

Тематический план
Курс программирования Python Start

Кол-во занятий (ак.ч.)	Темы	Образовательные результаты	Содержание
Модуль 1. Основы языка			
4 (8)	<ul style="list-style-type: none"> - Знакомство с языком - Переменные. Функция input() - Выражения. Типы данных 	<ul style="list-style-type: none"> - Ученик перечисляет встроенные типы данных (символ, число). - Ученик умеет программировать арифметические операции. - Ученик умеет считывать и выводить данные с помощью input() и print(). - Ученик умеет находить ошибку в своей программе. <p><i>Установки:</i></p> <ul style="list-style-type: none"> - Ученик понимает, что ошибка 	<ul style="list-style-type: none"> - Задания в рабочей тетради и на платформе на освоение нового материала. - Программа-аватар. Программа текстом приветствует ученика и интересуется его возрастом и увлечениями.

		— это инструмент обучения.	
Модуль 2. Управляющие конструкции			
5 (10)	<ul style="list-style-type: none"> - Условный оператор. Сравнение чисел и строк - Вложенные условные операторы - Циклы 	<p><i>Предметные:</i></p> <ul style="list-style-type: none"> - Ученик знает устройство условного оператора, цикла while. - Ученик понимает, как применяется условный оператор внутри условного оператора (вложенный условный оператор) или внутри цикла. - Ученик знает требования к оформлению кода и следует им (отступы, двоеточия, комментарии). - Ученик знает, что итерационный подход — это поэтапная разработка с постоянным анализом результата. - Ученик пишет программу 	<ul style="list-style-type: none"> - Задания в рабочей тетради и на платформе на освоение нового материала. - Программа «Чат-бот» в консоли. Она приветствует пользователя, может загадать ему загадку, поработать калькулятором, решить квадратное уравнение.

		поэтапно (итерационно).	
Модуль 3. Функции и модули			
3 (6)	<ul style="list-style-type: none"> - Создание функций - Циклы и функции - Создание модулей. Модуль random 	<ul style="list-style-type: none"> - Ученик умеет создать функцию и применить её. - Ученик знает, что модуль — это файл с программой, который может быть подключен к другой программе. - Ученик понимает назначение модуля random (задание случайных чисел). - Ученик умеет подключать random и использовать функцию randint() <p><i>Установки:</i></p> <ul style="list-style-type: none"> - Ученик понимает, что при работе над программой важно получать обратную связь. - Ученик умеет дорабатывать программу с учетом обратной связи. 	<ul style="list-style-type: none"> - Задания в рабочей тетради и на платформе на освоение нового материала. - Игра “Угадай число”.

Модуль 4. Модуль Turtle (черепашка)

4 (8)

- Turtle. Линейные алгоритмы
- Turtle. Переменные и функции
- Turtle. Циклы
- Turtle. Условный оператор

- Ученик понимает, что модуль Turtle нужен для создания графических объектов.
- Ученик использует функции модуля forward(), left(), color() и другие.
- Ученик использует изученные ранее конструкции вместе с инструментами Turtle.

Установки:

- Ученик не боится решать задачи, с которыми он раньше не сталкивался.
- Ученик умеет презентовать свою работу
- Ученик понимает, что любой рутинный процесс можно автоматизировать.

- Задания в рабочей тетради и на платформе на освоение нового материала.
- Проект «Город». Программа, которая сможет создавать рисунки (узор из звезд на небе, застройку деревни, изображение робота и т. д.). Пользователь вводит параметры города: сколько в нем домов, деревьев, какая в текущий момент погода, программа генерирует соответствующий пейзаж. Часть параметров может быть задана случайно или зависеть от ответов пользователя не явно, например, если настроение грустное, то генерируется дождь, а если весёлое, то солнце.

