Задание 1. Описание: Напишите функцию datetime\_info(date\_str), которая принимает строку в формате 'YYYY-MM-DD' и возвращает словарь с информацией о дате и времени:

- Дата в формате 'ДД-ММ-ГГГГ'

- День недели

- Количество дней до следующего года

**from datetime import datetime**

**def datetime\_info(date\_str):**

**date\_obj = datetime.strptime(date\_str, '%Y-%m-%d')**

**formatted\_date = date\_obj.strftime('%d-%m-%Y')**

**weekday = date\_obj.strftime('%A')**

**next\_year = datetime(date\_obj.year + 1, 1, 1)**

**days\_until\_next\_year = (next\_year - date\_obj).days**

**return {**

**'formatted\_date': formatted\_date,**

**'weekday': weekday,**

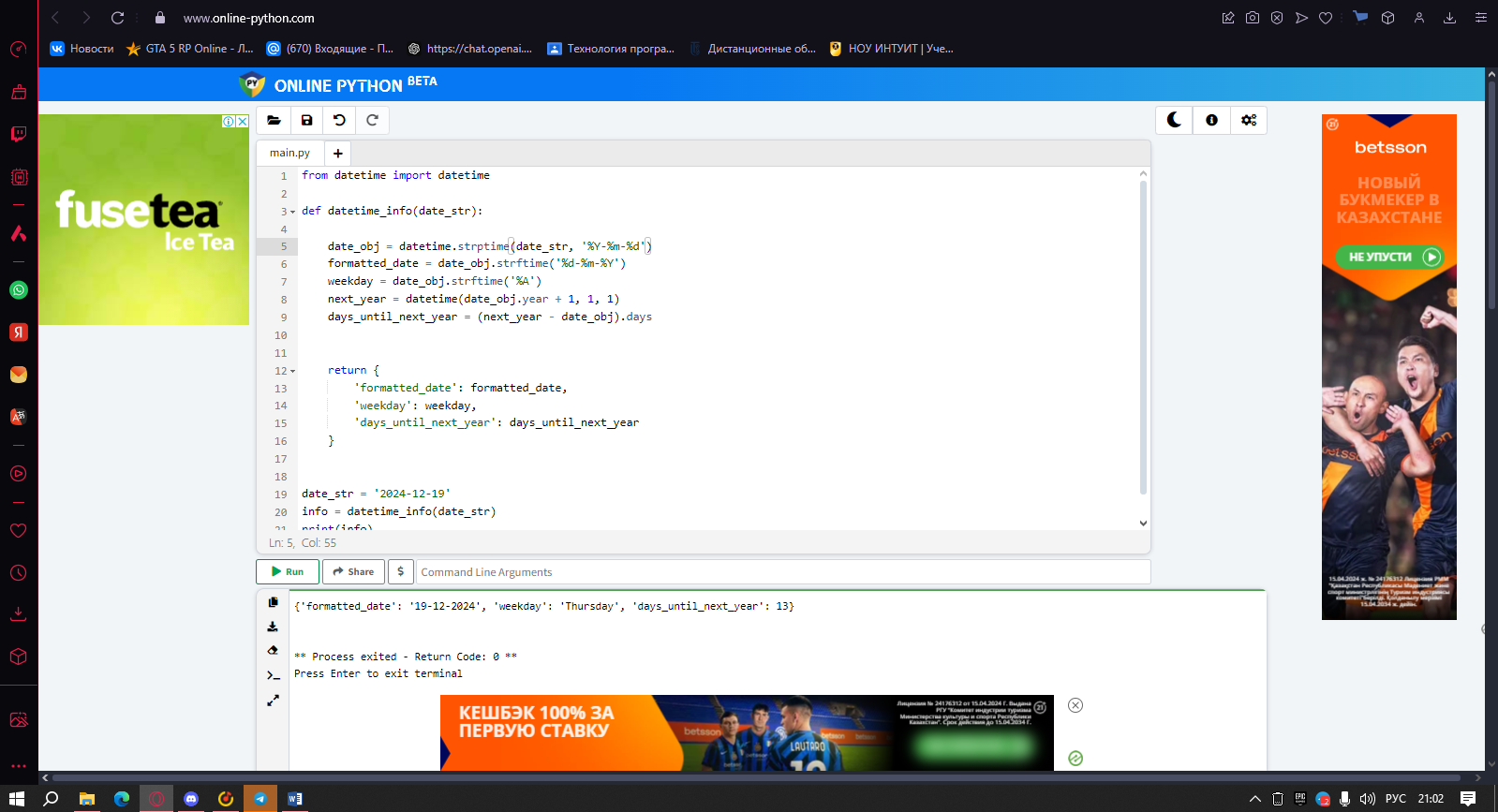
**'days\_until\_next\_year': days\_until\_next\_year**

**}**

**date\_str = '2024-12-19'**

**info = datetime\_info(date\_str)**

**print(info)**



Задание 2

Описание: Напишите функцию write\_and\_read\_file(filename, content), которая:

- Записывает строку content в файл с именем filename.

