

Управление образования администрации  
Ижморского муниципального округа  
Муниципальное бюджетное общеобразовательное учреждение  
«Ижморская средняя общеобразовательная школа №1»

Принята на заседании  
методического совета  
от 23 мая 2023г.  
Протокол №15

Утверждаю  
Директор организации  
Токарева И.И.  
23 мая 2023г.



**Дополнительная общеобразовательная общеразвивающая программа  
(центра образования цифрового и гуманитарного профилей «Точка роста»)  
технической направленности  
«Основы программирование в Python»**

Базовый уровень

Возраст учащихся: 10-13 лет

Срок реализации: 1 год

Разработчик:  
Нейбергер Н.В.,  
учитель информатики  
МБОУ «Ижморская СОШ №1»

## 1.1. Пояснительная записка

Дополнительная общеобразовательная общеразвивающая программа технической направленности «Основы программирования в Python» разработана на основе следующих нормативно-правовых документов:

1. Закон Российской Федерации «Об образовании» (Федеральный закон от 29 декабря 2012 г. № 273-ФЗ);
2. Приказ Минпросвещения России от 27.07.2022 № 629 «Об утверждении порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»;
3. Концепция развития дополнительного образования детей (Распоряжение Правительства РФ от 31 марта 2022 г. № 678-р);
4. Письмо Министерство просвещения РФ от 15 апреля 2022 года № СК-295/06 «Об использовании государственных символов Российской Федерации»
5. Письмо Министерства образования и науки РФ от 18.11.2015 № 09-3242 «Методические рекомендации по проектированию дополнительных общеразвивающих программ (включая разноуровневые программы);
6. Постановление Государственного санитарного врача РФ от 28.09.2020 г. СП 2.4. 3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи»;
7. Федеральный проект «Успех каждого ребенка» (протокол заседания проектного комитета по национальному проекту «Образование» от 07 декабря 2018 г. № 3);
8. Стратегия развития воспитания в Российской Федерации на период до 2025 года, (Распоряжение Правительства Российской Федерации от 29 мая 2015 г. № 996-р);
9. Устава МБОУ «Ижморская СОШ №1».

**Направленность программы.** Дополнительная общеобразовательная общеразвивающая программа «Основы программирования в Python» имеет техническую направленность.

**Актуальность программы** обусловлена широким распространением информационно-коммуникационных технологий в обществе и необходимостью обеспечивать связанную с этим инфраструктуру специалистами. Всё большее значение приобретает умение человека грамотно обращаться с компьютером, причём на уровне начинающего программиста.

Изучение основных принципов программирования невозможно без регулярной практики написания программ. Выбранный для обучения язык Python является достаточно эффективным и доступным инструментом достижения задач в области создания программных продуктов.

Синтаксис языка достаточно прост и интуитивно понятен, что понижает порог вхождения и позволяет сосредоточиться на алгоритмических аспектах программирования.

Кроме того, программа способствует ранней профессиональной ориентации учащихся.

**Педагогическая целесообразность** программы заключается в привлечении учащихся к занятиям техническим творчеством, что способствует развитию логического мышления, творческих способностей и навыков решения задач программирования. Так как занятия проводятся в формате «от простого к сложному», учащиеся вспоминают свои знания по основам алгоритмизации и программирования и на их основе, учатся составлять простые и сложные программы. Программирование мотивирует к занятиям в различных научных областях (физики, информатики, алгебры, геометрии и др.), развивает воображение и способствует ранней профориентации подростков.

**Отличительной особенностью** программы является её направленность на формирование у учащихся навыков поиска собственного решения поставленной задачи, составления алгоритма решения и его реализации с помощью средств программирования.

**Возраст учащихся**, участвующих в реализации данной программы: 13-16 лет.

**Количество учащихся в группе:** 10.

Режим - в неделю 1 занятие по 40 минут, часовая нагрузка 34 часа в год.

