

**Индивидуальный предприниматель
Селендеева Оксана Николаевна**

УТВЕРЖДАЮ

Индивидуальный предприниматель

_____/Селендеева О.Н./

**ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА
ДОПОЛНИТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА
«ПРОГРАММИРОВАНИЕ НА PYTHON 3»**

Москва, 2023

Оглавление

1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА	2
1.1. Общая характеристика программы	2
1.2. Цели и задачи программы	3
1.3. Планируемые результаты обучения	4
2. УЧЕБНЫЙ ПЛАН ПРОГРАММЫ	5
3. КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК	7
4. РАБОЧАЯ ПРОГРАММА	8
5. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ	11
5.1. Контроль знаний, умений и навыков	11
5.2. Критерии оценивания освоения программы при проведении различных форм контроля:	12
6. УЧЕБНО-МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ (УЧЕБНИКИ, РАЗДАТОЧНЫЕ МАТЕРИАЛЫ, МЕТОДИЧЕСКИЕ УКАЗАНИЯ, ПЛАКАТЫ, СЛАЙДЫ, ИНТЕРНЕТ- РЕСУРСЫ)	13
6.1. Информационные и учебно-методические условия реализации программы	13
6.2. Рекомендованная литература для обучающихся:	13
6.3. Перечень ресурсов информационно-телекоммуникационной сети "Интернет" (далее- сеть "Интернет"), рекомендованных для освоения программы:	14
7. ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИЕ УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ ..	14
7.1. Материально-техническая и ресурсная база	14
7.2. Кадровое обеспечение программы	14

1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

1.1. Общая характеристика программы

Данный документ описывает комплекс основных характеристик образования (объем, содержание, планируемые результаты) и организационно-педагогических условий, который представлен в виде учебного плана, календарного учебного графика, рабочих программ учебных предметов, модулей, иных компонентов, а также оценочных и методических материалов общеобразовательной общеразвивающей программы "Программирование на Python 3".

В ходе дополнительной общеобразовательной общеразвивающей программы "Программирование на Python 3" слушатели познакомятся с основными концепциями программирования, освоят язык программирования Python 3. В ходе курса ученики получат активный навык самостоятельного создания программ различного назначения степени сложности на языке программирования Python 3.

Выдача обучающимся документов о дополнительном образовании (сертификат установленного образца) осуществляется при условии успешного прохождения итоговой аттестации.

Программа разработана на основе следующих **нормативных документов**:

- Федеральный закон от 29 декабря 2012 г. №273-ФЗ «Об образовании в Российской Федерации»;
- Приказ Министерства просвещения РФ от 9 ноября 2018 г. N 196 "Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам"
- Постановление Главного государственного санитарного врача РФ от 28.09.2020 № 28 «Об утверждении санитарных правил СП 2.4.3648-20 "Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи" (вместе с "СП 2.4.3648-20. Санитарные правила...")» (Зарегистрировано в Минюсте России 18.12.2020 № 61573).

Направленность (профиль) программы: техническая

Актуальность программы, соответствие государственному, социальному заказу/запросам.

В современном мире умение программировать является **ценным и востребованным навыком**, который все больше и больше ценится на рынке, особенно сегодня, в условиях ограниченных ресурсов и повышенных расходов. В результате прохождения обучения слушатель сформирует знания в области основ программирования, разовьет логическое и алгоритмическое мышление, активизирует навыки использования полученных знаний и умений в практической деятельности, проявит интерес к науке, фантазии, изобретательности. Обучающимся будут созданы оптимальные условия для всестороннего удовлетворения потребностей и развития их индивидуальных склонностей и способностей, появится мотивация личности к познанию и творчеству. Обучение по данной программе будет полезно

для тех, кто интересуется развитием в области научно-технической направленности, будущих программистов, аналитиков, инженеров.

Отличительные особенности программы: На занятиях обучающиеся решают множество различных задач для развития активных практических навыков программирования. В процессе освоения программы обучающиеся смогут в раннем возрасте получить профессиональную ориентацию по направлению Python-разработки.

