**МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Ордена трудового Красного Знамени федеральное государственное бюджетное**

**образовательное учреждение высшего образования**

**«Московский технический университет связи и информатики»**

Кафедра Программная инженерия

Отчет по лабораторной работе №4

“ **Модули и пакеты: импорт, создание, использование** ”

по дисциплине «Введение в информационные технологии»

Выполнил: студент группы

БПИ2501

Карпеко Никита Александрович

Проверила: Мосева Марина Сергеевна

Москва, 2025

**Цель работы**

Понять, как импортировать модули и пакеты в Python, научиться создавать собственные модули и пакеты, изучить способы использования модулей и пакетов для структурирования программы.

**Задания**

**1. Импорт стандартных модулей**

1.1. Импортировать модуль **math** и использовать функцию **sqrt()** для вычисления квадратного корня.

1.2. Использовать модуль **datetime** для отображения текущей даты и времени.

**2. Создание и использование собственного модуля**

2.1. Создать модуль **my\_module.py**, который содержит минимум одну функцию. Например, функция может принимать два аргумента и возвращать их сумму.

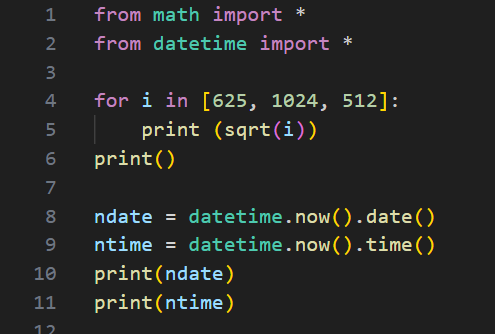
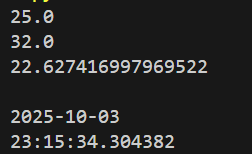
2.2. Импортировать **my\_module** в другой файл Python и вызвать функцию, определённую в модуле.

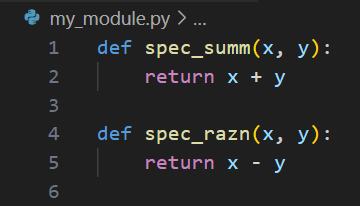
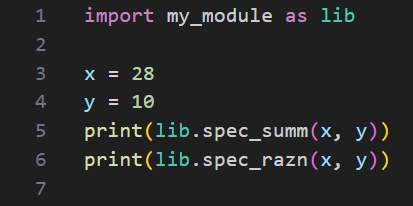
**3. Создание и использование пакетов**

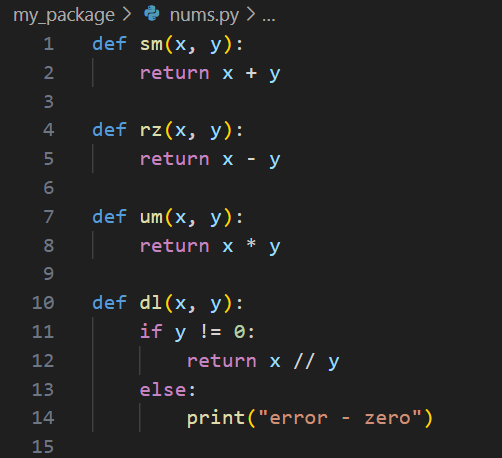
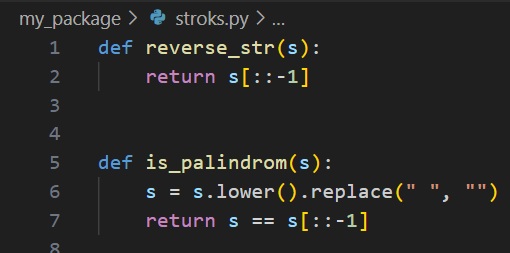
3.1. Создать пакет, содержащий несколько модулей. Каждый модуль должен выполнять определённую задачу (например, операции с числами, работа со строками и т.д.).

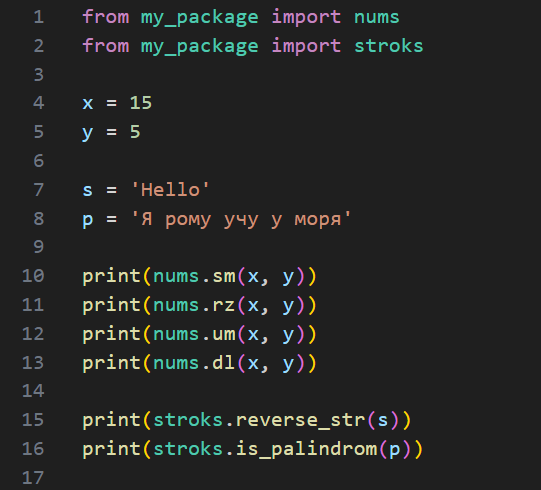
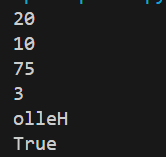
3.2. Продемонстрировать, как импортировать различные модули из вашего пакета в другой файл Python.

**Скриншоты выполнения (группы – код и вывод выполнения)**



**Исходный код**

from math import \*

from datetime import \*

for i in [625, 1024, 512]:

print (sqrt(i))

print()

ndate = datetime.now().date()

ntime = datetime.now().time()

print(ndate)

print(ntime)

#модуль my\_module

def spec\_summ(x, y):

return x + y

def spec\_razn(x, y):

return x – y

import my\_module as lib

x = 28

y = 10

print(lib.spec\_summ(x, y))

print(lib.spec\_razn(x, y))

#nums

def sm(x, y):

    return x + y

def rz(x, y):

    return x - y

def um(x, y):

    return x \* y

def dl(x, y):

    if y != 0:

        return x // y

    else:

        print("error - zero")

#stroks

def reverse\_str(s):

    return s[::-1]

def is\_palindrom(s):

    s = s.lower().replace(" ", "")

    return s == s[::-1]

from my\_package import nums

from my\_package import stroks

x = 15

y = 5

s = 'Hello'

p = 'Я рому учу у моря'

print(nums.sm(x, y))

print(nums.rz(x, y))

print(nums.um(x, y))

print(nums.dl(x, y))

print(stroks.reverse\_str(s))

print(stroks.is\_palindrom(p))

**Заключение**

Были изучены способы импортирования модулей и пакетов в Python, освоено создание собственных модулей и пакетов, изучены способы использования модулей и пакетов для структурирования программы.