Федеральное государственное автономное образовательное учреждение высшего образования

«ОМСКИЙ ГОСУДАРСТВЕННЫЙ

ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра Автоматизированные системы обработки информации и управления.

**Лабораторная работа №2**

по дисциплине «Основы профессиональной деятельности» на тему:

**«Разработка телеграм-бота.»**

Выполнил: студент группы ИВТ-244 Шмидт Антон Владиславович

Проверил: ассистент М.А. Семибратов

Омск 2025

**Введение**

Парсер — это программа для сбора и систематизации информации, размещенной на различных сайтах. Источником данных может служить текстовое наполнение, HTML-код сайта, заголовки, пункты меню, базы данных и другие элементы. Процесс сбора информации называется парсинг (parsing).

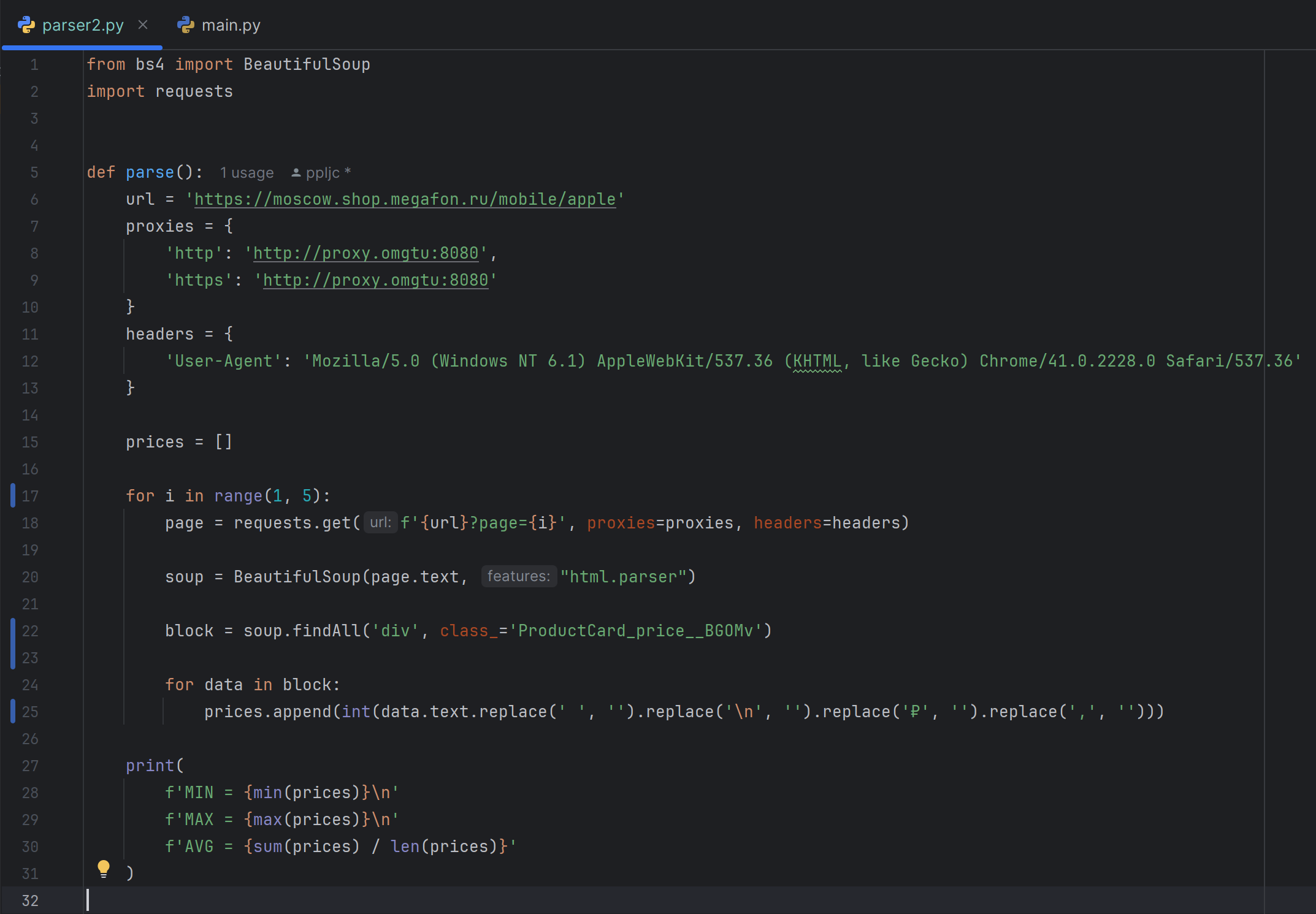
**Задача**

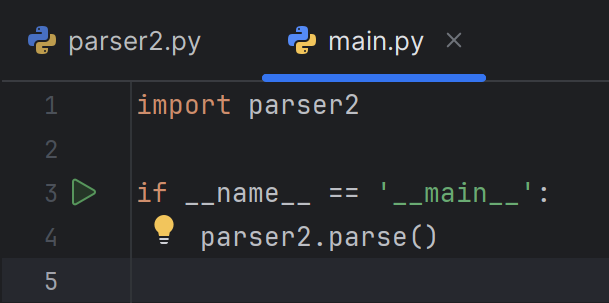
Распарсить сайт Эльдорадо, вкладку с iPhone, получить список цен всех айфонов, вывести из него min, max и среднее.

**Ход работы**

1. Создаём новый проект в PyCharm, инициализируем в нём Git и привязываем новый GitHub репозиторий.
2. Находим в браузере сайт Эльдорадо, открываем вкладку с iPhone и копируем URL, который вставляем в наш код.
3. В браузере открываем средства разработчика для данного сайта и в коде сайта находим элементы, которые отвечают за отображение позиций.
4. В данных элементах находим HTML-тег, отвечающий за цену, переносим его и его класс в код.
5. Проверяем наш код обычным выводом всех цен в консоль и понимаем, что ничего не выводится. Это происходит из-за того, что все позиции подгружаются динамически при загрузке сайта, а не сразу формируются в коде страницы.
6. Чтобы не усложнять парсер, меняем сайт Эльдорадо на Мегафон, на котором все позиции сразу формируются в HTML-коде.
