

Министерство образования и занятости населения Приморского края  
Краевое государственное автономное профессиональное образовательное  
учреждение  
«Владивостокский судостроительный колледж»



«УТВЕРЖДАЮ»

Директор КГА ПОУ «ВСК»

Глушкова И.В.

«25» августа 2023 г.

## Программирование на Python

Дополнительная общеобразовательная общеразвивающая программа  
технической направленности

Возраст учащихся: 14-17 лет  
Срок реализации программы: 1 год

Колмогоров А.Ю.,  
Преподаватель

г. Владивосток  
2023 г.

## Раздел № 1. ОСНОВНЫЕ ХАРАКТЕРИСТИКИ ПРОГРАММЫ

### 1.1 Пояснительная записка

#### **Актуальность программы**

Программа разработана в соответствии с методическими рекомендациями по созданию и функционированию центров цифрового образования “IT-Куб” от 18 апреля 2023 года, а так же федеральными нормативными правовыми актами в области дополнительного образования, государственными требованиями к образовательным программам системы дополнительного образования детей, а также локальными нормативными правовыми актами организации.

Актуальность данной программы обусловлена повышенным спросом на изучение языков программирования детьми, в частности языка Python. Python изучается в школьном курсе информатики, необходим для решения олимпиадных и конкурсных заданий, а также сдачи ЕГЭ, имеет прикладной характер и может использоваться для решения повседневных задач.

Программа предназначена для тех, кто не обладает знаниями в области разработки, поэтому она подойдет тем, кто только начинает изучать программирование. А благодаря обширности тем, которые затрагиваются в этом курсе, программа будет полезна и тем, кто уже имеет некоторый опыт изучения языков программирования.

**Направленность программы:** техническая.

**Язык реализации программы:** русский.

**Уровень освоения:** стартовый (ознакомительный).

**Отличительные особенности:** состоят в использовании следующих педагогических технологий:

– технология индивидуального обучения – проведение учебно-воспитательной работы с группой, ориентируясь на личностные особенности каждого обучающегося;

– технология коллективного взаимообучения – в процессе обучения во время коллективных работ обучающиеся обмениваются опытом

и знаниями, дополняя таким образом знания друг друга;

– технология модульного обучения – каждый модуль является самостоятельной, отдельной единицей, благодаря чему отслеживание усвоения программы отдельными учащимися значительно упрощается.

**Адресат программы:** школьники города Владивостока 14-17 лет.

**Особенности организации образовательного процесса:**

Для обучения на дополнительной общеобразовательной программе “Программирование на Python” не требуется начальных знаний, так как уровень освоения программы – “Стартовый (ознакомительный)”. Дети могут обучаться по данной программе без каких-либо знаний в этой области.

Поступление на дополнительную общеобразовательную программу “Программирование на Python” осуществляется на основании сертификата на право получения дополнительного образования.

***Инструкция по получению сертификата дополнительного образования:***

1. Получить сертификат дополнительного образования необходимо на сайте портала “Сетевой город. Образование (АИС СГО)” <https://sgo.prim-edu.ru/authorize/login>
2. Подать заявку на обучение в КГА ПОУ “ВСК” по выбранному направлению на портале “Персонафицированного дополнительного образования” <https://25.pfdo.ru/>.

Группы формируются в соответствии с количеством поданных заявок на портале персонафицированного дополнительного образования. Если количество поданных заявок превышает 12, то тогда будет вестись набор слушателей во вторую и последующую группы обучения.

- **Режим занятий:** занятия проводятся в группах от 8 до 12 человек, продолжительность одного занятия — 45 минут;

- **Продолжительность образовательного процесса:** общая продолжительность программы – 1 год, 72 часов, 2 часа в неделю.

## **1.2 Цель и задачи программы**

**Цель программы:** формирование базовых навыков программирования у школьников города Владивостока 14-17 лет с помощью языка “Python”.

**Задачи программы:**

**Воспитательные:**

1. Сформировать способности к саморазвитию и самообразованию средствами информационных технологий на основе иллюстрированной среды программирования, мотивации к обучению и познанию;
2. Сформировать умения работать индивидуально и в группе для решения поставленной задачи;
3. Сформировать целостное мировоззрение, соответствующее современному уровню развития информационных технологий;
4. Сформировать осознанное позитивное отношение к другому человеку, его мнению, результату его деятельности;
5. Обеспечить усвоение правил индивидуального и коллективного безопасного поведения при работе с компьютерной техникой;
6. Сформировать культуру начального программирования.

