

Лекция 3

Модули и оператор `import`.

Определение и подключение модулей.

Стандартная библиотека Python.

В программировании часто возникает необходимость **повторного использования кода**.

Например, если мы написали функцию для вычисления факториала, мы можем захотеть применять её в разных программах. Чтобы не копировать код вручную, в Python используется механизм **модулей**.

Модуль — это отдельный файл с расширением `.py`, в котором содержатся функции, классы и переменные.

Его можно подключить в любую программу с помощью оператора `import`.

2. Использование оператора `import`

Простейший пример:

```
import math
```

```
print(math.sqrt(16))    # 4.0  
print(math.pi)         # 3.141592653589793
```

Варианты подключения:

1. Импорт всего модуля:

```
2. import math  
3. print(math.cos(0))
```

2. Импорт конкретной функции/класса:

```
1. from math import sqrt  
2. print(sqrt(25))    # 5.0
```

3. Импорт с псевдонимом:

```
1. import numpy as np  
2. print(np.array([1, 2, 3]))
```

4. Импорт всех элементов модуля (не рекомендуется из-за конфликта имён):

```
1. from math import *  
2. print(sin(0))    # 0.0
```

3. Создание собственных модулей

Создадим файл **mymodule.py**:

```
def greet(name):  
    return f"Сәлем, {name}!"  
  
def square(x):  
    return x ** 2
```

В основной программе **main.py** подключаем модуль:

```
import mymodule  
  
print(mymodule.greet("Айжан"))  
print(mymodule.square(7))
```

При запуске будет:

```
Сәлем, Айжан!  
49
```

4. Пакеты

Если несколько модулей объединить в папку и добавить туда файл `__init__.py`, то это будет **пакет**.

Например:

```
mymath/  
    __init__.py  
    algebra.py  
    geometry.py
```

Подключение:

```
from mymath import algebra  
from mymath.geometry import area
```

5. Стандартная библиотека Python

Python обладает очень богатой стандартной библиотекой. Она включает модули для:

- **math** – математические функции;
- **random** – генерация случайных чисел;
- **datetime** – работа с датами и временем;
- **os** – доступ к файловой системе;
- **sys** – параметры интерпретатора;
- **json** – обработка формата JSON;
- **statistics** – среднее, дисперсия, медиана;
- **itertools** – работа с итераторами;
- **re** – регулярные выражения.

6. Примеры

Пример 1. Генерация случайного числа

```
import random
print(random.randint(1, 100))    # случайное число от 1 до
100
```

Пример 2. Работа с датами

```
import datetime
today = datetime.date.today()
print("Сегодня:", today)
```

Пример 3. Статистика

```
import statistics as st
data = [2, 5, 7, 3, 8, 10]
print("Среднее:", st.mean(data))
print("Медиана:", st.median(data))
```

Пример 4. Работа с файлами

```
import os  
print(os.getcwd())    # текущая рабочая директория
```

7. Заключение

- Модуль в Python – это файл с функциями и классами.
- Для подключения используется оператор `import`.
- Мы можем создавать **собственные модули и пакеты**.
- Python имеет огромную **стандартную библиотеку**, которая позволяет решать задачи из самых разных областей