7. Повторяем шаги 2-4, а при выполнении 5 шага видим, что в консоли появился список цен.
8. Убираем вывод в консоль и делаем добавление каждой цены в массив.
9. Добавляем функции для подсчёта минимального, максимального и среднего значения в массиве и выводим данные значения.
10. Сравниваем с значениями на сайте и видим, что цифры не совпадают, т.к. мы парсили только первую страницу, а их на сайте больше.
11. Меняем код парсера и делаем парсер в цикле с подстановкой в URL номера страницы.
12. Проверяем снова и видим, что результаты совпадают.
13. Загружаем проект на GitHub.

**Код**



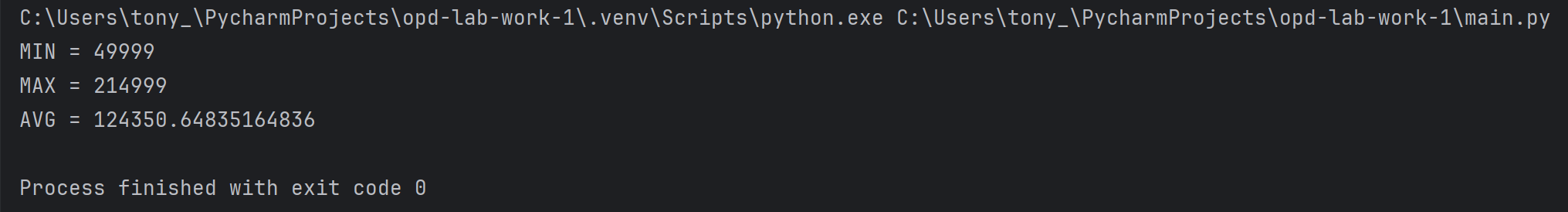


**Нотация к коду**

Для файла parser2.py. Импортируем библиотеки для парсинга; создаём функцию parser; определяем переменные url (сайта), proxies (для работы во внутренней сети ОмГТУ), headers (заголовки, чтобы сайт не думал, что мы - бот); определяем пустой список prices; в цикле на 4 итераций (количество страниц на сайте) получаем страницу с iPhone; загружаем её в BeautifulSoup с использованием стандартного парсера html.parser; находим все элементы с указанным тегом и классом (цены телефонов); затем в цикле проходимся по всем найденным элементам, удаляем пробелы, переносы на новую строку, запятые и знаки рубля и, преобразовав в целочисленный формат, добавляем в наш список prices; затем в самом конце подсчитываем и выводим минимальной, максимальное и среднее.

В файл main.py импортируем нашу функцию парсера и вызываем её (при запуске файла).

**Результат работы**



**Заключение**

В результате проделанной работы был написан парсер, который собрал данные о ценах всех айфонов на сайте магазина Мегафон и подсчитал минимальное, максимальное и среднее значение.

Ссылка на репозиторий: [ppljc/opd-lab-work-1](https://github.com/ppljc/opd-lab-work-1).