|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | |  |  |  | | МИНОБРНАУКИ РОССИИ | | | | Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** | | | | |
| Институт информационных технологий (ИИТ) | |
| Кафедра Прикладной Математики (ПМ) | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ОТЧЁТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ №1** | | | |
| **по дисциплине «Технологии и инструментарий анализа больших данных»** | | | |
|  | | | |
| Выполнил студент группы ИКБО-14-20 | | Вежновец Ф. Ю. | |
|  | |  | |
| Принял: асистент | | Горячев А. А. | |
| Практические работы выполнены | «\_\_\_»\_\_\_\_\_\_\_2023 г. | | (подпись студента) | |
| «Зачтено» | «\_\_\_»\_\_\_\_\_\_\_2023 г. | | (подпись руководителя) | |
|  |  | |  | |

Москва 2023

Задание 1

Установить Python, если это не было сделано ранее.

Реализация

Python был установлен ранее (Рисунок 1).

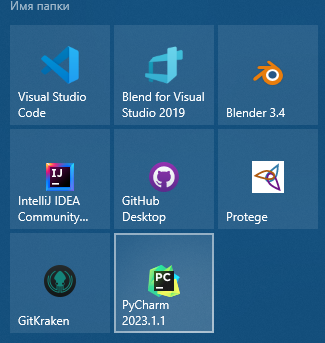


Рисунок 1 – установка python

Задание 2

Написать программу, которая вычисляет площадь фигуры, параметры которой подаются на вход. Фигуры, которые подаются на вход: треугольник, прямоугольник, круг. Результатом работы является словарь, где ключ – это название фигуры, а значение – это площадь.

Реализация

Для решения задачи был написан код (Листинг 1) Результат работы программы представлен на рисунке 2.

|  |  |
| --- | --- |
|  |  |
|  |  |

Рисунок 2 – Результат работы программы

Листинг 1

import math

figures = {}

name = input('Введите нзвание фигуры: ')

area = None

if name.lower() == 'треугольник':

height = float(input('Введите высоту треугольника: '))

base = float(input('Введите длину основания треугольника: '))

area = 0.5 \* base \* height

elif name.lower() == 'прямоугольник':

length = float(input("Введите длину прямоугольника: "))

width = float(input("Введите ширину прямоугольника: "))

area = length \* width

elif name.lower() == 'круг':

radius = float(input("Введите радиус круга: "))

area = math.pi \* radius \*\* 2

else:

print("Неверный тип фигуры")

if area != None:

figures[name] = area

print(f"\nПлощадь {name} равна {area}\n")

print(figures)

Задание 3

Написать программу, которая на вход получает два числа и операцию, которую к ним нужно применить. Должны быть реализованы следующие операции: +, -, /, //, abs – модуль, pow или \*\* – возведение в степень.

Реализация

Для решения задачи был написан код (Листинг 2) Результат работы программы представлен на рисунке 3.

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |

Рисунок 3 – Результат работы программы

Листинг 2

num1 = float(input('Введите первое число: '))

num2 = float(input('Введите второе число: '))

operation = input("Введите операцию (+, -, /, //, abs, pow): ")

result = 0.0

if operation == "+":

result = num1 + num2

elif operation == "-":

result = num1 - num2

elif operation == "/":

if num2 == 0:

print("Ошибка: деление на ноль!")

else:

result = num1 / num2

elif operation == "//":

if num2 == 0:

print("Ошибка: деление на ноль!")

else:

result = num1 // num2

elif operation == "abs":

result = abs(num1)

elif operation == "pow" or operation == "\*\*":

result = num1 \*\* num2

else:

result = "Неверная операция!"

print(f"Результат операции {operation}: {result}")

Задание 4

Напишите программу, которая считывает с консоли числа (по одному в строке) до тех пор, пока сумма введённых чисел не будет равна 0 и после этого выводит сумму квадратов всех считанных чисел.

Реализация

Для решения задачи был написан код (Листинг 3) Результат работы программы представлен на рисунке 4.

|  |
| --- |
|  |
|  |
|  |

Рисунок 4 – Результат работы программы

Листинг 3

array = []

while len(array) == 0 or sum(array) != 0: # длина = 0, или сумма != 0

x = float(input('Введите число: '))

array.append(x)

else:

sum\_2 = sum(x \*\* 2 for x in array)

print(f'Сумма квадратов введённых чисел = {sum\_2}')

Задание 5

Напишите программу, которая выводит последовательность чисел, длинною N, где каждое число повторяется столько раз, чему оно равно. На вход программе передаётся неотрицательное целое число N. Например, если N = 7, то программа должна вывести 1 2 2 3 3 3 4. Вывод элементов списка через пробел – print(\*list).