**Форма обучения** очная.

## **1.2. Цель и задачи программы**

**Цель программы:** формирование у учащихся базовых навыков разработки программ на языке программирования Python.

**Задачи программы:**

**Образовательные:**

- формировать у учащихся представление о роли информационных технологий в современном обществе;
- познакомить учащихся с возможностями компьютерных технологий в отношении обработки и представления графической и текстовой информации посредством написания программ;
- познакомить учащихся с основными алгоритмическими конструкциями и правилами их записи, с основными способами организации данных.
- научить учащихся составлять и записывать алгоритмы с использованием соответствующих алгоритмических конструкций, уметь распознавать необходимость применения той или иной алгоритмической конструкции при решении задач;
- научить учащихся разрабатывать алгоритмы и реализовывать их на языке программирования Python, осуществлять отладку и тестирование программы;

### **Развивающие:**

- развивать абстрактное и логическое мышление;
- развивать творческий подход к решению различных задач.

### **Воспитательные:**

- воспитывать самостоятельность;
- воспитывать культуру взаимодействия с другими людьми в условиях открытого информационного общества.

## **1.3. Содержание программы учебного курса**

### **Базовый уровень**

#### **Учебно-тематический план**

№ п/п	Наименование разделов и тем	Количество часов			Формы контроля
		Всего	Теория	Практика	
<b>1.</b>	<b>Языки программирования. Основные понятия и конструкции языков программирования на примере Python</b>	<b>4</b>	<b>2</b>	<b>2</b>	
<b>1.1</b>	Языки программирования. Обзор современных языков программирования.	1	1	0	Устный опрос
<b>1.2</b>	Язык программирования Python 3.x. Особенности ссылочных переменных. Блок-схемы. Команды ввода-вывода. Ветвления. Циклы.	3	1	2	Практическая работа
<b>2.</b>	<b>Синтаксис Python. Работа со стандартной библиотекой и с внешними библиотеками. Основы объектно-ориентированного программирования в Python</b>	<b>20</b>	<b>7,5</b>	<b>12,5</b>	
<b>2.1</b>	Разные способы ввода-вывода информации: генераторы. Решение задач.	2	0,5	1,5	Практическая работа
<b>2.2</b>	Ветвления: отличия. Решение задач.	2	0,5	1,5	Практическая работа
<b>2.3</b>	Циклы: виды, особенности, генераторы условий. Решение задач.	2	1	1	Практическая работа

2.4	Типы данных.	2	0,5	1,5	Практическая работа
2.5	Строки. Списки. Решение задач.	2	1	1	Практическая работа
2.6	Кортежи. Срезы. Решение задач. Промежуточная аттестация	2	1	1	Практическая работа, тест
2.7	Словари. Решение задач.	2	1	1	Практическая работа
2.8	Функции. Работа с функциями в Python. Решение задач.	2	1	1	Практическая работа
2.9	Работа со встроенным графическим исполнителем Черепашка.	2	0	2	Практическая работа
2.10	Работа с подключенным из модуля исполнителем Робот.	2	0	2	Практическая работа
<b>3.</b>	<b>Изучение принципов ООП Python. Технология разработки программ. Графические интерфейсы</b>	<b>5</b>	<b>2</b>	<b>3</b>	
3.1	Объекты. Теория объектов. Решение задач.	2	1	1	Практическая работа
3.2	Классы. Наследование свойств. Решение задач.	3	1	2	Практическая работа
<b>4.</b>	<b>Решение прикладных задач и создание приложений с графическим интерфейсом и использованием дополнительных библиотек Python. Профессиональная ориентация</b>	<b>5</b>	<b>2</b>	<b>3</b>	
4.1	Изучаем более совершенную графику с модулем Tkinter. Итоговая аттестация	5	2	3	Практическая работа, Итоговый тест
	<b>Всего часов</b>	<b>34</b>	<b>16</b>	<b>18</b>	

## **1. Языки программирования. Основные понятия и конструкции языков программирования на примере Python (4 часа).**

### ***Тема 1.1 Языки программирования. Обзор современных языков программирования.***

**Теория:** Проведение инструктажа по технике безопасности. Знакомство с понятием языка программирования. Обзор современных языков программирования.

### ***Тема 1.2 Язык программирования Python 3.x. Особенности ссылочных переменных. Блок-схемы. Команды ввода-вывода. Ветвления. Циклы.***

**Теория:** Знакомство с историей разработки языка программирования Python. Особенности ссылочных переменных. Блок-схемы. Команды ввода-вывода. Ветвления. Циклы.