**Развивающие:**

1. Развить умение самостоятельно ставить и формулировать для себя новые задачи, развивать мотивы своей познавательной деятельности;
2. Развить умение самостоятельно планировать пути решения поставленной проблемы для получения эффективного результата;
3. Развить умение критически оценивать правильность решения учебно-исследовательской задачи;
4. Сформировать владение основами самоконтроля, способность к принятию решений;
5. Формировать мотивацию к профессиональному самоопределению учащихся.

### Обучающие:

1. Знание основных алгоритмических структур и умение применять его для построения алгоритмов решения задач по их математическим моделям, умения и навыки составления простых программ по построенному алгоритму на Python;
2. Умения и навыки эффективного использования основных типов прикладных программ (приложений) общего назначения и информационных систем для решения с их помощью практических задач;
3. Умение грамотно интерпретировать результаты решения практических задач с помощью информационных технологий, применять полученные результаты в практической деятельности.

### 1.3 Содержание программы

#### Учебный план программы “Программирование на Python”

№ п/п	Наименование раздела	Количество часов			Форма аттестации
		Теория	Практика	Всего	
1	Среда программирования Python. Переменные	2	2	4	Беседа
2	Первые программы на языке Python, основные операторы	0	6	6	Устный опрос
3	Условный оператор if	3	3	6	Устный опрос
4	Циклы в языке Python	4	4	8	Устный опрос
5	Решение задач по темам: “Условный оператор и циклы”	0	6	6	Зачёт
6	Условные операторы в циклах	0	2	2	Тестирование
7	Списки в языке Python	3	3	6	Устный опрос
8	Строки в языке Python	3	3	6	Устный опрос

9	Решение задач по пройденным темам	0	6	6	Зачёт
10	Функции по работе со строками	0	2	2	Тестирование
11	Функции в языке Python	3	3	6	Устный опрос
12	Кортежи в языке Python	3	3	6	Устный опрос
13	Массивы в языке Python	3	3	6	Зачёт
14	Программирование на Python	0	2	2	Контрольная работа
<b>Итого</b>		<b>26</b>	<b>46</b>	<b>72</b>	

### Содержание учебного плана

#### 1. Тема: Среда программирования Python. Переменные.

*Теория.* Среда программирования в языке Python, основные инструменты среды, понятие «переменная».

*Практика.* Технические действия в среде в среде программирования.

#### 2. Тема: Первые программы на языке Python, основные операторы.

*Практика.* Простые программы на Python. Операторы присваивания, ввода/вывода данных, разработка программ.

#### 3. Тема: Условный оператор if.

*Теория.* Условный оператор if в Python.

*Практика.* Разработка программ, реализующих условные алгоритмы.

#### 4. Тема: Циклы в языке Python.

*Теория.* Операторы цикла for, while в Python.

*Практика.* Разработка программ, циклические алгоритмы.

#### 5. Тема: Решение задач по темам: “Условный оператор и циклы”.

*Практика.* Решение дополнительных задач по темам “Условный оператор”, “Циклы”.

#### 6. Тема: Условные операторы в циклах.

*Практика.* Проверка полученных навыков по темам «Условный оператор if», «Циклы в языке Python».

#### 7. Тема: Списки в языке Python.

*Теория.* Понятие «список» в Python.

*Практика.* Создание списка, различные способы задания списка, вывод элементов списка на экран, основные функции по работе со списками в Python.

#### **8. Тема: Строки в языке Python.**

*Теория.* Понятие «строка» в Python.

*Практика.* Способы задания строк, основные функции по работе со строками в Python.

#### **9. Тема: Решение задач по изученным темам.**

*Практика.* Решение дополнительных задач по темам «Списки в языке Python».

#### **10. Тема: Функции по работе со строками.**

*Практика.* Проверка полученных навыков по темам «Списки в языке Python», «Строки в Python».

#### **11. Тема: Функции в Python.**

*Теория.* «Функция» в Python, описание функции, основные приёмы структурного программирования.

*Практика.* Обращение к функции в тексте программы, приёмы написания программ с использованием вспомогательных алгоритмов

#### **12. Тема занятия: Кортежи в Python.**

*Теория.* Понятие «кортеж» в Python.

*Практика.* Функции по работе с кортежами в Python.

#### **13. Тема: Массивы в языке Python.**

*Теория.* Понятие и структура массива.

*Практика.* Программы, содержащие массив.

#### **14. Тема: Программирование на Python.**

*Практика.* Задачи по программе всего курса.