Модуль 5. Объектно-ориентированное программирование

5 (10)	<ul style="list-style-type: none"> - Объекты, свойства, методы - События и их обработка - Классы - Наследование 	<ul style="list-style-type: none"> - Ученик понимает, что называется объектом, свойством, методом. - Ученик умеет использовать готовые объекты в своих программах. - Ученик понимает, что наследование — это (важное свойство ООП) умение использовать готовые свойства и методы, добавляя к ним свои. - Ученик умеет создать класс-наследник уже имеющегося класса. 	<ul style="list-style-type: none"> - Задания в рабочей тетради и на платформе на освоение нового материала. - Программа, содержащая кнопки и реагирующая на их нажатия. Игра-аркада, в которой объект - наследник черепашки, ходит по полю, добирается до цели.
Модуль 6. Проектный модуль			
Часть 1 (4 ур)	<ul style="list-style-type: none"> - Библиотека Play: задание спрайтов, спрайты как физические объекты, таймер 	<ul style="list-style-type: none"> - Ученик понимает, что Play — это простой и мощный инструмент для создания игр и приложений. - Ученик умеет работать со спрайтами как с физическими 	<ul style="list-style-type: none"> - Учебный проект «Лабиринт». Программа, в которой спрайт должен обойти препятствия-спрайты и дойти до финиша. - Проекты по выбору «Экранный

		<p>объектами.</p> <ul style="list-style-type: none"> - Ученик понимает, как работает физика, описанная в Play. - Ученик знает, что список — это структура для хранения данных любого типа. - Ученик умеет обращаться к элементам списка по индексу. <p><i>Установки:</i></p> <ul style="list-style-type: none"> - Ученик понимает, что невозможно запомнить все, поэтому важно уметь пользоваться внешними источниками информации. 	<p>питомец», «Игровой автомат», «Пинг-понг», «Фрактал» и др.</p>
<p>Часть 2 (4 ур)</p>	<ul style="list-style-type: none"> - Списки: задание списка объектов, перебор списка по индексу - Цикл for и функция range() - Pygame: работа со 	<ul style="list-style-type: none"> - Ученик знает, что список — это удобная структура для хранения данных любого типа и работы с ними; - Ученик знает методы работы со списками (len, append и др) и применяет их в решении 	<ul style="list-style-type: none"> - Учебный проект "Пианино". Пианино обладает восемью клавишами. Когда пользователь мышкой кликает на клавиши, воспроизодятся звуки пианино. Звуки пианино записываются -> получается мелодия, которую

	<p>звуками</p>	<p>практических задач;</p> <ul style="list-style-type: none"> - Ученик умеет работать с циклом for и использовать в нем функцию range - Ученик понимает, что средств Play хватает не всегда, поэтому необходимо изучать средства других библиотек, например, Pygame; - Ученик умеет задавать приложению фоновую музыку и работать со звуковыми эффектами с помощью Pygame <p><i>Установки:</i></p> <ul style="list-style-type: none"> - <i>понимает, что для решения задачи не всегда нужно братья за новые инструменты. Часто нужно иначе взглянуть на старые и применить их по-новому</i> 	<p>можно прослушать, нажав на Play</p> <ul style="list-style-type: none"> - Выбор между проектами "Арканоид" или "Синтезатор" (является развитием проекта "Пианино" в мультиинструментальную программу)"
<p>Часть 3</p>	<p>Комплексное</p>	<p>-Ученик уверенно применяет</p>	<p>1) Проект по выбору:</p>

<p>(4 ур)</p>	<p>повторение пройденного материала и его применение при реализации проектов по выбору</p>	<p>конструкции Python на практике (функции, списки, циклы, работа с файлами); -Ученик умеет использовать ограниченный набор инструментов для реализации разных целей</p> <p><i>Установки:</i></p> <ul style="list-style-type: none"> - <i>понимает, что для решения задачи не всегда нужно братья за новые инструменты. Часто нужно иначе взглянуть на старые и применить их по-новому</i> 	<ul style="list-style-type: none"> - "Яйцелов". Игра, в которой пользователь "корзинкой" должен поймать падающие сверху "яйца"; - "Двадцать одно". Игра напоминает аналогичную карточную игру. Пользователь может нажимать только на 2 кнопки: ЕЩЕ и СТОП. При нажатии на ЕЩЕ на стол кладется карточка с числом. При СТОП раздача прекращается. Затем появляются N случайных карточек по количеству выложенных пользователем. Пользователь выигрывает, если сумма очков на его картах окажется больше. <p>2) Модуль завершается проектом</p>
---------------	--	---	---

			<p>"Платформер", который каждый ребенок может реализовать в соответствии со своими идеями. Год завершается презентацией проектов и общением с родителями.</p>
--	--	--	---