**Практика:** Составление линейных, разветвляющихся, циклических алгоритмов с использованием блок-схем.

## **2. Синтаксис Python. Работа со стандартной библиотекой и с внешними библиотеками. Основы объектно-ориентированного программирования в Python (20 часа).**

### ***Тема 2.1 Разные способы ввода-вывода информации: генераторы. Решение задач.***

**Теория:** Знакомство с параметрами *sep*, *end* и различными вариациями их применения.

**Практика:** Разработка программ с применением параметров *sep*, *end* при выводе данных.

### ***Тема 2.2 Ветвления: отличия. Решение задач.***

**Теория:** Знакомство с операторами ветвления, выбора.

**Практика:** Разработка программ с применением операторов ветвления (нахождение корней квадратного уравнения), выбора (выбор наибольшего/наименьшего из чисел).

### ***Тема 2.3 Циклы: виды, особенности, генераторы условий. Решение задач.***

**Теория:** Знакомство с операторами циклов с условием (*while*) и с параметром (*for*).

**Практика:** Разработка программ с применением операторов *while* и *for*.

### ***Тема 2.4 Типы данных.***

**Теория:** Типы данных: целый тип, вещественный тип, логический тип.

**Практика:** Разработка программ с использованием переменных различного типа.

### ***Тема 2.5 Строки. Списки. Решение задач.***

**Теория:** Строки, работа с текстовой информацией. Списки, работа с числовой информацией.

**Практика:** Разработка программ с использованием строк и списков.

***Тема 2.6 Кортежи. Срезы. Решение задач. Промежуточная аттестация.***

**Теория:** Кортежи (неизменяемые списки). Основы работы со срезами и примеры их использования.

**Практика:** Разработка программ с использованием кортежей и срезов.

***Тема 2.7 Словари. Решение задач.***

**Теория:** Знакомство со словарями. Создание словарей, Операции со словарями.

**Практика:** Работа со словарями, использование различных методов: update(), get(), pop(), keys(), values(), items().

***Тема 2.8 Функции. Работа с функциями в Python. Решение задач.***

**Теория:** Функции в Python — синтаксис, аргументы, вызов, выход. Встроенные функции.

**Практика:** Работа с готовыми функциями. Разработка собственных функций.

***Тема 2.9 Работа со встроенным графическим исполнителем Черепашка.***

**Практика:** Работа со встроенным графическим исполнителем Черепашка. Подключение модуля. Работа с библиотекой модуля.

***Тема 2.10 Работа с подключенным из модуля исполнителем Робот.***

**Практика:** Работа с созданным исполнителем Робот. Создание исполнителя Чертежник. Работа с Чертежником. Создание собственного исполнителя. **Проект «Символика РФ, Кемеровской области, Ижморского района»**

**3. Изучение принципов ООП Python. Технология разработки программ.  
Графические интерфейсы (5 часов)**

***Тема 3.1 Объекты. Теория объектов. Решение задач.***

**Теория:** Объекты. Теория объектов.

**Практика:** Разработка программ с применением теории объектов

***Тема 3.2 Классы. Наследование свойств. Решение задач.***

**Теория:** Классы. Наследование свойств.

**Практика:** Разработка программ с применением классов. Оформление отдельных абзацев и символов.

**4. Решение прикладных задач и создание приложений с графическим интерфейсом и использованием дополнительных библиотек Python.  
Профессиональная ориентация (5 часов)**

***Тема 4.1 Изучаем более совершенную графику с модулем Tkinter.***

**Теория:** Знакомство с графическим с модулем Tkinter. Знакомство с объектом Canvas, его свойствами и методами.

**Практика:** Работа с объектом Canvas. Создание собственных графических объектов. Итоговая аттестация.

## 1.4. Планируемые результаты

### Базовый уровень

**В конце обучения** учащийся будет иметь следующие

#### *личностные результаты:*

- представление о современном языке программирования высокого уровня Python;
- синтаксис языка Python: основные инструкции языка программирования Python, списки, словари, строки, кортежи, генераторы функций; понимание блок-схем; представление о реализации анимации с помощью языка программирования;
- объекты, свойства и методы объектов; принцип построения программ «снизу-вверх» и «сверху-вниз»;
- решение простых прикладных задач;

#### *метапредметные результаты:*

- работа в любой среде разработчика, поддерживающей Python;
- подготовка программы к запуску;
- составление программ на языке программирования Python;
- создание анимированных изображений с помощью Python;
- работа в операционной системе на уровне пользователя;
- набор и редактирование текста на английском языке;
- создание простых приложений.

