shell Debug Options Window Help
Python 3.7.1 (v3.7.1:260ec2c36a, Oct 20 2018, 14:05:16) [MSC v.191] on win32
Type "help", "copyright", "credits" or "license()" for more info
>>>
===== RESTART: K:\Лабораторные Python\example_string.py =====
s1 = Пропаганда
s2 = Сенсация
s3 = Сенсация*Сенсация*Сенсация*Сенсация
s4 = ОхОхОхАх
s1+s2 = ПропагандаСенсация
s1*3 = ПропагандаПропагандаПропаганда
s1[2] = о
s1[2,4] = оп
s3.count = 4
s1.find(a) = 4
s1.index(п) = 3
s1.rindex(д) = 9
s4.replace(Ох,Ах,2) = АхАхОхАх
s3.split(*) = ['Сенсация', 'Сенсация', 'Сенсация', 'Сенсация']
s1.upper = ПРОПАГАНДА
s1.lower = пропаганда

```

Результат выполнения программы с использованием функций и методов работы со строками

### Пример 1

Проверить, будет ли строка читаться одинаково справа налево и слева направо (т. е. является ли она палиндромом).

#### Решение

Сначала введём строку командой: `s=input('Введите строку ')`.

Затем определим логическую переменную `flag` и присвоим ей значение 1: `flag=1`.

Для начала в введённой строке нужно удалить пробелы. Для этого воспользуемся циклической конструкцией `for`, которая выполнится столько раз, какую имеет длину строка. Длину строки определим функцией `len(s)`.

В теле цикла будем проверять следующее условие: `s[i]!=' '`. Данное логическое выражение будет истинно в том случае, если  $i$ -ый элемент строки не будет равен пробелу, тогда выполнится команда следующая после двоеточия: `string+=s[i]`.

К строке `string`, которая была объявлена в начале программы, будет добавляться по символу строка `s`, но уже без пробелов.

Для проверки строки на "палиндром" воспользуемся циклической конструкцией `for`.

Длина половины строки находится делением нацело на 2. Если количество символов нечетно, то стоящий в середине не учитывается, т.к. его сравниваемая пара - он сам.

Количество повторов цикла равно длине половины строки. Длину строки определим функцией `len(s)`, где аргумент введённая нами строка `s`. Зная длину строки, можно вычислить количество повторов цикла. Для этого целочисленно разделим длину строки на 2: `len(s)//2`.

Для задания диапазона для цикла используем функцию `range()`, в которой аргументом будет являться половина длины строки: `range(len(s)//2)`.

```
for i in range(len(s)//2):
```

Если символ с индексом  $i$  не равен "симметричному" символу с конца строки (который находится путем индексации с конца)

```
if s[i] != s[-1-i],
```

то переменной `flag` присваивается значение 0 и происходит выход из цикла командой `break`.

Далее, при помощи условной конструкции `if-else` в зависимости от значения `flag` либо - 0, либо -1 выводится сообщение, что строка палиндром, либо нет.

```

s=input('Введите строку \n')
flag=1
string=''
for i in range(len(s)):
    if s[i]!=' ':
        string+=s[i]
print(string)
for i in range(len(s)//2):
    if string[i]!=string[-i-1]:
        flag=0
        break
if flag: print('Палиндром')
else: print('не палиндром')

```

### Пример программы на Python

