

УДК 371.31

https://doi.org/10.33619/2414-2948/105/54

МЕТОДЫ ОБУЧЕНИЯ ПРОГРАММИРОВАНИЮ PYTHON В ОСНОВНОЙ ШКОЛЕ

©*Талипов А. Т.*, ORCID: 0000-0003-4699-3776, SPIN-код: 6498-8019, Ошский государственный университет, г. Ош, Кыргызстан, talipovalmambet@gmail.com

©*Калдыбаев С. К.*, SPIN-код: 2421-8192, д-р пед. наук, Международный университет Ала-Тоо, г. Бишкек, Кыргызстан.

METHODS OF TEACHING PYTHON PROGRAMMING IN BASIC SCHOOL

©*Talipov A.*, ORCID: 0000-0003-4699-3776, SPIN-code: 6498-8019, Osh State University, Osh, Kyrgyzstan, talipovalmambet@gmail.com

©*Kaldybaev S.*, SPIN-code: 2421-8192, Dr habil., Ala-Too International University, Bishkek, Kyrgyzstan

Аннотация. Рассмотрены методы преподавания программирования Python в основной школе, которые помогут заинтересовать учащихся и дать им прочную основу для дальнейшего изучения программирования. Использование различных методов обучения и современных образовательных технологий позволит сделать процесс обучения эффективным и интересным для учащихся. Обучение программированию Python в основной школе является перспективным направлением. Приведены примеры программы на языке Python.

Abstract. The article discusses methods for teaching Python programming in primary schools, which will help to interest students and give them a solid foundation for further study of programming. The use of various teaching methods and modern educational technologies will make the learning process effective and interesting for students. Teaching Python programming in primary school is a promising direction. The article provides examples of programs in Python.

Ключевые слова: программирование, Python, обучение, основная школа, методы обучения, преимущества.

Keywords: programming, Python, training, basic school, teaching methods, advantages.

Целью обучения информатике в школе является формирование у учащихся трансформация компетенций по работе с информационными системами в глобальную цифровую систему образования; формирование компетенций по созданию среды программирования путем обучения фундаментальным математическим основам информатики [1].

Современное образование невозможно представить без внедрения информационных технологий и обучения программированию. Одним из наиболее популярных и востребованных языков программирования сегодня является Python. Благодаря своей простоте и универсальности, Python идеально подходит для обучения школьников.

Учащиеся при изучении программы различают программное и аппаратное обеспечение, необходимое для реализации информационных процессов для решения проблемы [2].

Для успешного начала обучения Python необходимо сначала познакомить школьников с базовыми понятиями программирования и компьютеров. На этом этапе важно использовать наглядные материалы, такие как презентации и видео, чтобы объяснить, что

такое программирование и какие задачи оно решает. Простые и интересные задачи, например, создание простых алгоритмов или объяснение принципов работы компьютера, помогут ученикам понять, что программирование — это не только сложно, но и увлекательно. После введения в программирование можно перейти к изучению основ Python. На этом этапе важно объяснить следующие ключевые темы: Переменные и типы данных: строки, числа, списки; Операторы: арифметические и логические; Условные конструкции: if, else, elif; Циклы: for, while [3].

Для закрепления материала рекомендуется использовать демонстрации на экране, совместное решение задач и практические задания, где ученики сами пишут код. Игры и интерактивные задания, например, написание простых программ для рисования с использованием библиотеки Turtle, помогут сделать процесс обучения более интересным. Когда основы Python освоены, можно переходить к более сложным концепциям, таким как функции и модули. На этом этапе учащиеся учатся определять функции, передавать параметры и возвращать значения. Важно познакомить их с основными стандартными модулями, такими как math и random. Практические задания, требующие написания функций и использования нескольких модулей, помогут ученикам понять, как структурировать код и повторно использовать его части. Это также подготовит их к более сложным проектам в будущем. Следующий важный этап обучения — работа с файлами и обработка данных. Учащиеся должны научиться открывать и закрывать файлы, читать из них и записывать в них данные. Простейшая обработка данных, такая как поиск и сортировка, также входит в программу обучения. Задачи на чтение данных из файла и их обработку помогут закрепить эти навыки. Например, можно предложить ученикам создать программу для анализа данных о погоде, которая будет читать данные из файла и выводить статистику по температуре. Для того чтобы сделать процесс обучения еще более интересным, можно включить в программу создание графических интерфейсов. Библиотека Tkinter предоставляет отличные возможности для этого. Учащиеся могут научиться создавать окна, кнопки и метки, что позволит им создавать простые, но наглядные приложения. Примеры проектов включают создание калькулятора или простой игры. Эти проекты не только развивают навыки программирования, но и стимулируют творческое мышление и интерес к предмету.