#### *Регулятивные УУД:*

- *определять и формулировать* цель деятельности на занятии с помощью учителя, а далее самостоятельно;
- *проговаривать* последовательность действий;
- уметь *высказывать* своё предположение (версию) на основе данного задания, уметь *работать* по предложенному учителем плану, а в дальнейшем уметь самостоятельно планировать свою деятельность;
- средством формирования этих действий служит технология проблемного диалога на этапе изучения нового материала;
- учиться совместно с учителем и другими воспитанниками *давать* эмоциональную *оценку* деятельности команды на занятии.

Средством формирования этих действий служит технология оценивания образовательных достижений (учебных успехов).

#### *Познавательные УУД:*

- добывать новые знания: *находить ответы* на вопросы, используя разные источники информации, свой жизненный опыт и информацию, полученную на занятии;



- перерабатывать полученную информацию: *делать* выводы в результате совместной работы всей команды;

Средством формирования этих действий служит учебный материал и задания.

*Коммуникативные УУД:*

- умение донести свою позицию до других: оформлять свою мысль. *Слушать* и *понимать* речь других;
- совместно договариваться о правилах общения и поведения в игре и следовать им;
- учиться выполнять различные роли в группе (лидера, исполнителя, критика).

Средством формирования этих действий служит организация работы в парах и малых группах.

*Оздоровительные результаты программы:*

- осознание учащимися необходимости заботы о своём здоровье и выработки форм поведения, которые помогут избежать опасности для жизни и здоровья, уменьшить пропуски занятий по причине болезни, регулярно посещать спортивные секции и спортивно-оздоровительные мероприятия;
- социальная адаптация детей, расширение сферы общения, приобретение опыта взаимодействия с окружающим миром.

## Раздел 2. Комплекс организационно-педагогических условий

### 2.1. Календарный учебный график

Реализация	Название темы	Кол-во часов
<b>1. Языки программирования. Основные понятия и конструкции языков программирования на примере Python</b>		
Сентябрь	Языки программирования. Обзор современных языков программирования.	1
Сентябрь	Язык программирования Python 3.x. Особенности ссылочных переменных. Блок-схемы. Команды ввода-вывода. Ветвления. Циклы.	3
<b>2. Синтаксис Python. Работа со стандартной библиотекой и с внешними библиотеками. Основы объектно-ориентированного программирования в Python</b>		
Октябрь	Разные способы ввода-вывода информации: генераторы. Решение задач.	2
Октябрь	Ветвления: отличия. Решение задач.	2
Ноябрь	Циклы: виды, особенности, генераторы условий. Решение задач.	2
Ноябрь	Типы данных: целый тип, вещественный тип.	2
Декабрь	Строки. Списки. Решение задач.	2
Декабрь	Кортежи. Срезы. Решение задач. Промежуточная аттестация	2
Январь	Словари. Решение задач.	2
Январь-февраль	Функции. Работа с функциями в Python. Решение задач.	2
февраль	Работа со встроенным графическим исполнителем Черепашка.	2
Февраль-март	Работа с подключенным из модуля исполнителем Робот.	2
<b>3. Изучение принципов ООП Python. Технология разработки программ. Графические интерфейсы</b>		
Март	Объекты. Теория объектов. Решение задач.	2
Апрель	Классы. Наследование свойств. Решение задач.	3
<b>4. Решение прикладных задач и создание приложений с графическим интерфейсом и использованием</b>		

<b>дополнительных библиотек Python. Профессиональная ориентация</b>		
Апрель-май	Изучаем более совершенную графику с модулем TKinter. Итоговая аттестация	5

## 2.2. Условия реализации программы

Для реализации программы используются следующие **методы обучения**:

- по источнику полученных знаний: словесные, наглядные, практические.
- по способу организации познавательной деятельности:
  - ✓ развивающее обучение (проблемный, проектный, творческий, частично-поисковый, исследовательский, программированный);
  - ✓ дифференцированное обучение (уровневые, индивидуальные задания).
  - ✓ игровые методы (конкурсы, игры-конструкторы, турниры с использованием мультимедиа, дидактические).

### **Средства обучения:**

- дидактические материалы (опорные конспекты, раздаточный материал для практических работ).
- методические разработки (презентации, видеоуроки, flash-ролики).

