

Среда программирования. Конструкции языка программирования

Инструменты: Visual Studio, Python

На уроке мы узнаем

- Познакомимся со средой разработки Visual Studio и ее возможностями для работы с Python.
- Изучить основные и продвинутые конструкции языка программирования Python.
- Научимся создавать, отлаживать и запускать программы в Visual Studio

Задачи урока

- Научится, как настроить Visual Studio для работы с Python.
- Изучить основные конструкции Python: переменные, условные операторы, циклы, функции.
- Практическая работа: написание и отладка программы в Visual Studio

Введение в тему

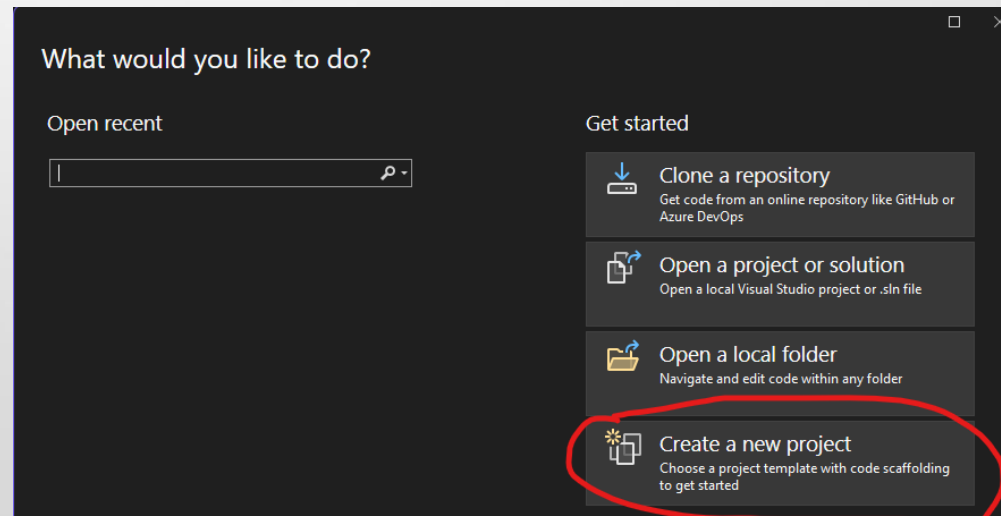
Краткий рассказ о том, что такое среда разработки и зачем она нужна

Настройка Visual Studio для работы с Python



- Запуск Visual Studio (нажать иконку Visual Studio на панели задач).
- Создание нового проекта Python: Выбрать "Create a new project"

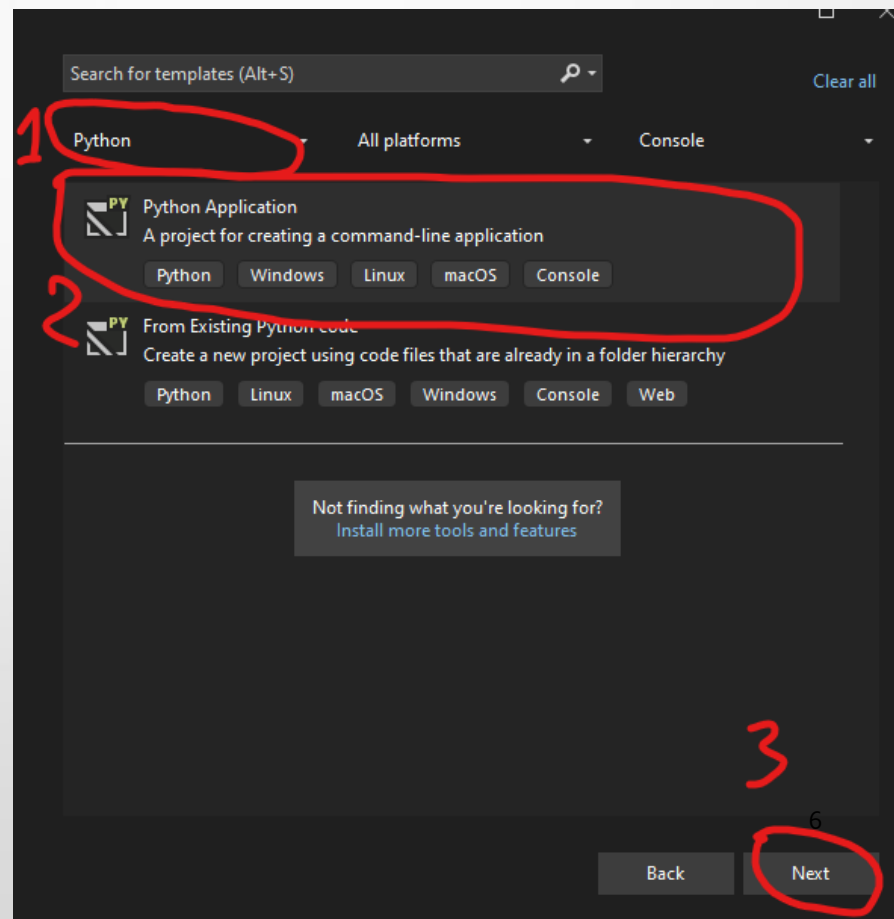
<https://github.com/agorbatniov/inf10>



Настройка Visual Studio для работы с Python (2)