Срок обучения: программа реализуется в объеме 32 академических часов, 12 недель (3 месяцев).

Режим занятий: 3-4 академических часа в неделю

Продолжительность академического часа – 45 минут.

Занятия начинаются не ранее 9.00 часов утра и заканчиваются не позднее 20.00 часов. Для обучающихся в возрасте 16-18 лет допускается окончание занятий в 21.00 часов.

Продолжительность занятий в учебные дни - не более 3-х академических часов в день, в выходные и каникулярные дни - не более 4 академических часов в день. После 30-45 минут теоретических занятий организуется перерыв длительностью не менее 10 мин.

Адресат программы и примерный портрет слушателя курсов: программа разработана для учащихся от 11 до 16 лет, которых интересует программирование на языке Python. К освоению дополнительной общеобразовательной программы – дополнительной общеразвивающей программы допускаются: лица без предъявления требований к уровню образования.

По завершении реализации программы, как правило, проводится анкетирование обучающихся с целью изучения мнения по вопросу эффективности и информативности проведенного обучения, уровню организации учебного процесса, удовлетворенности учебно-методическим материалом, работниками образовательной организации проводится анализ высказанных предложений и пожеланий.

1.2. Цели и задачи программы

Цель программы – получение обучающимися знаний по основным концепциям программирования, а также навыков по созданию программ на языке Python.

Задачи программы:

1. Сформировать системные знания в области компьютерных технологий и программирования.
2. Сформировать навыки программирования на языке Python 3.
3. Предоставить специализированные практические навыки написания программ на Python 3.
4. Дать представление о профессии Python-разработчика.
5. Научить создавать собственные программы для решения различных задач.
6. Сформировать интерес к увлечению программированием и раскрытию своих способностей в сфере IT-технологий.

1.3. Планируемые результаты обучения

По итогам освоения дополнительной общеобразовательной программы - дополнительной общеразвивающей программы "Программирование на Python 3" обучающиеся должны будут овладеть следующими знаниями, умениями и навыками:

Знать:

- основы языка Python
- основные концепции и понятия программирования
- типы данных в Python
- правила создания и комментирования кода на Python
- виды условных конструкций в Python
- виды циклов в Python
- способы работы со строками и кортежами в Python
- понятие функции
- понятие объектно-ориентированного программирования
- основные методы работы с классами
- понятие библиотек в программировании
- основные этапы разработки, тестирования и запуска в работу программного обеспечения.

Уметь:

- писать собственные скрипты и программы на Python
- тестировать и отлаживать созданные программы
- создавать программы с использованием условных конструкций
- создавать программы с использованием циклов
- применять комментарии в программах
- создавать программы с использованием операторов из модуля turtle
- использовать в программах списки и кортежи
- создавать функции в программах
- выполнять действия по работе с файлами в программах
- использовать в своих программах классы
- подключать библиотеки к своим программам
- использовать в программах функции
- работать в команде на результат
- создавать и защищать свои проекты по разработке программного обеспечения
- применять развитое алгоритмическое и креативное мышление при решении задач программирования
- проводить техническую презентацию созданных проектов.

Владеть навыками в области:

- Разработки программного обеспечения на языке Python
- Использования различных типов данных в программировании
- Объектно-ориентированного программирования
- Применения классов и функций в программах
- Тестирования и отладки созданных программ
- Подготовки презентаций проектов и защиты проектов.

2. УЧЕБНЫЙ ПЛАН ПРОГРАММЫ

В процессе преподавания курса «Программирование на Python 3» используются как классические методы обучения (лекции), так и различные виды практической работы обучающихся по заданию преподавателя, которые направлены на развитие навыков разработки программного обеспечения, креативных качеств и на поощрение интеллектуальных инициатив учащихся.