### **Материальное обеспечение программы**

1. Ноутбук (12 шт);
2. Доступ к сети Интернет.

### **Программное обеспечение:**

1. Сферум,
2. Интерпретатор Python версии 3.7 и выше;
3. IDE JetBrains PyCharm;

## 2.3. Форма аттестации

1. Промежуточная аттестация - **тест** на знание теории и практическая **задача**.
2. Итоговая аттестация – **тест** на знание теории и практическая **задача**.

### **Критерии оценивания результатов**

1. Тест на знание теории проводится на 30 ключевых вопросах, правильный ответ на которые оценивается 1 первичным баллом.
2. Практическая задача проверяется на компьютерных тестах. В зависимости от кол-ва пройденных тестов к результату теста по теории добавляется от 0 (нет пройденных тестов) до 5 (все тесты пройдены) баллов.

3. Итоговый первичный балл, набранный по сумме баллов тестовой и практической части переводится в проценты делением на 35.
4. При наборе от 70% и более ученик показывает высокий уровень подготовки по дисциплине.
5. При наборе от 50% до 69% уровень полученных знаний и умений считается базовым.
6. При наборе ниже 50% уровень подготовки ученика считается низким.

## 2.4. Методические материалы

<https://pythontutor.ru/>

<https://www.python.org/>

<https://pythonworld.ru/samouchitel-python>

<https://pythoner.name/>

## 2.5. Список литературы

### *для педагогов:*

1. Константин Поляков: Программирование. Python. C++. Часть 1. Учебное пособие – М.: Просвещение/Бином, 2021. – 144 с.
2. Константин Поляков: Программирование. Python. C++. Часть 2. Учебное пособие – М.: Просвещение/Бином, 2021. – 176 с.
3. Константин Поляков: Программирование. Python. C++. Часть 3. Учебное пособие – М.: Просвещение/Бином, 2020. – 208 с.
4. Константин Поляков: Программирование. Python. C++. Часть 4. Учебное пособие – М.: Просвещение/Бином, 2020. – 192 с.
5. Банкрашков, А.В. Программирование для детей на языке Python - М.: АСТ, 2018. - 288 с.
6. Вордерман, К. Программирование на Python. Иллюстрированное руководство для детей - М.: Манн, Иванов и Фербер, 2017. - 346 с.

### *для учащихся:*

1. МакГрат, М. Программирование на Python для начинающих - М.: Эксмо, 2015. - 192 с.
2. Мэтиз, Э. Изучаем PYTHON. Программирование игр, визуализация данных, веб-приложения - СПб.: Питер, 2017. - 496 с.
3. Саммерфилд, М. Программирование на Python 3. Подробное руководство. - М.: Символ-Плюс, 2011. - 608 с.
4. Саммерфилд, М. Программирование на Python 3. Подробное руководство - М.: Символ, 2016. - 608 с.
5. Саммерфилд, М. Программирование на Python 3. Подробное руководство - СПб.: Символ-плюс, 2015. - 608 с.

**Оценочные материалы****КИМ для промежуточной аттестации****Тест «Язык Python 3»**

**Задание: Продолжите каждое предложение по смыслу. За каждый правильный ответ вы набираете 1 балл.**

1. Программа на Python называется ...
2. Расширение файла Python – ...
3. Переменная в Python – это ...
4. Регистр букв в идентификаторах значение ...
5. Выражение в Python – это ...
6. Символ # в Python обозначает ...
7. ... в Python это тип данных для вещественных чисел, встроенный в Python по умолчанию.
8. Операция  $3 ** 4$  – это
9.  $345$  – данные ... типа.
10. Операция  $46 \% 10$  – это ...
11. Функция `round(d)` – это ...
12. Функция `input()` – предназначена для ...
13. Для вывода данных в Python есть функция - ...
14. ... в Python - это логический тип данных, встроенный в Python по умолчанию.
15. Строки – это ...
16.  $A = 'pri', s = 'vet'. A + s$  – это ...
17.  $E = 'no', \text{print}(E * 5)$  выведет на экран ...
18. К элементу в строке можно обратиться по ...
19.  $S = 'asdfgh'$   
`print(s[-1])`. Программа выведет ...
20.  $S = 'asdfgh'$   
`print(s[2:4])`. Программа выведет ...
21. Функция `len(S)` – возвращает ...
22. Списки – это ...
23. Пример списка - ...
24. Словари – это ...
25. Пример словаря - ...
26. Условный оператор в Python - ...
27. Цикл `for` называется циклом ...
28. Переведите конструкцию языка  
 $S = [1, 2, 3]$

```
for i in S:  
    Print (i*4)
```

29. Функция range() переводится как ...