Итоговые проекты — важный элемент обучения, который помогает закрепить полученные знания и применить их на практике. Индивидуальные или групповые проекты, включающие в себя все изученные темы, дают ученикам возможность проявить свои способности и продемонстрировать результаты своей работы. Презентация проектов перед классом стимулирует учеников и развивает их навыки общения и презентации.

Пример программы на языке Python, которая соответствует уровню восьмого класса в основной школе. Эта программа использует основные конструкции, такие как циклы, условные операторы и функции. Программа решает задачу угадывания числа.

Программа «Угадай число»

```
import random
def guess_number():
    # Компьютер загадывает случайное число от 1 до 100
    number_to_guess = random.randint(1, 100)
    attempts = 0
    print("Добро пожаловать в игру 'Угадай число!'")
    print("Я загадал число от 1 до 100. Попробуй угадать его.")
    while True:
```

```
# Пользователь вводит число
user_guess = int(input("Введите ваше предположение: "))
attempts += 1
# Проверяем, угадал ли пользователь число
if user_guess < number_to_guess:
    print("Загаданное число больше.")
elif user_guess > number_to_guess:
    print("Загаданное число меньше.")
else:
    print(f"Поздравляем! Вы угадали число за {attempts} попыток.")
    break
if __name__ == "__main__":
    guess_number()
```

1. Импорт модуля random: Модуль random используется для генерации случайного числа.

2. Функция guess_number: В этой функции реализована логика игры:

Компьютер загадывает случайное число от 1 до 100 с помощью функции random.randint(1, 100);

Инициализируется счётчик попыток attempts;

Программа приветствует пользователя и объясняет правила игры;

В цикле while пользователь вводит свои предположения;

Программа сравнивает введённое пользователем число с загаданным числом и даёт подсказки;

Если пользователь угадывает число, программа выводит сообщение о победе и количество попыток, после чего цикл прекращается с помощью команды break.

Запуск функции guess_number: Проверка if __name__ == "__main__" используется для того, чтобы функция guess_number запускалась только при непосредственном выполнении скрипта, а не при его импорте как модуля.

Эта программа поможет школьникам понять основные концепции программирования на языке Python, такие как: генерация случайных чисел; использование циклов и условных операторов; написание и использование функций; взаимодействие с пользователем через консольный ввод и вывод.

Использование программирования на языке Python развивает у учащихся: игровые элементы: использование игровых элементов и соревнований помогает повысить интерес учеников к программированию; творчество и самостоятельность: поощряйте учеников к творческому подходу и самостоятельному решению задач; регулярные проверки знаний: организовывайте тесты и контрольные работы для проверки знаний и умений.

Доступ к компьютерам: обеспечьте ученикам доступ к компьютерам и современным средам разработки, таким как IDLE или Jupyter Notebook.

Обучение программированию Python в основной школе — это важный шаг в подготовке учеников к современному миру, где информационные технологии играют ключевую роль. Применение описанных методов позволит заинтересовать школьников, развить их логическое и творческое мышление и дать им прочные знания, которые они смогут использовать в дальнейшем обучении и карьере.

Список литературы:

1. Предметный стандарт по «Информатике» для 5-9 классов общеобразовательных организаций Кыргызской Республики. Бишкек. 2023. 45 с.
2. Учебная программа по «Информатике» для 5-9 классов общеобразовательных организаций Кыргызской Республики. Бишкек. 2023. 33 с.
3. Тагаева Д. А., Талипов А. Т., Саипбекова С. Э. Изучение программирования python в средней школе - инновационный путь к цифровой грамотности // Бюллетень науки и практики. 2024. Т. 10. №6. С. 675-678. <https://doi.org/10.33619/2414-2948/103/80>

References:

1. Predmetnyi standart po "Informatike" dlya 5-9 klassov obshcheobrazovatel'nykh organizatsii Kyrgyzskoi Respubliki (2023). Bishkek. (in Russian).
2. Uchebnaya programma po "Informatike" dlya 5-9 klassov obshcheobrazovatel'nykh organizatsii Kyrgyzskoi Respubliki (2023). Bishkek. (in Russian).
3. Tagaeva, D., Talipov, A., & Saipbekova, S. (2024). Learning Python Programming in Middle School - an Innovative path to Digital Literacy. *Bulletin of Science and Practice*, 10(6), 675-678. (in Russian). <https://doi.org/10.33619/2414-2948/103/80>

*Работа поступила
в редакцию 12.07.2024 г.*

*Принята к публикации
20.07.2024 г.*

Ссылка для цитирования:

Талипов А. Т., Калдыбаев С. К. Методы обучения программированию Python в основной школе // Бюллетень науки и практики. 2024. Т. 10. №8. С. 438-441. <https://doi.org/10.33619/2414-2948/105/54>

Cite as (APA):

Talipov, A. & Kaldybaev, S. (2024). Methods of Teaching Python Programming in Basic School. *Bulletin of Science and Practice*, 10(8), 438-441. (in Russian). <https://doi.org/10.33619/2414-2948/105/54>