Формы организации образовательного процесса (индивидуальные, групповые и т.д.) и другие виды занятий по программе определяются содержанием программы. Образовательная деятельность обучающихся предусматривает следующие виды учебных занятий и учебных работ: лекции, практические и семинарские занятия, мастер-классы, тренинги, семинары по обмену опытом, проведение открытых занятий, консультации и другие виды учебных занятий и методической работы, определенные учебным планом.

№	Наименование дисциплины/раздела/темы	Количество академических часов				Форма аттестации /контроля
		Всего	в т.ч. аудиторных		СРС,	
			теория	практич. занятия		
1	Знакомство с языком программирования Python 3	2	1	0,5	0,5	Практическое задание
2	Вычисления в Python	2	0,5	1	0,5	Практическое задание
3	Условные операторы в Python 3	2	1	0,5	0,5	Практическое задание
4	Циклы в Python. Цикл for	4	1	1,5	1,5	Практическое задание, Контрольные вопросы
5	Циклы в Python. Цикл while	2	0,5	1	0,5	Практическое задание
6	Строки и списки. Функции в Python 3	3	1	1	1	Практическое задание
7	Список и кортеж	3	1	1	1	Практическое задание
8	Работа с файлами и строками	3	1	1	1	Практическое задание, Контрольные вопросы
9	Сложные типы данных. Словари	2	0,5	0,5	1	Практическое задание
10	Классы в Python 3	3	1	1	1	Практическое задание
11	Работа с классами и	2	1	0,5	0,5	Практическое

	функциями. Закрепление материала					задание
12	Решение практических задач. Итоги курса. Презентация проектов	4	0,5	1,5	2	Практическое задание. Контрольные вопросы. Защита проекта.
	ИТОГО	32	10	11	11	

3. КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК

Календарный учебный график формируется при осуществлении обучения в течение всего календарного года. По мере набора групп слушателей по программе составляется календарный график, учитывающий объемы лекций, практики, самоподготовки, другие формы организации занятий.

Темы / недели	1	2	3	4	5	6	7	8	9	10	11	12	Итого часов
1. Знакомство с языком программирования Python 3	2												2
2. Вычисления в Python		2											2
3. Условные операторы в Python 3			2										2
4. Циклы в Python. Цикл for				3,5									3,5
Промежуточный контроль				0,5									0,5
5. Циклы в Python. Цикл while					2								2
6. Строки и списки. Функции в Python 3						3							3
7. Список и кортеж							3						3
8. Работа с файлами и строками								2,5					2,5
Промежуточный контроль								0,5					0,5
9. Сложные типы данных. Словари									2				2
10. Классы в Python 3										3			3
11. Работа с классами и функциями. Закрепление материала											2		2
12. Решение практических задач. Итоги курса. Презентация проектов												3	3
Итоговая аттестация												1	1
ИТОГО	2	2	2	4	2	3	3	3	2	3	2	4	32

4. РАБОЧАЯ ПРОГРАММА

1. Знакомство с языком программирования Python 3

- Что такое программирование;
- Преимущества и простота программирования на Python3;
- Обзор IDE (Интегрированная среда разработки);
- Простые переменные в Python3
- Практика написания первых команд.

Практическое задание: написание первых программ на языке Python 3.

2. Вычисления в Python

- Типы данных в Python 3;
- Арифметические и логические операции;
- Преобразование типов данных;
- Логические операторы.

Практическое задание: написания программ по решению математических задач на Python.

3. Условные операторы в Python 3

- Что такое условный оператор;
- Понятие блока кода;
- Операторы if, else, elif;
- Логические операторы and, or и not.

Практическое задание: написать не менее 5 программ с использованием переменных, условных конструкций и логических операторов.

4. Циклы в Python. Цикл for

- Циклы в программировании. Цикл for;
- Функция range;
- Комментирование кода;
- Библиотека turtle()
- Вложенные циклы.

Практическое задание: решение задач с применением циклов, функции range(), рисование фигур с использованием библиотеки turtle.

