

МУНИЦИПАЛЬНОЕ ОБЩЕОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
«СРЕДНЯЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ШКОЛА № 102»
ЛЕНИНСКОГО РАЙОНА Г. САРАТОВА

«Рассмотрено» На заседании ШМО _____ / Е.Г. Егорова/ Протокол № <u>1</u> от <u>31.08</u> 2023г.	«Согласовано» Заместитель директора по УР _____ /И.И. Чезганова/ <u>«31»</u> <u>08</u> 2023г	«Утверждено» Директор МОУ «СОШ №102» _____ /Н.Ю. Кожевникова/ Приказ № <u>317-0</u> от « <u>31</u> » <u>08</u> 2023г.
--	---	---

Календарно-тематическое планирование
внеурочной деятельности
«Python для начинающих»
Возраст учащихся: 12—18 лет
на 2023-2024 учебный год

уровень получения образования: среднее общее

форма обучения: очная

Учитель: Качула Евгений Евгеньевич

Рассмотрено на заседании
педагогического совета
протокол № 1
от «31» 08 2023 г.

Саратов 2023 г.

Раздел 1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

1.1. Актуальность программы

По результатам исследования компании Jet Brains (<https://www.jetbrains.com/ru-ru/lp/devecosystem-2019/>) язык Python – самый изучаемый в 2019 году и один из самых востребованных на рынке труда. Python достаточно широко используется при изучении основ алгоритмизации и программирования в школьном курсе информатики. В частности, используется в качестве базового языка для изучения в УМК по информатике авторов Поляков К. Ю., Еремин Е. А., допущенном к использованию в общеобразовательных организациях Министерством просвещения РФ (Приказ Министерства Просвещения РФ от 28 декабря 2018 года № 345).

Изучение Python в школе откроет ученикам возможности дальнейшего развития в области IT и поможет профориентации в старших классах, пригодится в олимпиадах по программированию и решению заданий ЕГЭ.

Курс предполагает смешанный формат обучения. Сочетание групповой работы с учителем в классе и индивидуальной работы в личном кабинете на онлайн-платформе позволяет ученикам выработать не только технические навыки программирования, но и навыки социального взаимодействия при работе над финальным проектом курса, а главное – научиться самостоятельно выстраивать свое профессиональное развитие.

Цель реализации программы – формирование у обучающихся навыков программирования на языке Python и самонаправленного обучения.

Направленность дополнительной общеобразовательной программы:
техническая.

Задачи реализации программы:

Обучающие

1. Изучить основы программирования на языке Python;

Развивающие

2. Научиться применять полученные знания для решения практических задач.

Воспитательные

3. Научиться применять полученные знания для решения практических задач.
4. Повысить уровень самостоятельности в обучении (по четырехступенчатой шкале Г. Гроу)

Организация курса:

Курс состоит из 25 уроков длительностью 90 минут. Урок продолжительностью 90 минут делится на 2 части по 45 минут с перерывом в 10 минут.

Методические материалы курса состоят из:

1. Методических указаний для учителя в текстовом виде,
2. Презентации с иллюстративным изложением теоретического материала;
3. Упражнений на платформе Stepik с теоретическим и практическим материалом языка Python;
4. Интерактивных проверочных заданий в приложениях Kahoot и Learning Apps;
5. Подвижных игр, направленных на закрепление знаний, полученных на занятии.
6. Инструкций для проведения рефлексии процесса обучения с учениками.

Планируемые результаты обучения:

Личностные

По окончании курса учащийся сможет:

- Программировать на языке Python.
- Использовать инструменты разработки среды Wing.
- Самостоятельно реализовывать проекты, связанные с разработкой игр.

Метапредметные

- Ставить учебные цели.
- Формулировать достигнутый результат.
- Планировать свою самостоятельную учебно-познавательную деятельность; выбирать индивидуальную траекторию достижения учебной цели.
- Определять подходы и методы для достижения поставленной цели.
- Отбирать необходимые средства для достижения поставленной цели.
- Осуществлять самооценку промежуточных и итоговых результатов своей самостоятельной учебно-познавательной деятельности.

- Проводить рефлексию своей учебно-познавательной деятельности.