### **1.4 Планируемые результаты**

**Личностные результаты:**

Обучающийся будет готов и способен к саморазвитию и личностному самоопределению;

У обучающегося будет сформирована мотивация к обучению и целенаправленной познавательной деятельности.

### **Метапредметные результаты:**

Обучающийся будет знать, как ставить и формулировать для себя цели действий, прогнозировать результаты, анализировать их (причём как положительные, так и отрицательные), делать выводы в процессе работы и по её окончании, корректировать намеченный план, ставить новые цели; умение соотносить свои действия с планируемыми результатами, осуществлять контроль своей деятельности.

Обучающийся приобретёт умение оценивать правильность выполнения учебной задачи, корректировать свои действия в соответствии с изменяющейся ситуацией, определять способы действий в рамках предложенных условий.

### **Предметные результаты:**

1. Обучающийся будет владеть теоретической базой и практическими навыками программирования на языке Python;
2. Обучающийся будет владеть навыками разработки, тестирования и отладки несложных программ;
3. Обучающийся будет знать навыки разработки эффективных алгоритмов и программ на основе изучения языка программирования Python;
4. Обучающийся будет уметь развивать логические, аналитические и творческие способности.

## **РАЗДЕЛ № 2. ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИЕ УСЛОВИЯ**

### **2.1 Условия реализации программы**

#### **1. Материально-техническое обеспечение:**

Характеристика помещения для занятий по программе - учебное помещение образовательного учреждения, оснащенное наглядными пособиями, учебным оборудованием, мебелью и техническими средствами

обучения, в котором проводится учебная, индивидуальная и внеклассная работа с обучающимися в соответствии с действующими государственными образовательными стандартами, учебным планом и образовательными программами.

Для реализации данного курса требуется следующее оборудование:

- Проектор и экран для демонстрации учебного материала
- Доска
- Персональные компьютеры для обучающихся
- Раздаточные материалы
- Наушники с микрофоном

Требуемое программное обеспечение:

- Пакет офисных приложений;
- Программное обеспечение PyCharm, Python;
- Браузер Google Chrome, Mozilla Firefox или Яндекс Браузер.

## **2. Учебно-методическое и информационное обеспечение:**

Ссылки на печатные источники:

1. Лутц М. Изучаем Python. СПб.: Символ Плюс, 2019;
2. Буйначев С. К. Основы программирования на языке Python: учебное пособие. Екатеринбург: Изд-во Урал. ун-та, 2020;
3. Шуманн Ханс-Георг. Python для детей. М.: ДМК-Пресс, 2022;
4. Пэйн, Брайсон. Python для детей и родителей. М.: Издательство «Э», 2019.

### **2.2 Оценочные материалы и формы аттестации**

В процессе изучения программы для оценки текущей работы используются методы устного, письменного, практического, машинного контроля и самоконтроля. Педагогический контроль осуществляется в несколько этапов и включает в себя несколько уровней.

#### **Виды и формы контроля знаний и навыков учащихся:**

- Текущий контроль осуществляется регулярно, посредством проведения

лабораторных занятий, заключается в ответе учащихся на контрольные вопросы, фронтальных опросах учителем, а также в демонстрации полученных скриптов в среде Python (опрос, лабораторные работы).

- Тематический контроль—определение результатов обучения за определенный раздел программы, проводится посредством выполнения учащимися контрольных работ (контрольные работы, тестовые занятия).
- Итоговый контроль — проводится по окончании обучения по программе, он предполагает комплексную проверку образовательных результатов по всем ключевым целям и направлениям (разработка учащимися индивидуальных/групповых проектов и их защита).

*Формы контроля результатов (промежуточная аттестация):*

- Беседа - это диалогический метод обучения, при котором учитель путем постановки тщательно продуманной системы вопросов подводит учеников к пониманию нового материала или проверяет усвоение ими уже изученного. (применяется во время занятия “Тема 1: Среда программирования Python. Переменные);

- Устный опрос - позволяет выявить правильность ответа по содержанию, его последовательность, самостоятельность суждений и выводов, степень развития логического мышления, культуру речи учащихся. Эта форма применяется для текущего и тематического учета, а также для отработки и развития экспериментальных умений учащихся. Причем устную проверку считают эффективной, если она направлена на выявление осмысленности восприятия знаний и осознанности их использования, если она стимулирует самостоятельность и творческую активность учащихся. (применяется во время занятий: “Тема: 2-4”, “Тема: 7-8”, “Тема 11-12”);