5. Циклы в Python. Цикл while

- Знакомство с циклом while;
- Отличия конструкций циклов и сравнений;
- Прерывание цикла while;
- Практика программирования.

Практическое задание: написание программ Python с использованием различных видов циклов и условных конструкций.

6. Строки и списки. Функции в Python 3

- Тип данных строка. Базовые операции со строками;
- Списки и операции над списками;
- Понятие "функции" в программировании;
- Примеры функций. Команда return.

Практическое задание: написания программ с применением списков, строк, функций. Создание программы "камень-ножницы-бумага".

7. Список и кортеж

- Концепция массива в программировании;
- Списки как массивы;
- Понятие кортежа. Операции над кортежами;
- Методы кортежа.

Практическое задание: написания программ с применением различных видов списков.

11. Работа с файлами и строками

- Способы форматирования строк. f-строки;
- Чтение и запись в файл;
- Команды для р;
- Логические операторы and, or и not.

Практическое задание: написать не менее 5 программ на работу со строками и действия с файлами.

9. Сложные типы данных. Словари

- Словари и множества в Python;
- Отличие списков и словарей;
- Методы работы со словарями;
- Практика программирования - решение задач с использованием словарей.

Практическое задание: поиск алгоритмов и написание кода для решения задач по теме "словари" и "списки".

10. Классы в Python 3

- Введение в объектно-ориентированное программирование;
- Концепция классов. Методы классов;
- Атрибуты объектов;
- Практика: написание программ с использованием классов и библиотеки turtle().

Практическое задание: создание программ с использованием библиотеки turtle и применением классов.

11. Работа с классами и функциями. Закрепление материала

- Повторение свойств class и def;
- Библиотеки в программировании;
- Основные этапы создания программы;
- Поэтапное создание программы "Прыгающий мяч".

Практическое задание: написание программы "Прыгающий мяч" с использованием библиотеки turtle.

12. Решение практических задач. Итоги курса

- Знакомство с библиотеками tkinter и gtt;
- Подготовка к презентации созданных программ;
- Дальнейшее развитие и обучение в сфере Python-разработки;
- Подведение итогов курса.

Практическое задание: подготовка выступления и презентация своего проекта.

5. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

5.1. Контроль знаний, умений и навыков

Формой подведения итогов реализации дополнительной образовательной программы "Программирование на Python 3" выступает текущая, промежуточная и итоговая аттестация. Образовательный процесс осуществляется на основании учебного плана и регламентируется расписанием занятий для каждой учебной группы.

В целях оценки показателей знаний, умений и навыков обучающихся по дополнительной образовательной общеразвивающей программе "Программирование на Python 3" проводится текущий и промежуточный контроль знаний, а также итоговая аттестация.

Виды текущего контроля:

- устный ответ на поставленный вопрос;
- проверка результатов выполнения практических заданий

Виды промежуточного контроля:

- тестирование устное/письменное/с помощью электронных форм
- проверка результатов выполнения практических работ/проектов по итогам учебного модуля.

Тестирование - это форма измерения знаний обучающихся, основанная на применении тестов. Материалы для промежуточного и итогового тестирования предоставляются вместе с комплектом учебно-методических материалов к программе.

Итоговая аттестация

Итоговая аттестация проводится с целью установления уровня знаний обучающихся с учетом прогнозируемых результатов обучения и требований к результатам освоения образовательной программы.

Итоговая аттестация обучающихся осуществляется в форме презентации созданных в курсе проектов – ботов различной тематики, которая сопровождается демонстрацией проекта и презентацией. Презентация – это электронный документ, предназначенный для визуальной демонстрации выполненной работы. Как правило, презентация имеет сюжет, сценарий и структуру, созданную для удобного восприятия информации.

Выдача обучающимся документов о дополнительном образовании (сертификат о прохождении курса) осуществляется при условии успешного прохождения итоговой аттестации.