Реализация

Для решения задачи был написан код (Листинг 4) Результат работы программы представлен на рисунке 5.

|  |
| --- |
|  |

Рисунок 5 – Результат работы программы

Листинг 4

N = int(input('Введите N: '))

array = []

for i in range(1, N + 1):

for j in range(i):

if len(array) != N:

array.append(i)

print(\*array)

Задание 6

Даны два списка:

А = [1, 2, 3, 4, 2, 1, 3, 4, 5, 6, 5, 4, 3, 2]

В = [‘a’, ’b’, ’c’, ’c’, ’c’, ’b’, ’a’, ’c’, ’a’, ’a’, ’b’, ’c’, ’b’, ’a’]

Создать словарь, в котором ключи – это содержимое списка В, а значения для ключей словаря – это сумма всех элементов списка А в соответствии с буквой, содержащийся на той же позиции в списке В. Пример результата программы: {‘a’ : 10, ‘b’ : 15, ‘c’ : 6}.

Реализация

Для решения задачи был написан код (Листинг 5) Результат работы программы представлен на рисунке 6.

|  |
| --- |
|  |

Рисунок 6 – Результат работы программы

Листинг 5

A = [1, 2, 3, 4, 2, 1, 3, 4, 5, 6, 5, 4, 3, 2]

B = ['a', 'b', 'c', 'c', 'c', 'b', 'a', 'c', 'a', 'a', 'b', 'c', 'b', 'a']

result\_dict = {}

for i in range(len(B)):

key = B[i]

value = A[i]

if key in result\_dict:

result\_dict[key] += value

else:

result\_dict[key] = value

print(result\_dict)

Задание 7

Скачать и загрузить данные о стоимости домов в калифорнии, используя библиотеку sklearn.

Реализация

Для решения задачи был написан код (Листинг 6) Результат работы программы представлен на рисунке 7.

|  |
| --- |
|  |

Рисунок 7 – Результат работы программы

Листинг 6

from sklearn.datasets import fetch\_california\_housing

data = fetch\_california\_housing(as\_frame=True)

dataset = data.frame

dataset

Задание 8

Использовать метод info().

Реализация

Для решения задачи был написан код (Листинг 7) Результат работы программы представлен на рисунке 8.

|  |
| --- |
|  |

Рисунок 8 – Результат работы программы

Листинг 7

dataset.info()

Задание 9

Узнать, есть ли пропущенные значения, используя isna().sum().

Реализация

Для решения задачи был написан код (Листинг 8) Результат работы программы представлен на рисунке 9.

|  |
| --- |
|  |

Рисунок 9 – Результат работы программы

Листинг 8

dataset.isna().sum()

Задание 10

Вывести записи, где средний возраст домов в районе более 50 лет и население более 2500 человек, используя метод loc().

Реализация

Для решения задачи был написан код (Листинг 9) Результат работы программы представлен на рисунке 10.

|  |
| --- |
|  |

Рисунок 10 – Результат работы программы

Листинг 9

dataset.loc[(dataset['HouseAge'] > 50) & (dataset['Population'] > 2500)]

Задание 11

Узнать максимальное и минимальное значения медианной стоимости дома.

Реализация

Для решения задачи был написан код (Листинг 10) Результат работы программы представлен на рисунке 11.

|  |
| --- |
|  |

Рисунок 11 – Результат работы программы

Листинг 10

dataset['MedHouseVal'].describe()

Задание 12

Используя метод apply(), вывести на экран название признака и его среднее значение.

Реализация

Для решения задачи был написан код (Листинг 11) Результат работы программы представлен на рисунке 12.

|  |
| --- |
|  |

Рисунок 12 – Результат работы программы

Листинг 11

def calculate\_mean(column):

return column.mean()

means = dataset.apply(calculate\_mean)

for feature, mean\_value in means.items():

print(f"Признак: {feature}, Среднее значение: {mean\_value}")

Задание со звёздочкой 1

Дан текст на английском языке. Необходимо закодировать его с помощью азбуки Морзе, где каждой букве соответствует последовательность точек и тире. Например, буква «g» превратится в строку «--.». В переменной morze для удобства хранится словарь соответствия латинских букв коду Морзе.

morze = {'a': '.-', 'b': '-…', 'c': '-.-.', 'd': '-..',

'e': '.', 'f': '..-.', 'g': '--.', 'h': '….',

'i': '..', 'j': '.---', 'k': '-.-', 'l': '.-..',

'm': '--', 'n': '-.', 'o': '---', 'p': '.--.',

'q': '--.-', 'r': '.-.', 's': '…', 't': '-',

'u': '..-', 'v': '…-', 'w': '.--', 'x': '-..-',

'y': '-.--', 'z': '--..'}

На входе: В одной строке вам дан текст, который состоит из латинских букв и пробелов.