Предметные

Практическим результатом работы служит финальный проект каждого ученика: 2D игра на движке Pygame, либо серия самостоятельно разработанных мини-проектов в консольном режиме языка Python.

Основные формы организации занятий программы «Python для начинающих»:

- Практические занятия с использованием онлайн-платформы Stepik;
- Работа в IDE «Wing 101»;
- Прохождение опросов в приложениях Kahoot и Learning Apps;
- Домашние практические занятия с использованием онлайн-платформы Stepik, направленные на отработку навыков программирования на языке Python.

Категория обучающихся: ученики общеобразовательных школ от 12 до 18 лет в рамках внеурочной деятельности и дополнительного образования.

Форма обучения: смешанная: очное и онлайн-обучение. В очных занятиях в группах по 10-15 человек используются задания на онлайн-платформе с автоматизированной проверкой. Задания на платформе доступны ученикам для самостоятельного изучения в любое время. В ходе курса запланирован плавный переход к формату “перевернутого класса”: на первых пяти занятиях учеников знакомят с онлайн-платформой и основными инструментами программирования на Python. Начиная с шестого урока предлагаются различные формы самостоятельной активности на уроке, с групповой взаимопроверкой. После десятого урока иногда ученикам предлагается самостоятельно изучить новую тему, опираясь на конспект онлайн-курса, а роль учителя сдвигается к консультации и модерации. Последние пять уроков отведены на почти самостоятельную работу над проектами.

Режим занятий: Курс состоит из 25 уроков длительностью 90 минут.

Срок освоения программы: в течение одного учебного года, в объеме 50—64 часов.

Отличительные особенности программы: авторский подход к организации образовательного процесса. Реализация программы предполагает проведение аудиторных занятий с использованием вариативных дистанционных ресурсов:

- авторских онлайн уроков, размещенных на платформе Stepik;
- авторских интерактивных заданий (опросов, викторин, дидактических игр и др.), размещенных на бесплатных специализированных сервисах Kahoot и Learning Apps;
- авторских проектных заданий для создания учащимися игр на языке Python.

Раздел 2. Содержание программы

В рамках смешанного подхода к обучению каждая из тем, изложенных преподавателем, отрабатывается в уроках платформы Stepik и в практических заданиях. (<https://stepik.org/course/58852/syllabus>)

2.1. Учебный (тематический) план

№	Название темы	Количество часов	Теоретические	Практические
1	Знакомство с Python. Команды input() и print().	2	1	1
2	Параметры sep, end. Переменные. Комментарии. PEP 8	2	1	1
3	Работа с целыми числами	2	1	1
4	Условный оператор. Логические операции and, or, not	2	1	1
5	Вложенный и каскадный условный оператор	2	1	1
6	Типы данных int, float, str. Встроенные функции min(), max(), abs(). Оператор in.	2	1	1
7	Цикл for. Функция range().	2	1	1
8	Частые сценарии при написании циклов. Расширенные операторы присваивания.	2	1	1
9	Цикл с предусловием while	2	1	1
10	Операторы break, continue, else.	2	1	1
11	Вложенные циклы	2	1	1
12	Строковый тип данных: индексация и срезы	2	1	1
13	Методы строк	2	1	1
14	Резервное время. Введение в списки.	2	1	1
15	Основы работы со списками. Методы списков	2	1	1
16	Вывод элементов списка. Строковые методы split() и join()	2	1	1

17	Методы списков. Списочные выражения	2	1	1
18	Функции	2	1	1
19	Локальные и глобальные переменные. Функции возвращающие значения.	2	1	1
20	Функции возвращающие значения.	2	1	1
21	Работа над проектом	2	1	1
22	Работа над проектом	2	0	2
23	Работа над проектом	2	0	2
24	Работа над проектом	2	0	2
25	Работа над проектом	2	0	2
	Итого	50		