5.2. Критерии оценивания освоения программы при проведении различных форм контроля:

Тестирование (Приложение 1. Примерные вопросы для промежуточного тестирования).
Процент результативности (правильных ответов при выполнении тестовых заданий):

Выполнение теста	Итоговая оценка
70% и более правильных ответов	"Зачтено"
Менее 70% правильных ответов	"Не зачтено"

Проверка выполнения практических работ (Приложение 2. Примерные задания для проверки усвоения качества учебного материала). Система оценивания:

"Зачтено" – необходимый уровень выполнения задания достигнут, обучающийся демонстрирует хорошее знание теоретической и практической части материала занятия/учебного модуля, достигнуты промежуточные и/или итоговые результаты работы над заданием.

"Не зачтено" - необходимый результат/уровень освоения не достигнут, обучающийся не усвоил теоретические основы и/или изученные практические приемы и методы разработки, инструменты создания чат-ботов, не достиг промежуточных и итоговых результатов при выполнении задания.

Проверка результатов демонстрации проектов на итоговой аттестации:

Критерии оценки	БАЛЛЫ
Обоснование выбора темы и ее актуальность	0-2 балла
Портфолио с решенными задачами	0-4 балла
Применение изученных программных средств, техническая сложность программ	0-3 балла
Креативность идеи	0-3 балла
Самостоятельность работы над проектом	0-4 балла
Компетентность докладчика (ответы на вопросы)	0-3 балла
Итоговая оценка: «Не зачтено» «Зачтено»	0-12 баллов 13-20 баллов

6. УЧЕБНО-МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ (УЧЕБНИКИ, РАЗДАТОЧНЫЕ МАТЕРИАЛЫ, МЕТОДИЧЕСКИЕ УКАЗАНИЯ, ПЛАКАТЫ, СЛАЙДЫ, ИНТЕРНЕТ-РЕСУРСЫ)

6.1. Информационные и учебно-методические условия реализации программы

Учебно-методический комплект

Для реализации целей и задач обучения по используется общеобразовательной общеразвивающей программы "Программирование на Python 3" используется комплект материалов преподавателя, который включает:

1. Текстовое методическое пособие с описанием целей, результатов каждого занятия, теоретического материала и практических работ.
2. Видеоурок для преподавателя с методическими указаниями и порядком объяснения учебного материала.
3. Раздаточный материал для учащихся - описание дополнительной самостоятельной работы учащихся по каждому занятию с примерами и рекомендациями по выполнению.
4. Описание мероприятий по контролю знаний – тестовые вопросы, практические задания.
5. Рекомендации по проведению итоговой аттестации и защиты проектов.
6. Дополнительные материалы – примеры кодов реализованных ботов, скрипты.
7. Дополнительные материалы – инструкции по установке необходимого программного обеспечения, описание технических требований к компьютерному оборудованию.

Материалы преподавателя размещаются на учебном портале преподавателей, размещенном на сервере информационно-телекоммуникационной сети "Интернет", и доступны по ссылке для всех преподавателей курса. Материалы обучающихся раздаются в печатном виде или рассылаются преподавателем индивидуально каждому обучающемуся.

6.2. Рекомендованная литература для обучающихся:

Основная:

1. Бриггс Джейсон. Python для детей. Самоучитель по программированию / Д.Бриггс. - Москва: Манн Иванов и Фербер (МИФ), 2016. – 289 с.
2. Грацианова Т.Ю. Информатика. Программирование в примерах и задачах / Т.Ю. Грацианова, Москва: ВМК МГУ (Лаборатория знаний), 2020. – 393 с.
3. Брайсон Пейн. Программирование на Python для детей и родителей / Б. Пейн. – Москва: Эксмо, 2017 – 354 с.

Дополнительная:

5. Воронин, И. Программирование для детей. От основ к созданию роботов / И. Воронин, В. Воронина. - СПб.: Питер, 2019. - 304 с.
6. Стоуэл Л., Программирование для детей на языке Python / Л. Стоуэл, Москва: Издательство АСС, 2018. – 98 с.