- Зачёт - проводится для определения достижения конечных результатов обучения по определенной теме каждым учащимся. Перед

началом изучения материала, учащиеся знакомятся с перечнем вопросов и обязательных задач по теме, а также дополнительными вопросами и задачами. Иногда целесообразны закрытые зачеты, когда учащиеся получают вопросы и задания непосредственно во время проведения зачета. Его достоинство заключается в том, что он предполагает комплексную проверку всех знаний и умений учащихся. (применяется во время занятий “Тема: 5”, “Тема: 9”, “Тема 13”);

- Тестирование - представляет собой кратковременное технически сравнительно просто составленное испытание, проводимое в равных для всех испытуемых условиях и имеющее вид такого задания, решение которого поддается качественному учету и служит показателем степени развития к данному моменту известной функции у данного испытуемого. (применяется во время занятия “Тема 6: Условные операторы в циклах.”, “Тема 10: Функции по работе со строками”);

- Контрольная работа - проводится с целью определения конечного результата в обучении по данной теме или разделу, контролировать знания одного и того же материала неоднократно. Целесообразно проводить контрольные работы различного вида. С помощью промежуточной контрольной работы Преподаватель проверяет усвоение учащимися материала в период изучения темы. Итоговая контрольная работа проводится с целью проверки знаний и умений учащихся по отдельной теме, курсу. (применяется во время занятия “Тема 14: Программирование на Python”).

*Формы фиксации результатов:*

- Ведомости результатов аттестации учащихся (применяются после завершения курса для фиксации итоговых результатов);

- Бланки тестовых заданий и контрольных работ по темам программы (применяются на протяжении всего периода обучения. Тема 1 – Тема 14);

- Разработанные, в ходе проведения занятий, программы в виде файлов (применяются на протяжении всего периода обучения. Тема 1 – Тема 14);

- Журнал посещаемости (применяется на протяжении всего периода обучения. Тема 1 – Тема 14);
- Отзывы детей и родителей (собираются после завершения курса для оперативной обратной связи по поводу предложений и замечаний);
- Свидетельство (сертификат). Выдается участнику после успешного завершения учебной программы.

### **2.3 Методические материалы**

**Дидактические и методические материалы: раздел «Среда программирования Python. Переменные» дидактических материалов пособия.**

На рисунке 1 изображен фрагмент практической работы по теме “Среда программирования Python. Переменные”.

## Практическая часть

*Цель работы:* ознакомление со средой программирования Python и запуском первых программ.

*Ход работы*

1. Проверить, установлена ли на компьютере/ноутбуке среда IDE PyCharm.
2. Ознакомиться с помощью учителя со справочным материалом, посвящённым установке языка Python, установке и первичной настройке IDE PyCharm.

3. Создать первую программу на языке Python по приведённому ниже описанию.

Хорошей традицией при изучении языка программирования является написание первой вашей программы с простым приветствием на экране «Привет, я изучаю Python!».

На жёстком диске создать каталог (папку) с именем *PythonProject*. Запустить среду разработки PyCharm. Выбрать пункт *Открыть (Open)* и указать вашу папку *PythonProject*. Откроется рабочее окно вашего проекта (рис. 29).