2.2. Календарно – тематическое планирование

№	Название темы	Дата План.	Дата Факт.	Примечание
1	Знакомство с Python. Команды input() и print().	5.09		
2	Параметры sep, end. Переменные. Комментарии. PEP 8	12.09		
3	Работа с целыми числами	19.09		
4	Условный оператор. Логические операции and, or, not	26.09		
5	Вложенный и каскадный условный оператор	3.10		
6	Типы данных int, float, str. Встроенные функции min(), max(), abs(). Оператор in.	10.10		
7	Цикл for. Функция range().	17.10		
8	Частые сценарии при написании циклов. Расширенные операторы присваивания.	24.10		
9	Цикл с предусловием while	7.11		
10	Операторы break, continue, else.	14.11		
11	Вложенные циклы	21.11		
12	Строковый тип данных: индексация и срезы	28.11		
13	Методы строк	5.12		
14	Резервное время. Введение в списки.	12.12		
15	Основы работы со списками. Методы списков	19.12		
16	Вывод элементов списка. Строковые методы split() и join()	26.12		
17	Методы списков. Списочные	9.01		

	выражения			
18	Функции	16.01		
19	Локальные и глобальные переменные. Функции возвращающие значения.	23.01		
20	Функции возвращающие значения.	30.01		
21	Работа над проектом	6.02		
22	Работа над проектом	13.02		
23	Работа над проектом	20.02		
24	Работа над проектом	27.02		
25	Работа над проектом	6.03		
26	Работа над проектом	13.03		
27	Работа над проектом	20.03		
28	Работа над проектом	3.04		
29	Работа над проектом	17.04		
30	Работа над проектом	24.04		
31	Работа над проектом	8.05		
32	Работа над проектом	15.05		
33	Работа над проектом	22.05		

Раздел 3. Формы аттестации и оценочные материалы

Аттестация проводится в форме выполнения индивидуальных и групповых заданий по пройденному материалу. Контроль в указанной форме осуществляется как промежуточный, так и итоговый. Отметочная форма контроля отсутствуют.

Для заданий на онлайн-тренажере указан необходимый минимум — 70% выполненных заданий, чтобы тема считалась пройденной успешно и был открыт доступ к следующей теме. После каждой темы в онлайн-курсе стоит итоговая работа: от ученика требуется в ограниченное время (три часа) решить набор задач по пройденной теме. В среднем, ученик справляется с решением за 30 минут. Преподаватели могут использовать эти итоговые работы в качестве промежуточных проверочных работ.

В конце курса, по итогам работы над групповыми и индивидуальными проектами проводится обсуждение результатов в коллективе с опорой на чек-лист, исправление ошибок и, тем самым, коррекция и закрепление полученных знаний.

Раздел 4. Организационно-педагогические условия реализации программы

4.1. Литература для педагога

1. **Васильев, А. Н. Python на примерах** [Текст]: практ. курс / А. Н. Васильев - Наука и Техника, 2019 - 432 с.
2. **Прохоренок, Н. А. Python 3: самое необходимое** [Текст]: практ. курс / Н. А. Прохоренок, В. А. Дронов - БХВ-Петербург, 2019 - 608 с.
3. **Гэддис, Т. Начинаем программировать на Python** [Текст]: учебник / Т. Гэддис - БХВ-Петербург, 2019 - 768 с.
4. **Седжвик, Р. Программирование на языке Python** / Р. Седжвик, К. Уэйн, Р. Дондеро - Вильямс, 2017 - 736 с.
5. **Харрисон, М. Как устроен Python.** [Текст]: практ. курс / М. Харрисон - Питер, 2002 - 272 с.

Литература для обучающихся: Не предусмотрена

Электронные ресурсы:

1. Курс Поколение Python: курс для начинающих на платформе Stepik.
(<https://stepik.org/course/58852/syllabus>)

4.2. Материально-технические условия реализации программы

1. Обязательные

- помещение (предпочтительно, изолированное);
- 10—15 рабочих мест: стол, стул, розетка, компьютеры на каждое рабочее место;
- проектор, аудио колонки;
- Интернет-соединение, скорость загрузки не менее 2 Мбит/сек;
- меловая, магнитно-маркерная доска или флипчарт;
- общие условия в соответствии с СанПиН 2.4.4.3172-14

Требования к ПО:

- Операционная система Windows 7 или моложе / MacOS / Unix-based системы с поддержкой протокола HTML5;
- Приложения Google Chrome, Gimp, Brackets;
- интерактивная оболочка (бесплатная IDE Wing101 или аналог).