6.3. Перечень ресурсов информационно-телекоммуникационной сети "Интернет" (далее - сеть "Интернет"), рекомендованных для освоения программы:

- <https://pythonru.com/uroki/vvedenie-uroki-po-python-dlja-nachinajushhih> - PythonRu — Уроки по Python для начинающих
- <https://pythontutor.ru/> - проект Питонтьютор
- https://ru.hexlet.io/courses/python_101 – Хекслет. Курс «Введение в Python»
- <https://www.udemy.com/course/python-project/> - Python. Практика для начинающих
- http://window.edu.ru/catalog/resources?p_rubr=2.1.6 – единое окно доступа к образовательным ресурсам, раздел "Информатика и ОКТ"
- <https://www.python.org/downloads/windows/> - официальный сайт Python

7. ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИЕ УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ

7.1. Материально-техническая и ресурсная база

Для реализации программы предполагается использование учебных аудитории для проведения занятий лекционного типа, занятий семинарского типа, выполнения проектных работ, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации:

- Компьютерные классы, обеспечивающие доступ в Интернет, и оснащенные мультимедиа проектором или иными средствами визуализации учебного материала, магнитной доской или флипчартом.
- Электронный информационно-образовательный портал, размещенный на сервере в информационно-телекоммуникационной сети "Интернет".
- Специальное программное обеспечение для Python-разработки, необходимое для реализации образовательных задач курса.
- Стандартное программное обеспечение для работы над разработкой учебно-методических материалов.
- Мастерские и аудитории для проведения открытых занятий.

Специальных помещений, предполагающих наличие какого-либо специального оборудования для реализации данной программы, не предусматривается.

7.2. Кадровое обеспечение программы

Образовательный процесс по программе осуществляется педагогом дополнительного образования с профильным высшим или средним профессиональным образованием.

К занятию педагогической деятельностью по дополнительной общеобразовательной программе также допускаются лица, обучающиеся по образовательным программам высшего образования по специальностям и направлениям подготовки, соответствующим направленности дополнительных общеобразовательных программ, и успешно прошедшие промежуточную аттестацию не менее чем за два года обучения.

Реализация дополнительной общеобразовательной (общеразвивающей) программы обеспечивается руководящими и педагогическими работниками организации, а также лицами, привлекаемыми к реализации программы на условиях гражданско-правового договора.

У педагогического работника, реализующего дополнительную общеобразовательную программу, должны быть сформированы основные компетенции, необходимые для

обеспечения успешного достижения обучающимися планируемых результатов освоения программы, в том числе умения:

- обеспечивать условия для успешной деятельности, позитивной мотивации, а также самомотивирования обучающихся;
- осуществлять самостоятельный поиск и анализ информации с помощью современных информационно-поисковых технологий;
- разрабатывать программы учебных предметов, выбирать учебники и учебно-методическую литературу, рекомендовать обучающимся дополнительные источники информации, в том числе Интернет-ресурсы;
- реализовывать педагогическое оценивание деятельности обучающихся;
- работать с текстовыми редакторами, электронными таблицами, электронной почтой и браузерами, мультимедийным оборудованием.

1. Какой тип переменных отсутствует в Python?

- dict
- str
- int
- **func**
- list

2. Продолжите фразу: “В языке программирования Python множество значений в одну переменную ...

- записать нельзя
- можно записать только с помощью функции
- можно записать в переменную типа str
- **можно записать в переменную типа tuple**

3. Вы пишете программу, которая будет выводить на экран температуру человека. Температура тела, как известно, принимает значения типа 36,6, 38,3, 37,0. Какой тип переменной нужно использовать для хранения значения температуры:

- **float**
- dict
- int
- list

4. Выберите верное утверждение о циклах в Python:

- **цикл for нельзя сделать бесконечным**
- цикл while всегда бесконечный
- цикл for может работать с любым типом переменных
- цикл for можно зациклить

5. Функции в программировании используются, чтобы:

- **сократить время на создание программы за счет переноса повторяющихся действий в одно место кода**
- несколько раз напечатать повторяющуюся информацию
- выводить на экран имя создателя программы

6. Какая функция выводит что-либо в консоль?