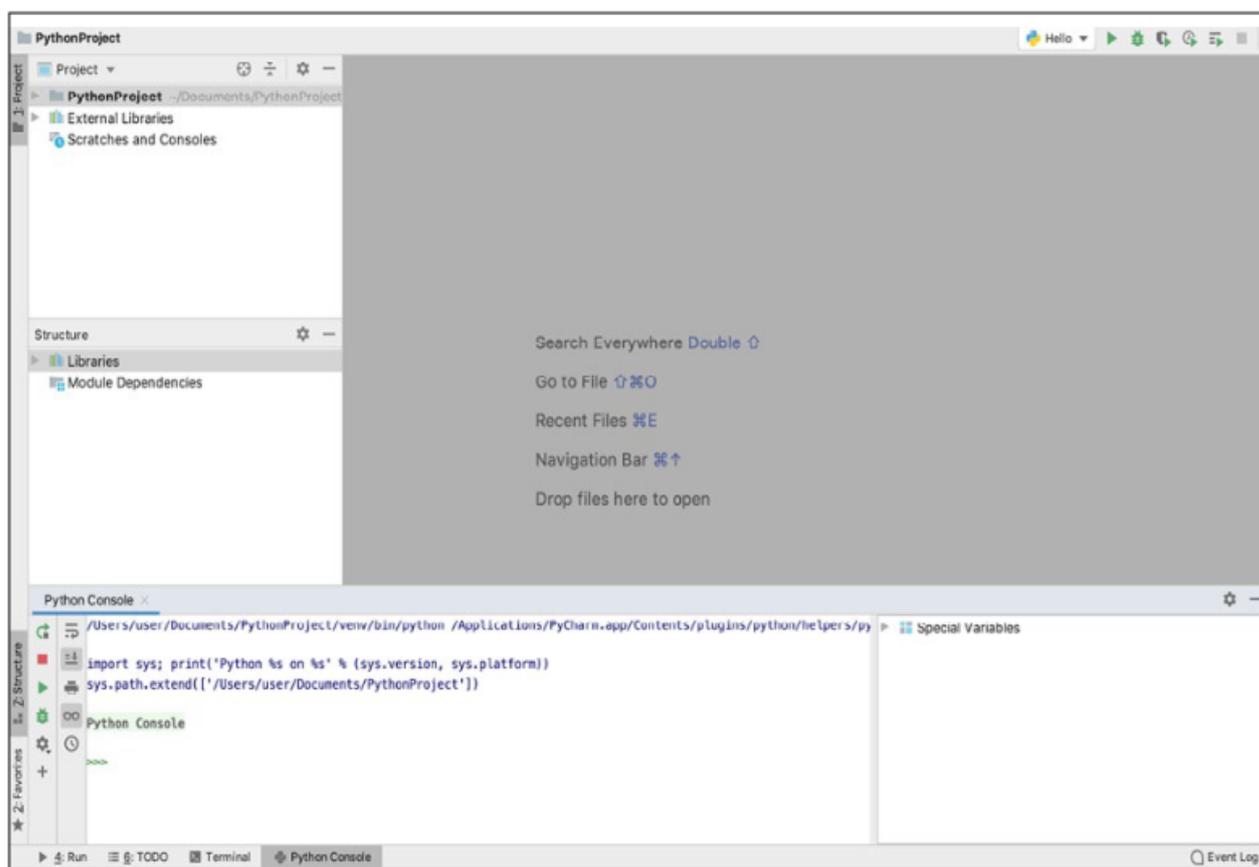


Рисунок 1 – Фрагмент дидактических материалов по Теме №1

**Дидактические и методические материалы: раздел «Первые программы на языке Python, основные операторы» дидактических материалов пособия.**

На рисунке 2 изображен фрагмент практической работа по теме «Первые программы на языке Python, основные операторы».

## Лабораторная работа 2.1. Первые программы на языке Python, основные операторы

### Теоретическая часть

В первой лабораторной работе были показаны основные аспекты работы со средой для языка Python, вы писали простые команды (операторы).

Далее предлагается рассмотреть структуру программы на языке Python, а также основные правила написания программ.

Любую программу на языке Python можно представить как набор лексем (допустимых символов), записанных в определённом порядке и по определённым правилам. Лексема может представлять собой: литералы, знаки пунктуации, переменные, специальные ключевые слова, комментарии.

Программа на языке Python может содержать достаточное количество комментариев, каждый комментарий начинается с символа # «решётка».

Литералы представляют собой значения, заданные в коде программы, например числа (25) или строки ("Привет"). В языке Python используется динамическая типизация (типы данных определяются автоматически, и их не требуется объявлять в программном коде), но при этом Python является языком со строгой типизацией (вы сможете выполнять над объектом только те операции, которые применимы к его типу).

Если говорить об использовании знаков пунктуации, то стоит отметить, что каждая строка в программе на языке Python не должна заканчиваться точкой с запятой, как, например, в C++, но если есть необходимость записать несколько операторов в одну строчку, то их можно разделять точкой с запятой.

Переменные используются для хранения данных.

#### **Важно!**

В языке Python нет специального раздела описания переменных, в котором указывается тип переменной перед её первым использованием. Тип переменной определяется по тому значению, которое ей присваивается.

Есть определённые правила для задания имён переменных (идентификаторов): это последовательность букв и цифр, которая не может начинаться с цифры, но может содержать символ подчёркивания (\_). Имена переменных чувствительны к регистру. Имена переменных не могут совпадать с ключевыми словами.

Ключевые слова в языке Python имеют специальное назначение и представляют собой управляющие конструкции языка. Примеры: `and`, `break`, `for` и т. д.

Рисунок 2 – Фрагмент дидактических материалов по Теме №2

**Дидактические и методические материалы: раздел «Первые программы на языке Python, основные операторы» дидактических материалов пособия.**

На рисунке 3 изображен фрагмент практической работы по теме «Первые программы на языке Python, основные операторы».