30. Переведите конструкцию языка

```
S = 0  
while S < 10:  
    print (S)  
    S += 1
```

### Ответы:

1. Скрипт
2. Py
3. имя/идентификатор, который может принимать некоторое значение.
4. Имеет
5. это фрагмент языка программирования, представляющий способ вычисления некоторого значения.
6. Комментарий
7. Float
8. Возведение в степень
9. Целочисленный, int
10. Остаток от деления
11. Округление числа
12. Ввода данных в строку
13. Print()
14. Bool
15. Упорядоченные неизменяемые последовательности символов, используемые для хранения и представления текстовой информации
16. Объединение, сложение строк. Конкатенация
17. Повторение строки 5 раз. Дублирование
18. Индексу
19. h
20. dfg
21. длину строки
22. изменяемая последовательность произвольных объектов.
23. C=[2,3,4.5,'gh']
24. Изменяемые неупорядоченные коллекции произвольных объектов с доступом по ключу
25. K={1:'a',2:'b',3:'c'}
26. If
27. Обхода
28. для каждого элемента I в списке s делать следующее (то, что в теле цикла)

29.диапазон

30.пока условие истинно, то выполняется инструкция, после чего условие проверяется снова и снова выполняется инструкция. Так продолжается до тех пор, пока условие будет истинно, в противном случае мы выйдем из цикла.

### Задача на промежуточную аттестацию

Ограничение времени	1 секунда
Ограничение памяти	64Mb
Ввод	стандартный ввод или input.txt
Вывод	стандартный вывод или output.txt

Ленты облаков, расшитые серебряными нитями, стягивались к Краедуге, где, увлекаемые вращением мира, свертывались в стомильные бигуди. Два помела, взрыхляя туман, оставляли за собой клубящиеся туннели пара, так что наблюдающие за этим полетом боги — которые наверняка наблюдали, поскольку им просто нечем больше заниматься — явились свидетелями грандиозного окучивания неба.

Поднявшись на тысячу футов и войдя в ледяную полосу атмосферы, ведьмы вновь принялись спорить. Когда спорят ведьмы, слова разлетаются фейерверком. Даже богам опасно вставать у них на пути. Слово ведьмы, да еще сказанное в горячке спора, может обладать бронебойной силой.

Напишите программу, выбирает слова по определенному правилу.

#### Формат ввода

Строка слов, разделенных символами `<&>`.

Строка слов, разделенных символами `-{-}`.

#### Формат вывода

Для каждого слова из первой строки нужно записать список слов из второй строки, в которых есть ровно 2 общие буквы с заданным. Одинаковые буквы в слове считаются за одну. Порядок вывода строк и слов в списке не важен. Слова в списке записываются через запятую и пробел. Если таких слов не оказалось, вывести **нет слов**.

#### Пример 1

Ввод	Вывод
туман<&>метла<&>слово земля-{}-камни-{}-твердый-{}-мягкий-{}-удар-{}- черт	туман: удар метла: камни, твердый, черт слово: нет слов

## Пример 2

Ввод	Вывод
пейзаж<&>красота<&>ландшафт сон-{}-ярость-{}-суровый-{}-пик-{}-долина	пейзаж: нет слов красота: сон, долина ландшафт: нет слов

## КИМ для итоговой аттестации

### Тест «Язык Python3»

**Задание: Продолжите каждое предложение по смыслу.** За каждый правильный ответ вы набираете 1 балл.

1. Программа на Python называется ...
2. Расширение файла Python – ...
3. Переменная в Python – это ...
4. Регистр букв в идентификаторах значение ...
5. Выражение в Python – это ...
6. Символ # в Python обозначает ...
7. ... в Python это тип данных для вещественных чисел, встроенный в Python по умолчанию.
8. Операция  $3 ** 4$  – это
9.  $345$  – данные ... типа.
10. Операция  $46 \% 10$  – это ...
11. Функция `round(d)` – это ...
12. Функция `input()` – предназначена для ...
13. Для вывода данных в Python есть функция - ...
14. ... в Python - это логический тип данных, встроенный в Python по умолчанию.
15. Строки – это ...
16. `A = 'pri', s = 'vet'`. `A + s` – это ...
17. `E = 'no', print(E * 5)` выведет на экран ...
18. К элементу в строке можно обратиться по ...
19. `S = 'asdfgh'`  
`print(s[-1])`. Программа выведет ...
20. `S = 'asdfgh'`