- Считывает содержимое файла и возвращает его.

**def write\_and\_read\_file(filename, content):**

**with open(filename, 'w') as file:**

**file.write(content)**

**with open(filename, 'r') as file:**

**file\_content = file.read()**

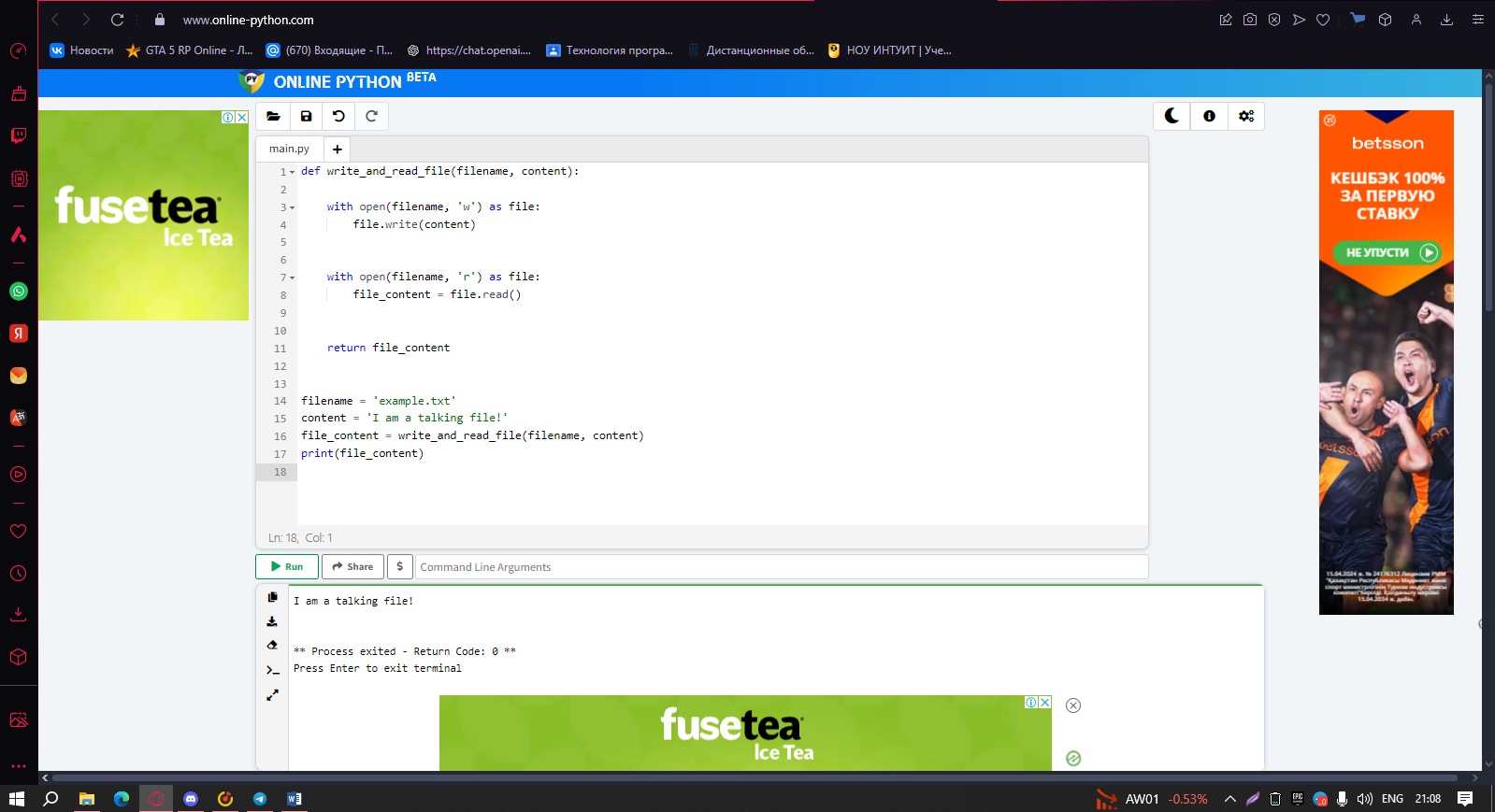
**return file\_content**

**filename = 'example.txt'**

**content = 'I am a talking file!'**

**file\_content = write\_and\_read\_file(filename, content)**

**print(file\_content)**



Задание 3

Напишите функцию list\_files\_in\_directory(directory\_path), которая принимает путь к каталогу и возвращает список всех файлов в этом каталоге (без подкаталогов).

