**Санкт-Петербургский Государственный Технологический Институт (Технологический Университет)**

Кафедра Системного Анализа и Информационных Технологий

Лабораторная работа №5

Выполнили:

Смелковская Е. С.

Щедрина Е. И.

Кораблев Д. В.

Проверил:

Мусаев А.А.

Санкт-Петербург,

2023

Ссылка на гитхаб: https://github.com/KorablikDimak/Labs/tree/main/lab5

**Задание 1**

1. Создать модуль под названием "math\_operations". В этом модуле определить функции для выполнения основных математических операций: сложение, вычитание, умножение и деление. Вызвать функции из основной программы и ознакомиться с результатами.

2. Создать модуль под названием "temperature\_conversion". В этом модуле определить функции для конвертации температуры из градусов Цельсия в градусы Фаренгейта и наоборот. Импортировать эти функции в основную программу и проверить их работу, вызывая их с разными значениями температуры.

3. Создать модуль под названием "validation". В этом модуле определить функцию для проверки введенного пользователем числа на четность. Используя эту функцию, импортировать модуль в основную программу и просить пользователя ввести число, а затем проверить, является ли оно четным или нечетным.

4. Создать модуль под названием "geometry". В этом модуле определить функции для вычисления площади прямоугольника, круга и треугольника. Импортировать эти функции в основную программу и вызвать их для разных фигур, передавая нужные параметры.

Код модуля "math\_operations":

def add(left, right):

return left + right

def sub(left, right):

return left - right

def mul(left, right):

return left \* right

def div(left, right):

return left / right

Код модуля "temperature\_conversion":

def to\_fahrenheit(celsius: float) -> float:

return celsius \* 9 / 5 + 32

def to\_celsius(fahrenheit: float) -> float:

return (fahrenheit - 32) \* 5 / 9

Код модуля "validation":

def is\_even(number: int) -> bool:

return number % 2 == 0

Код модуля "geometry":

from math import pi

def circle\_square(radius: float) -> float:

return pi \* radius \* radius

def rectangle\_square(length: float, width: float) -> float:

return length \* width

def triangle\_square(base: float, height: float) -> float:

return 0.5 \* base \* height

Вывод

Мы научились создавать модули в python для последующего использования. Разделять функциональность на модули, логически комбинируя функции в определенные модули.

**Задание 2**

Изучить библиотеку NumPy (https://numpy.org/). Установить её с помощью инструкции pip install.

1. Создать одномерный массив с числами от 1 до 10 и вывести его на экран.

2. Создать двумерный массив размером 3x3, состоящий из случайных чисел и вывести его на экран.

3. Создать массив состоящий из 10 нулей и изменить значение третьего элемента на 5.

4. Создать массив, содержащий числа от 1 до 12, и преобразовать его в двумерный массив размером 3x4.

5. Создать массив размером 5x5, заполненный случайными числами от 0 до 9. Найти сумму всех элементов этого массива.

6. Создать массив размером 4x4, заполненный случайными числами от 1 до 10, найти максимальное значение в каждой строке и вывести их на экран.

7. Создать массив размером 3x3, заполненный случайными числами. Умножить его на 2 и вывести результат.

8. Создать два массива размером 3x3 с числами от 0 до 8. Посчитать их матричное произведение и вывести результат.

9. Создать массив размером 3x3, заполненный случайными числами от 0 до 5. Найти определитель матрицы.

Код программы для задания 2:

Вывод

Изучили библиотеку numpy и освоили ее основные функции. Научились использовать массивы numpy для выполнения различных задач.

**Задание 3**

Изучить библиотеку Matplotlib (https://matplotlib.org/). Установить её с помощью инструкции pip install.

1. Создать линейный график, отображающий изменение температуры в течение дня (ось X - время, ось Y - температура). Использовать данные в виде списков или массивов.

2. Создать столбчатую диаграмму, сравнивающую продажи по разным категориям товаров. Ось X должна содержать названия категорий, а ось Y - количество продаж. Использовать случайные данные в виде списков.

3. Отобразить точечную диаграмму, показывающую зависимость роста людей от их веса. Использовать случайные данные в виде списков.

4. Отобразить гистограмму, показывающую распределение оценок учеников по математике. Ось X должна содержать промежутки оценок, а ось Y - количество учеников.

5. Создать двумерный график функции z = f(x, y), где x и y - массивы значений, а z - результат функции. Использовать функцию meshgrid для создания массивов x и y.

Код программы для задания 3 пункты 1-4:

import matplotlib.pyplot as plt

import random

plt.figure(figsize=(10,10))

time = [x for x in range(0, 24)]

temperature = [-11.0, -11.2, -11.3, -11.5, -10.9, -10.7,

-10.7, -10.6, -10.4, -10.2, -10.1, -10.0,

-9.8, -9.7, -9.8, -9.7, -10.0, -10.1,

-10.4, -10.5, -10.7, -10.7, -10.9, -11.3]

plt.subplot(221)

plt.plot(time, temperature)

plt.title("график температуры")

plt.xlabel("время, ч")

plt.ylabel("температура, градусы цельсия")

plt.ylim(-13, -8)

categories = ["бакалея", "молочка", "мясо", "напитки", "рыба"]

total = [random.randint(100, 500) for y in range(5)]

plt.subplot(222)

plt.bar(categories, total)

plt.title("продажи")

plt.xlabel("категория")

plt.ylabel("количество, шт")

weight = []

height = [random.randint(150, 200) for y in range(100)]

for i in range(100):

weight.append(height[i] + random.randint(-30, 30))

plt.subplot(223)

plt.scatter(height, weight)

plt.title("зависимость роста от веса")

plt.xlabel("вес, кг")

plt.ylabel("рост, см")

marks = [random.randint(1, 10) for y in range(1000)]

plt.subplot(224)

plt.hist(marks)

plt.title("распределение оценок")

plt.xlabel("оценка")

plt.ylabel("количество учеников")

plt.show()