На выходе:

Выведите каждое слово исходного текста, закодированное азбукой Морзе. Количество строк в ответе должно совпадать с количеством слов в исходном тексте. Между закодированными буквами ставится ровно один пробел. Например, слово «Help» превратится в «.... . .-.. .--.».

Реализация

Листинг кода (листинг 12). Рисунок 13.

*Листинг 12*

morze = {

'a': '.-', 'b': '-...', 'c': '-.-.', 'd': '-..',

'e': '.', 'f': '..-.', 'g': '--.', 'h': '....',

'i': '..', 'j': '.---', 'k': '-.-', 'l': '.-..',

'm': '--', 'n': '-.', 'o': '---', 'p': '.--.',

'q': '--.-', 'r': '.-.', 's': '...', 't': '-',

'u': '..-', 'v': '...-', 'w': '.--', 'x': '-..-',

'y': '-.--', 'z': '--..', ' ': ' '

}

def text\_to\_morse(text):

words = text.split()

encoded\_words = []

for word in words:

encoded\_word = ' '.join([morze[char.lower()] for char in word])

encoded\_words.append(encoded\_word)

return encoded\_words

input\_text = input("Введите текст на английском языке: ")

encoded\_text = text\_to\_morse(input\_text)

for word in encoded\_text:

print(word)

\

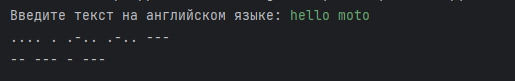


Рисунок 13 – результат работы программы

Задание со звёздочкой 2

В некотором городе открывается новая служба по доставке электронных писем. Необходимо наладить систему регистрации новых пользователей.

Регистрация должна работать следующим образом: если новый пользователь хочет зарегистрироваться на сайте, то он должен послать системе запрос name со своим именем. Система должна определить, существует ли уже такое имя в базе данных. Если такого имени не существует, то оно заносится в базу данных системы и пользователю возвращается ответ "ОК", подтверждающий успешную регистрацию. А если пользователь с таким именем уже существует, то система должна сформировать новое имя и выдать его пользователю в качестве подсказки, при этом сама подсказка также добавляется в базу данных. Новое имя формируется следующим образом: к name последовательно приписываются числа, начиная с 1 (name1, name2 и так далее), и среди них находят такое наименьшее i, что namei еще не содержится в системе.

Входные данные

В первой строке входных данных задано число n (1 ≤ n ≤ 100000). Следующие n строк содержат запросы к системе. Каждый запрос представляет собой непустую строку длиной не более 32 символов, состоящую только из строчных букв латинского алфавита.

Реализация

Листинг кода (листинг 13). Рисунок 14.

*Листинг 13*

registered\_users = []

count\_name = int(input())

for i in range(count\_name):

new\_name = str(input())

if new\_name not in registered\_users:

registered\_users.append(new\_name)

print('OK')

else:

for i in range(1, 100000):

find\_name = new\_name + str(i)

if find\_name not in registered\_users:

registered\_users.append(find\_name)

print(find\_name)

break

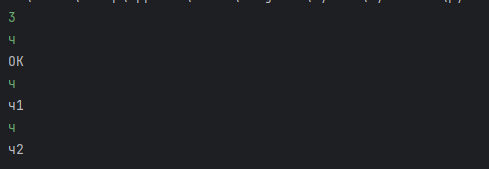


Рисунок 14 – результат работы программы

Задание со звёздочкой 3

Необходимо создать программу обработки запросов пользователей к файловой системе компьютера. Над каждым файлом можно производить следующие действия: запись – w ("write"), чтение – r ("read"), запуск – x ("execute").

Реализация

Листинг кода (листинг 14). Рисунок 15.

*Листинг 14*

array\_actions = {}

n = int(input())

for i in range(n):

requests = str(input()).split()

key = requests[0]

requests.pop(0)

for j in range(len(requests)):

if requests[j] == 'r':

requests[j] = 'read'

elif requests[j] == 'w':

requests[j] = 'write'

elif requests[j] == 'e':

requests[j] = 'execute'

array\_actions[key] = requests

print(array\_actions)

m = int(input())

for i in range(m):

requests = str(input()).split() # write 1.py

if requests[0] in array\_actions[requests[1]]:

print('OK')

else:

print('Access denied')

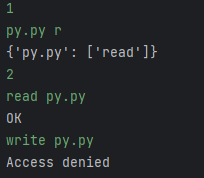


Рисунок 15 – результат работы программы