- log
- **print**
- write
- out

7. Что покажет этот код?

```
for i in range(5):  
    if i % 2 == 0:  
        continue  
    print(i)
```

- Ошибку, так как `i` не присвоена
- Числа: 0, 2 и 4
- **Числа: 1 и 3**
- Ошибку из-за неверного вывода
- Числа: 1, 3 и 5

8. Сколько библиотек можно импортировать в один проект?

- **Неограниченное количество**
- Не более 23
- Не более 10
- Не более 3
- Не более 5

9. В какой из этих строк создание переменной не вызовет ошибки?

- `var num = 2`
- `int num = 2`
- `$num = 2`
- **`num = 2`**

10. Какая библиотека обеспечивает поиск по YouTube?

- `searchingvideos`
- `utube`
- `youtubearchpython`
- `findvideosininternet`

11. Чтобы все команды бота распознавались во всех регистрах нужно использовать функцию:

- `makemini`
- `greater`
- `lower`
- `is`

Приложение 2. Примерные задания для оценки качества освоения учебного материала

Задание 1. Используя изученные операторы присваивания и вывода, выполните следующие действия:

- Выведите на печать ваше имя
- Выведите на печать текущую дату в виде “ДД.ММ.ГГГГ”
- Выведите на печать число - сколько минут идет наше занятие
- Выведите на печать сумму, разность, произведение, частное чисел

Задание 2. В программе определите три текстовые переменные. Соберите их в официальное приветствие, например, “Приветствую вас, земляне!”

Задание 3. Определите две переменные: цена товара в рублях (например, 230), и купюра, которой расплачивается покупатель (например, 500). Нужно рассчитать и вывести, сколько нужно дать сдачи с этой купюры.

Задание 4. На вход подается два числа. Напишите программу, которая печатает все числа, расположенные между этими двумя числами

Задание 5. Настройте работу бота автоответчика следующим образом:

- дайте боту команду дублировать ваше сообщение и приветствовать вас: "Ваше сообщение" - ответ бота “Ваше сообщение, привет, я бот!”
- пусть бот в ответ на ваше сообщение поприветствует вас и расскажет о своих функциях.

Задание 6. В некоторой школе занятия начинаются в 9:00. Продолжительность урока — 45 минут, после 1-го, 3-го, 5-го и т.д. уроков перемена 5 минут, а после 2-го, 4-го, 6-го и т.д. — 15 минут. Дан номер урока (число от 1 до 10). Определите, когда заканчивается указанный урок. Выведите два целых числа: время окончания урока в часах и минутах. Решенное задание поместите в репозиторий GitHub

Задание 7. Задача - написать функцию bank, которая вычисляет проценты с банковского вклада - принимает аргументы a и years, и возвращает сумму, которая будет на счету пользователя. Пользователь делает вклад в размере a рублей сроком на years лет под 10% годовых (каждый год размер его вклада увеличивается на 10%. Эти деньги прибавляются к сумме вклада, и на них в следующем году тоже будут проценты). Решенное задание поместите в репозиторий GitHub.

Задание 8. Напишите программу, которая принимает текст и выводит два слова: наиболее часто встречающееся и самое длинное.

Задание 9. Соберите статистику по работе бота «Угадай мелодию» для песен - как правильных, так и полученных от пользователя. Создайте выгрузку данных, постройте графики, по которым можно будет узнать: какие песни наиболее популярны (их будут отгадывать чаще всего), какие песни наименее популярны - меньше всего отгаданы.

Задание 10. Создайте бота-«телефонную книгу», который будет сохранять номер телефона для каждого пользователя отдельно. Пользователь отправляет контакт боту, который его сохраняет. При вызове команды /list бот выводит список всех контактов, которые он сохранил. Создайте для бота две кнопки: добавить новый контакт и посмотреть список контактов.