Код программы для задания 3 пункт 5:

import matplotlib.pyplot as plt

import numpy as np

x = np.linspace(-10, 10, 100)

y = np.linspace(-10, 10, 100)

xgrid, ygrid = np.meshgrid(x, y)

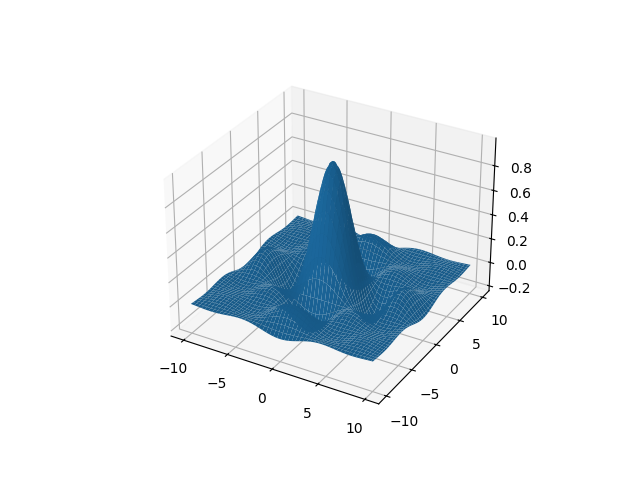
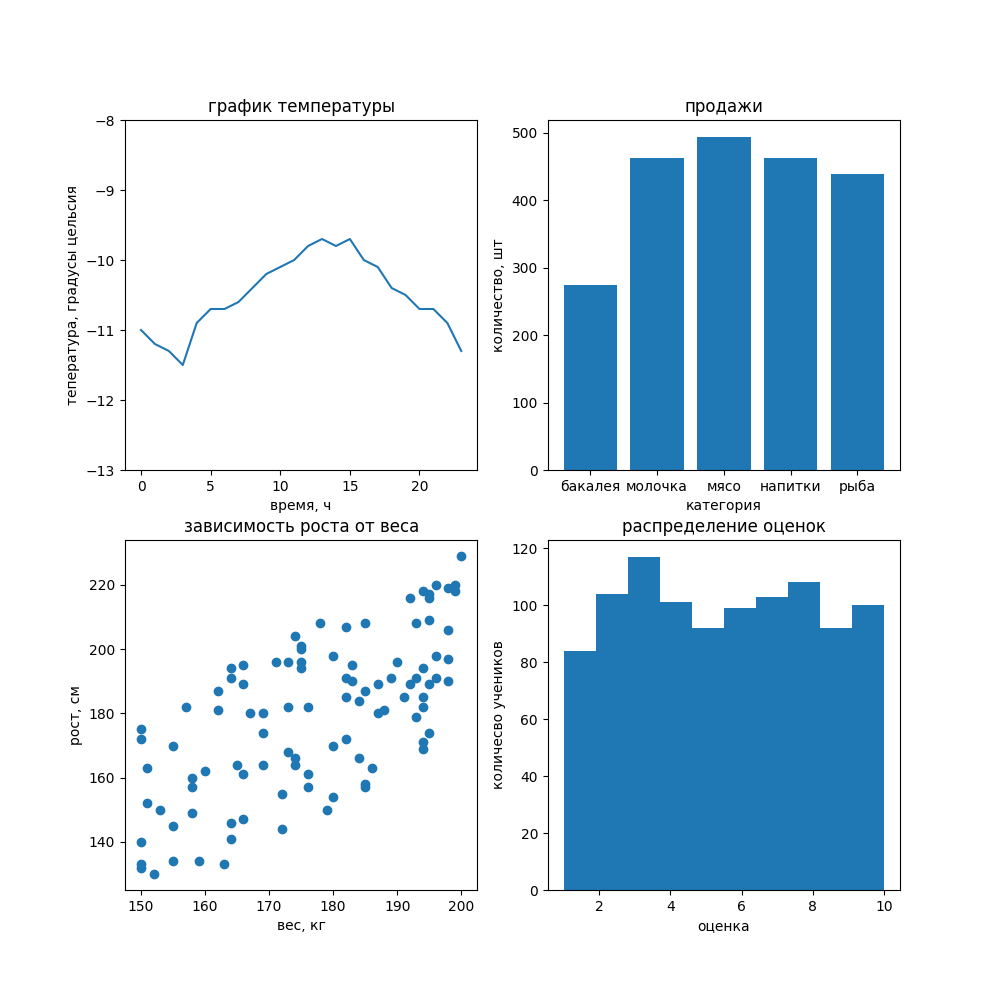
zgrid = np.sin(xgrid) \* np.sin(ygrid) / (xgrid \* ygrid)

fig = plt.figure()

axes = fig.add\_subplot(projection='3d')

axes.plot\_surface(xgrid, ygrid, zgrid)

plt.show()

Результат работы программы:

Вывод

Изучили библиотеку matplotlib, ее основные функции, с помощью которых построили различные виды графиков. Ознакомились со взаимодействием данной библиотеки с библиотекой numpy.