**import os**

**def list\_files\_in\_directory(directory\_path):**

**files = []**

**for entry in os.listdir(directory\_path):**

**full\_path = os.path.join(directory\_path, entry)**

**if os.path.isfile(full\_path):**

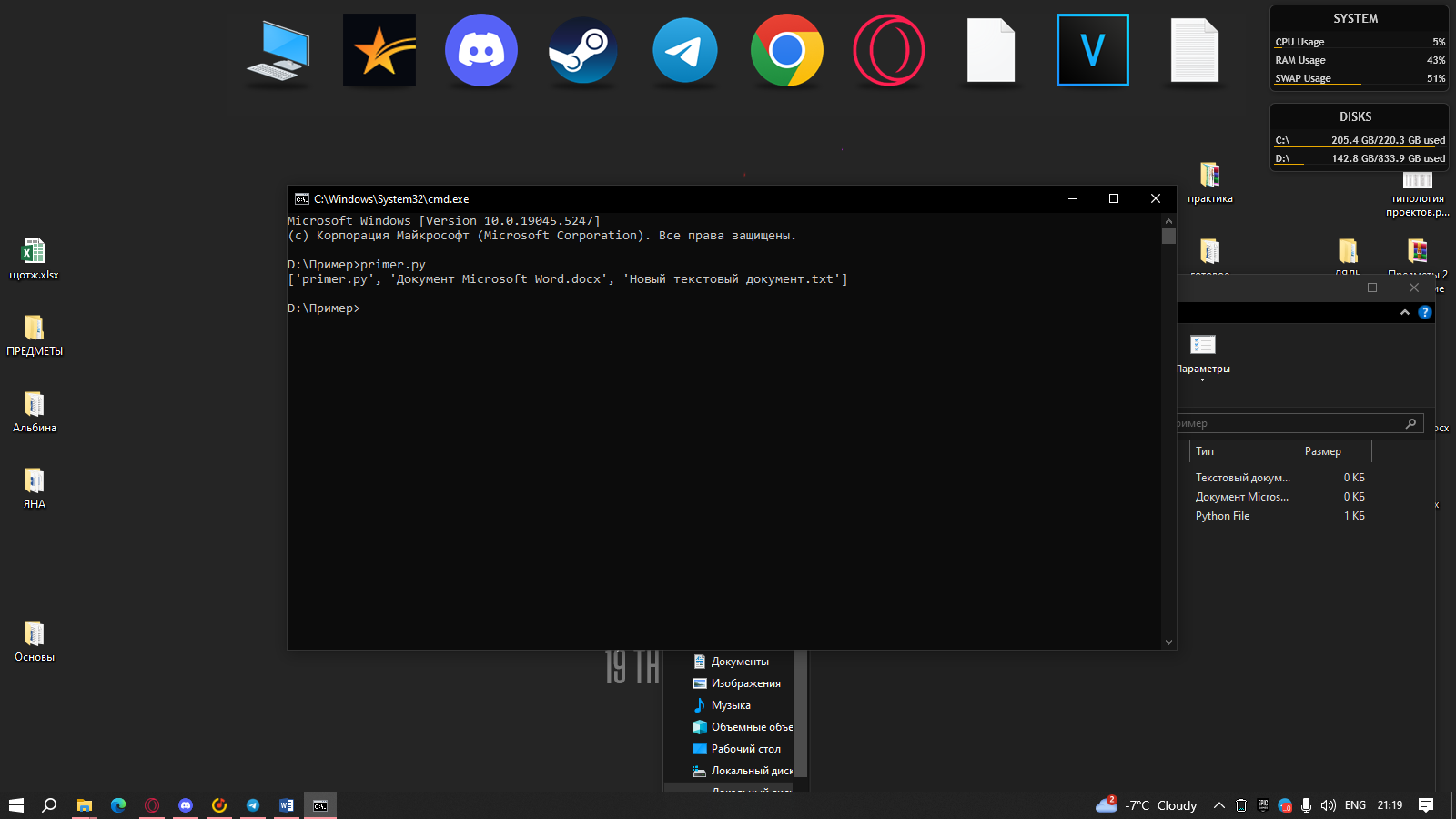
**files.append(entry)**

**return files**

**directory\_path = r'D:\Пример'**

**files = list\_files\_in\_directory(directory\_path)**

**print(files)**



Задание 4

Описание: Напишите функцию ensure\_directory\_exists(directory\_path), которая проверяет, существует ли указанный каталог. Если каталог не существует, функция должна создать его

**import os**

**def ensure\_directory\_exists(directory\_path):**

**if not os.path.exists(directory\_path):**

**os.makedirs(directory\_path)**

**print(f"Каталог '{directory\_path}' был успешно создан.")**

**else:**

**print(f"Каталог '{directory\_path}' уже существует.")**

**directory\_path = r'D:\Пример2\Пример' # Путь к каталогу**

**ensure\_directory\_exists(directory\_path)**