```

Введите строку
а роза упала на лапу азора
арозаупаланалапуазора
Палиндром

```

### Результат выполнения программы

## Задания для самостоятельной работы (по вариантам)

### Вариант 1

Дана строка, содержащая русскоязычный текст. Найти количество слов, начинающихся с буквы "е".

### Вариант 2

В строке заменить все двоеточия (:) знаком процента (%). Подсчитать количество замен.

### Вариант 3

В строке удалить символ точку (.) и подсчитать количество удаленных символов.

### Вариант 4

В строке заменить букву(а) буквой (о). Подсчитать количество замен. Подсчитать, сколько символов в строке.

### Вариант 5

В строке заменить все заглавные буквы строчными.

### Вариант 6

В строке удалить все буквы "а" и подсчитать количество удаленных символов.

### Вариант 7

Дана строка. Преобразовать ее, заменив звездочками все буквы "п", встречающиеся среди первых  $n/2$  символов. Здесь  $n$  - длина строки.

**Вариант 8**

Дана строка, заканчивающаяся точкой. Подсчитать, сколько слов в строке.

**Вариант 9**

Определить, сколько раз в тексте встречается заданное слово.

**Вариант 10**

Дана строка-предложение на английском языке. Преобразовать строку так, чтобы каждое слово начиналось с заглавной буквы.

**Вариант 11**

Дана строка. Подсчитать самую длинную последовательность подряд идущих букв «н». Преобразовать ее, заменив точками все восклицательные знаки.

**Вариант 12**

Дана строка. Вывести все слова, оканчивающиеся на букву "я".

**Вариант 13**

Дана строка символов, среди которых есть одна открывающаяся и одна закрывающаяся скобки. Вывести на экран все символы, расположенные внутри этих скобок.

**Вариант 14**

Дана строка. Вывести все слова, начинающиеся на букву "а" и слова оканчивающиеся на букву "я".

**Вариант 15**

Дана строка текста. Подсчитать количество букв «т» в строке.



## Список литературы

### *Основные источники:*

1 Гаврилов, М. В. Информатика информационные технологии: учебник для СПО М. В. Гаврилов, В. А. Климов. изд., пер. и доп. — М.: Издательство Юрайт, 2018 г. 383 с. (Серия: Профессиональное образование) Режим доступа [www.biblioonline.ru/book/5964F5AF-7DB6-44A0-812B-527A36890184](http://www.biblioonline.ru/book/5964F5AF-7DB6-44A0-812B-527A36890184).

2 Советов Б.Я. Информационные технологии. учебник для СПО / Б.Я. Советов, В.В. Цехановский - 7-е изд., пер. и доп. — М. Издательство Юрайт, 2018 г., (Серия: Профессиональное образование). ISBN 978-5,534-06399-8. Режим доступа [www.biblio-online.mboowE5577F47-875445EA-8E5F-E8ECBC2E4731](http://www.biblio-online.mboowE5577F47-875445EA-8E5F-E8ECBC2E4731)).

3 Сергеева, И. И. Информатика: учебник / И.И. Сергеева, А.А. Музалевская, Н.В. Тарасова. — 2-е изд., перераб. и доп. — Москва: ФОРУМ: ИНФРА-М, 2020. — 384 с. — (Среднее профессиональное образование). - ISBN 978-5-8199-0775-7. — Текст: электронный. - URL: <https://znanium.com/catalog/product/1083063>

4 Гвоздева, В. А. Информатика, автоматизированные информационные технологии и системы: учебник / В.А. Гвоздева. — Москва: ИД «ФОРУМ» : ИНФРА-М, 2020. — 542 с. — (Среднее профессиональное образование). - ISBN 978-5-8199-0856-3. - Текст: электронный. - URL: <https://znanium.com/catalog/product/1067007>

### *Дополнительные источники:*

1 Плотникова, Н. Г. Информатика и информационно-коммуникационные технологии (ИКТ): учеб. пособие / Н.Г. Плотникова. — Москва: РИОР: ИНФРА-М, 2019. — 124 с. — (Среднее профессиональное образование). — <https://doi.org/10.12737/11561>. - ISBN 978-5-369-01308-3. - Текст: электронный. - URL: <https://znanium.com/catalog/product/994603>

2 Немцова, Т. И. Практикум по информатике. Компьютерная графика и web-дизайн: учебное пособие / Т.И. Немцова, Ю.В. Назарова; под ред. Л.Г. Гагариной. — Москва: ИД «ФОРУМ»: ИНФРА-М, 2020. — 288 с. + Доп. материалы [Электронный ресурс]. — (Среднее профессиональное образование). - ISBN 978-5-8199-0800-6. - Текст: электронный. - URL: <https://znanium.com/catalog/product/1073058>

3 Советов, Б. Я. Информационные технологии: учебник для среднего профессионального образования / Б. Я. Советов, В. В. Цехановский — 7-е изд., перераб. и доп. — Москва: Издательство Юрайт, 2020. — 327 с. — (Профессиональное образование). — ISBN 978-5-534-06399-8. — Текст: электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/450686>

***Интернет ресурсы:***

- 1 Web-сайт тексты для web-мастера.
- 2 Web-сайт создание webстраниц.
- 3 Web-сайт HTML.
- 4 [www.teachvideo.ru](http://www.teachvideo.ru) – Онлайн видео уроки.