1. Создание нового проекта Python: Выбрать "Python"
2. Выбрать "Python Application".
3. Нажать кнопку "Next"

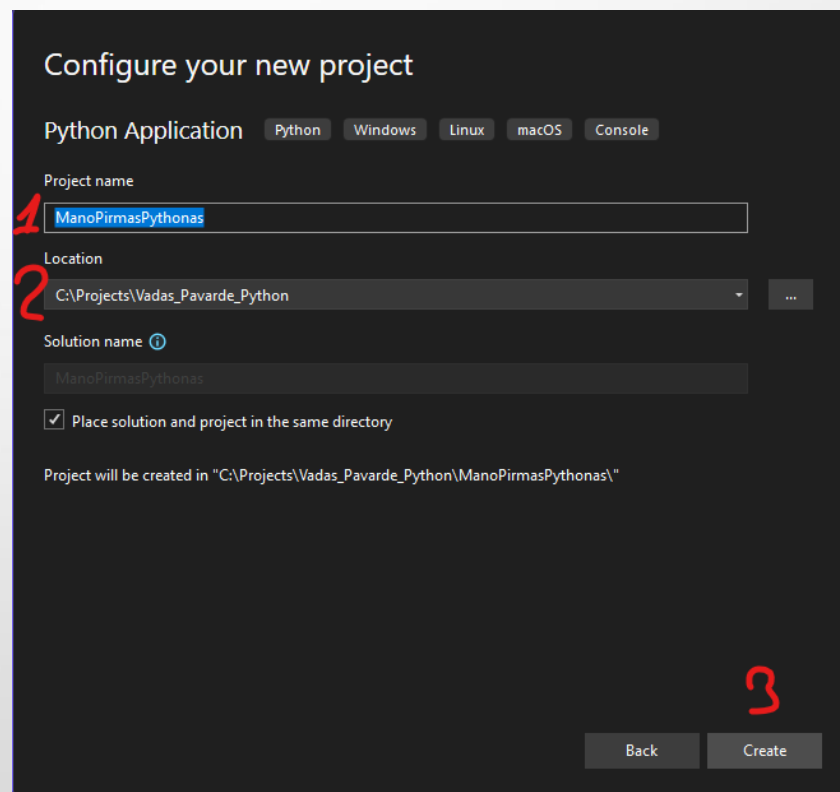
<https://github.com/agorbatniov/inf10>



Настройка Visual Studio для работы с Python (3)

1. Название проекта:
"ManoPirmasPythonas"
2. Место на диске: "
C:\Projects\Vadas_Pavarde_Pyt
hon".
3. Нажать кнопку "Next"

<https://github.com/agorbatniov/inf10>



Žingsniai derinimui

- Atidaryti projektą Visual Studio.
- Nustatyti breakpoint **F9** kodo eilutėje, kurioje norima pradėti analizę.
- Paleisti programą derinimo režime (Debug -> Start Debugging **F5**).
- Stebėti kintamųjų reikšmes naudojant Watch Window.
- Naudoti "Step Over **F10**" ir "Step Into **F11**" mygtukus, kad analizuotumėte kodo vykdymą.
- Surasti klaidą ir ją ištaisyti.

Основные и продвинутые конструкции языка Python

1. Переменные и типы данных:

- Пример: `x = 10`, `name = "Alice"`

2. Условные операторы:

- Конструкция `if-elif-else`.

<https://github.com/agorbatniov/inf10>

```
x = 10
if x > 10:
    print("x больше 10")
elif x == 10:
    print("x равно 10")
else:
    print("x меньше 10")
```

Основные и продвинутые конструкции языка Python (2)

3. Цикл for и while

```
for i in range(5):  
    print(i)
```

```
count = 0  
while count < 5:  
    print(count)  
    count += 1
```

4. Функции:

- Создание и вызов функций.

```
def greet(name):  
    print(f"Привет, {name}!")  
  
greet("Алексей")
```

Практическая работа

Задание:

Написать программу, которая:

- Запрашивает у пользователя число n .
- Вычисляет сумму квадратов всех чисел от 1 до n .
- Выводит результат.

Пример работы программы:

Введите число: 3

Сумма квадратов чисел от 1 до 3 равна 14

Объяснение

Для числа $n = 3$ программа должна вычислить:

$$1^2 + 2^2 + 3^2 = 1 + 4 + 9 = 14$$

Вопросы для закрепления

- Какие возможности Visual Studio мы изучили?
- Какие конструкции языка Python мы использовали?
- Как отлаживать программу в Visual Studio?



Конец