- `print(s[2:4])`). Программа выведет ...
21. Функция `len(S)` – возвращает ...
  22. Область видимости функции – это ...
  23. Глобальная переменная - это ...
  24. Словари – это ...
  25. Пример словаря - ...
  26. Условный оператор в Python - ...
  27. Цикл `for` называется циклом ...
  28. Переведите конструкцию языка  
`S = {1, 2, 3}`  
`for I not in S:`  
`S.add(i)`
  29. Функция `round()` переводится как ...
  30. Переведите конструкцию языка  
`S = { }`  
`for x in input().split():`  
`s[x[0]] = x[1]`

### Ответы:

1. Скрипт
2. Py
3. имя/идентификатор, который может принимать некоторое значение.
4. Имеет
5. это фрагмент языка программирования, представляющий способ вычисления некоторого значения.
6. Комментарий
7. Float
8. Возведение в степень
9. Целочисленный, `int`
10. Остаток от деления
11. Округление числа
12. Ввода данных в строку
13. `Print()`
14. `Bool`
15. Упорядоченные неизменяемые последовательности символов, используемые для хранения и представления текстовой информации
16. Объединение, сложение строк. Конкатенация
17. Повторение строки 5 раз. Дублирование
18. Индексу

- 19.h
- 20.dfg
- 21.длину строки
- 22.рамки подпрограммы или программы, где работает та или иная переменная
- 23.переменная, видимая из всех частей программы
- 24.Изменяемые неупорядоченные коллекции произвольных объектов с доступом по ключу
- 25. $K = \{1:'a', 2:'b', 3:'c'\}$
- 26.If
- 27.Обхода
- 28.для элемента, не входящего в множество  $S$ , добавить его в множество  $S$
- 29.математическое округление
- 30.для элемента, введенного клавиатуры и разделенного на слова по пробелу, добавить в словарь по ключу – первому слову его значение – второе слово

### Критерии оценивания:

№ п/п	Количество набранных баллов	Оценка
1	27-30	Высокий уровень
2	20-26	Базовый уровень
3	до 19	Низкий уровень

### Задача на итоговую аттестацию

Ограничение времени	1 секунда
Ограничение памяти	64Mb
Ввод	стандартный ввод или input.txt
Вывод	стандартный вывод или output.txt

Напишите функцию **export\_check(text)**, которая принимает на вход текст, где каждая строка состоит из:

1. названия товара (строка)
2. цены за единицу товара (вещественное число)
3. количества единиц товара (целое число)

(Значения внутри строки разделены символом табуляции).  
 Ваша программа должна обработать полученные данные и создать документ Excel **res.xlsx**, в котором будут следующие столбцы:  
**«Товар», «Цена за единицу товара», «Количество товара», «Общая стоимость товара»** (названия столбцов выводить не надо)

В столбце «**Общая стоимость товара**» должна содержаться формула, которая рассчитывает общую стоимость товара как произведение цены и количества товара (с использованием простого перемножения ячеек)

Последней строкой должно идти «**Итого**», с пустыми значениями в столбцах «**Цена за единицу товара**» и «**Количество товара**». В столбце «**Общая стоимость товара**» должна содержаться формула, которая считает общую сумму покупок. (с использованием функции Excel СУММ)

Для разделения товаров разных чеков используется строка «---». В документе **res.xlsx** информация о каждом чеке должна располагаться на отдельном листе. При дублировании товара (совпадении названия и цены за единицу) в чеке в **res.xlsx** должна создаваться только одна строка с общим количеством купленного товара.

Товары в чеке должны быть отсортированы по алфавиту.

### **Формат ввода**

Многострочный текст в параметре **text**.

### **Формат вывода**

Файл **res.xlsx**, содержащий обработанную информацию.