

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ  
АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО  
ОБРАЗОВАНИЯ «СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»  
ИНСТИТУТ ЦИФРОВОГО РАЗВИТИЯ**

**Отчёт о лабораторной работе №1 по дисциплине основы программной  
инженерии**

Выполнила:

Нестеренко Тамара Антоновна,  
2 курс, группа ПИЖ-б-о-20-1,

Проверил:

Доцент кафедры инфокоммуникаций,  
Воронкин Р.А.

Ставрополь, 2022 г.

## ВЫПОЛНЕНИЕ

### 1. Практическая часть

```
>>> import math
>>> math.factorial(5)
120
```

Рисунок 1 – Пример импорта и использования модуля math, который содержит математические функции

```
>>> import math, datetime
>>> math.cos(math.pi/4)
0.7071067811865476
>>> datetime.date(2017, 3, 21)
datetime.date(2017, 3, 21)
```

Рисунок 2 – Пример использования нескольких модулей

```
>>> import math as m
>>> m.sin(m.pi/3)
0.8660254037844386
```

Рисунок 3 – Пример задания псевдонима

```
>>> from math import cos
>>> cos(3.14)
-0.9999987317275395
```

Рисунок 4 – Пример импортирования отдельной функции

```
>>> from math import cos
>>> cos(3.14)
-0.9999987317275395
>>> sin(3.14)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'sin' is not defined. Did you mean: 'bin'?
>>> math.sin(3.14)
0.0015926529164868282
```

Рисунок 5 – Пример импорта конкретного объекта

```
>>> from math import cos, sin, pi
>>> cos(pi/3)
0.5000000000000001
>>> sin(pi/3)
0.8660254037844386
```

Рисунок 6 – Пример импортирования нескольких функций из модуля

```
>>> from math import factorial as f
>>> f(4)
24
```

Рисунок 7 – Пример задания псевдонима импортируемому объекту

```
>>> from math import *
>>> cos(pi/2)
6.123233995736766e-17
>>> sin(pi/4)
0.7071067811865476
>>> factorial(6)
720
```

Рисунок 8 – Пример импортирования всех функций и классов

```
import fincalc.simper
fv = fincalc.simper.fv(pv, i, n)
import fincalc.simper as sp
fv = sp.fv(pv, i, n)
from fincalc import simper
fv = simper.fv(pv, i, n)
```

Рисунок 9 – Пример использования пакетов

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

import sys

if __name__ == "__main__":
    x = input("Введите строку: ")
    c = input("Введите символ, который нужно заменить: ")
    h = input("Введите символ, на который заменить: ")
    import fun1 as f
    rep = f.fun1(h, c, x)
    print(rep)
```

Рисунок 10 – Пример решения индивидуального задания №1. Основной файл

```

# !/usr/bin/env python3
# -*- coding: utf-8 -*-

import sys

def fun1(to_replace, replacer, string):
    result = string.replace(replacer, to_replace)
    return result

```

Рисунок 11 – Пример решения индивидуального задания №1. Модуль

```

#!/usr/bin/env python3
# -*- coding: utf-8 -*-

import ...

if __name__ == '__main__':
    workers = []

    while True:

        command = input(">>> ").lower()

        if command == 'exit':
            break

        elif command == 'add':
            worker = get_worker()

            workers.append(worker)
            if len(workers) > 1:
                workers.sort(key=lambda item: item.get('name', ''))

        elif command == 'list':
            display_workers(workers)

        elif command.startswith('select'):
            selected = select_workers(workers)
            display_workers(selected)

        elif command == 'help':
            print("Список команд:\n")

```

Рисунок 12 – Пример решения индивидуального задания №2. Основной файл

```
__all__ = ["get_worker", "display_workers", "select_workers"]
```

Рисунок 13 – Пример решения индивидуального задания №2. Файл `__init__.py`

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

import ...

def display_workers(staff):
    if staff:

        line = '+-{}-+-{}-+-{}-+-{}-+-{}-+'.format(
            '-' * 4,
            '-' * 30,
            '-' * 20,
            '-' * 15,
            '-' * 15
        )
        print(line)
        print(
            '| {:^4} | {:^30} | {:^20} | {:^15} | {:^15} |'.format(
                "№",
                "Фамилия",
                "Имя",
                "Знак зодиака",
                "Дата рождения"
            )
        )
        print(line)

        # Вывести данные о всех сотрудниках.
        for idx, worker in enumerate(staff, 1):
            print(
                '| {:^4} | {:^30} | {:^20} | {:^15} | {:^15} |'.format(
```

Рисунок 14 – Пример решения индивидуального задания №2. Модуль `display_workers`

```

#!/usr/bin/env python3
# -*- coding: utf-8 -*-

import ...

def get_worker():
    surname = input("Фамилия: ")
    name = input("Имя: ")
    zodiac = input("Знак зодиака: ")
    date = input("Дата: ")

    return {
        'surname': surname,
        'name': name,
        'zodiac': zodiac,
        'date': datetime.strptime(date, "%Y-%m-%d")
    }

```

Рисунок 15 – Пример решения индивидуального задания №2. Модуль get\_worker

```

#!/usr/bin/env python3
# -*- coding: utf-8 -*-

import ...

def select_workers(staff):
    month1 = int(input("Введите месяц: "))
    result = []
    for worker in staff:
        if worker.get('date', '').month == month1:
            result.append(worker)
    return result

```

Рисунок 16 – Пример решения индивидуального задания №2. Модуль select\_workers

## 2. Вопросы для защиты

### 1. Что является модулем языка Python?

Под модулем в Python понимается файл с расширением .py. Модули предназначены для того, чтобы в них хранить часто используемые функции, классы, константы и т.п.

### 2. Какие существуют способы подключения модулей в языке Python?

```

import math
import math, datetime
import math as m
from math import cos

```

```
from math import cos, sin
from math import factorial as f
from math import *
```

### 3. Что является пакетом языка Python?

Пакет – это каталог, включающий в себя другие каталоги и модули, но при этом дополнительно содержащий файл `__init__.py`. Пакеты используются для формирования пространства имён, что позволяет работать с модулями через указание уровня вложенности (через точку).

### 4. Каково назначение файла `__init__.py`?

Файл `__init__.py` может быть пустым или может содержать переменную `__all__`, хранящую список модулей, который импортируется при загрузке через конструкцию.

Пример: `from имя_пакета import *`

### 5. Каково назначение переменной `__all__` файла `__init__.py`?

Пример: `__all__ = ["simper", "compper", "annuity"]`

Ссылка на репозиторий: [https://github.com/tamaranesterenko/Python\\_LR\\_1-2](https://github.com/tamaranesterenko/Python_LR_1-2)