### Теоретическая часть

Воспользоваться материалами из лабораторной работы 2.1.

### Практическая часть

*Цель работы:* ознакомление с основами написания программ на языке программирования Python, работа с операторами присваивания, ввода/вывода данных.

*Ход работы*

1. Открыть среду разработки PyCharm.
2. Составить программу вычисления значения функции  $y(x) = x^2 - 7x + 8$  для заданного с клавиатуры значения аргумента  $x$ .

*Указание.* Примерный вид программы:

```
x=float(input("x= "))
y=x**2-7*x+8
print("y(x)= ", y)
```

3. Составить программу вычисления расстояния между двумя точками, заданными на плоскости своими координатами.

*Указание.* Примерный вид программы:

```
x1=float(input("x1= "))
y1=float(input("y1= "))
x2=float(input("x2= "))
y2=float(input("y2= "))
r=((x1-x2)**2+(y1-y2)**2)**(1/2)
print("расстояние= ", r)
```

Рисунок 3 – Фрагмент дидактических материалов по Теме №2

**Пример шаблона формы для фиксации результатов индивидуальных проектов:**

### Лист оценивания проекта

Критерии оценивания	1-я группа	2-я группа	...
Актуальность темы			
Соответствие содержания проекта заявленной теме			
Техническая сложность			
Оригинальность			
Дизайн			
Наличие соответствующего <u>музыкального</u> сопровождения с указанием в титрах <u>авторов</u> музыки			
Уровень проработанности проекта			
Возможность применения проекта в школе			
Итоговое количество баллов			

## 2.4 Календарный учебный график

**Вариант календарного учебного графика, когда программа размещается на сайте:**

Этапы образовательного процесса		1 год
Продолжительность учебного года, неделя		36
Количество учебных дней		36
Продолжительность учебных периодов	1 полугодие	11.09.2023- 31.12.2023
	2 полугодие	10.01.2023- 31.05.2024
Возраст детей, лет		14-17
Продолжительность занятия, час		2
Режим занятия		1 раз /нед
Годовая учебная нагрузка, час		72

## 2.5 Календарный план воспитательной работы

Для выстраивания адресной образовательной деятельности необходимо хорошо понимать логистическую и содержательную структуру работы, включающей обучение и воспитание учащихся.

Рабочая программа воспитательной работы Центра едина для всех творческих объединений и создает единую воспитательную среду учреждения, которая объединяет всех участников образовательного процесса. Практическая реализация цели и задач воспитания учащихся осуществляется в рамках направлений воспитательной работы образовательной организации, представленных в соответствующем модуле.

При составлении календарного плана воспитательной работы творческого объединения было адаптировано содержание модулей календарного плана программы для работы с учащимися творческого объединения по приоритетным направлениям различного уровня с учетом

конкретных условий и особенностей деятельности.

Вызовы времени заставили снова активно заговорить о воспитании личности. С 1 сентября 2020 года вступил в силу Федеральный закон от 31 июля 2020 г. № 304-ФЗ «О внесении изменений в Федеральный закон «Об образовании в Российской Федерации» по вопросам воспитания учащихся». Политика государства в сфере образования определяет воспитание как первостепенный приоритет в образовании, а в качестве важнейших задач выдвигает формирование гражданской ответственности, правового самосознания, духовности и культуры, инициативности, самостоятельности, толерантности, способности к успешной социализации в обществе.

В Федеральном законе акцентировано внимание на том, что система образования не только учит, но и воспитывает, формирует личность, передает ценности и традиции, на которых основано общество, что смысл предлагаемых поправок — «укрепить, акцентировать воспитательную составляющую отечественной образовательной системы».

В соответствии с Федеральным законом № 304-ФЗ вводится механизм организации воспитательной работы (программа воспитания), который является частью общеобразовательной (общеразвивающей) программы педагога.

## **СПИСОК ЛИТЕРАТУРЫ**

### **Ссылки на печатные источники:**

1. Лутц М. Изучаем Python. СПб.: Символ Плюс, 2019;
2. Буйначев С. К. Основы программирования на языке Python: учебное пособие. Екатеринбург: Изд-во Урал. ун-та, 2020;
3. Шуманн Ханс-Георг. Python для детей. М.: ДМК-Пресс, 2022;

4. Пэйн, Брайсон. Python для детей и родителей. М.:Издательсов «Э»,  
2019.