

Федеральное государственное бюджетное образовательное учреждение высшего образования  
«Сибирский государственный университет телекоммуникаций и информатики»  
(СибГУТИ)

09.03.01 "Информатика и вычислительная техника"  
профиль "Программное обеспечение средств  
вычислительной техники и автоматизированных систем"

## ОТЧЕТ

по учебной практике  
на кафедре Прикладной Математики и Кибернетики

Выполнил:

студент гр. ИП-016

\_\_\_\_\_/Мосолов П.А./

«11» мая 2022 г.

Руководитель практики

доцент каф. ПМиК

\_\_\_\_\_/Приставка П.А./

«\_\_» \_\_\_\_\_ г.

Оценка \_\_\_\_\_

Новосибирск 2022г.

## Содержание

|   |    |
|---|----|
| 1. Постановка задачи .....                | 3  |
| 2. Описание используемых алгоритмов ..... | 2  |
| 3. Листинг программы .....                | 6  |
| 4. Результаты тестирования .....          | 12 |
| 5. Список использованных источников ..... | 15 |

## 1. Постановка задачи

Разработать программу реализующую ввод, хранение и обработку данных о котировках криптовалют на основе данных сайта [coinmarketcap.com](https://coinmarketcap.com).

Общие требования к программе:

1. Язык разработки: Python версии не ниже 3.x
2. Операционная система: определяются студентом
3. Набор свойств криптовалют:
  - Name – наименование
  - Market\_cap – рыночная капитализация
  - Price – стоимость 1 ед. в долларах США (USD)
4. Ввод данных

| Оценки «хорошо» и «удовлетворительно»  | Оценка «отлично»   |
|--|--|
| Из файла <code>currencies22.csv</code> .<br>Файл содержит данные о 25 наиболее ценных криптовалютах на 05.03.2022 в формате<br><br>Name; Market_cap; Price<br><br>Файл доступен для скачивания в ЭИОС в директории с заданием на практику. | Непосредственно с главной страницы сайта <a href="https://coinmarketcap.com">coinmarketcap.com</a> в момент запуска программы.<br>Загрузка и парсинг веб-страницы производится с помощью библиотек <code>Requests</code> и <code>Beautifulsoup</code> или их аналогов<br><br>Примечание: допускается считывание строчек в количестве менее 25 (Например, 10 строчек с данными о криптовалютах) |

## 5. Хранение

Типы и структуры для хранения данных: определяются студентом

## 6. Обработка

Реализовать функцию поиска информации о свойствах криптовалюты по ее названию.

## 2. Описание используемых алгоритмов

**def merge\_information():** данная функция объединяет три массива в “структуру”, для дальнейшего использования в выводе графики.

**def mass\_clear():** функция очистки массивов после использования.

**def Pulling\_information\_site():** данная функция запускает считывание данных с сайта. Внутри функции стоит проверка, возможно ли подключиться к нему и запустить парсинг. Если нет, то вызываются исключения. Если все хорошо, функция стягивает разметку сайта, и запускает процедуру поиска интересующих нас данных.

**def find\_name\_crypto(crypto\_information\_list):** функция стягивания названий крипто валют с сайта. Алгоритм ищет по заданным параметрам, названия и записывает их в массив “names\_crypto”. Принимает “crypto\_information\_list” – разметка страницы для осуществления поиска.

**def find\_price\_crypto(crypto\_information\_list):** функция стягивания цены крипто валют с сайта. Алгоритм ищет по заданным параметрам, цену и записывает ее в массив “price\_crypto”. Принимает “crypto\_information\_list” – разметка страницы для осуществления поиска.

**def find\_market\_cap\_crypto(crypto\_information\_list):** функция стягивания рыночной капитализации крипто валют с сайта. Алгоритм ищет по заданным параметрам, рыночную капитализацию и записывает ее в массив “market\_cap\_crypto\_long”. Принимает “crypto\_information\_list” – разметка страницы для осуществления поиска.

**def terminal\_table\_output(length):** функция вывода данных в табличном виде, использует библиотеку “PrettyTable”.

**def searching\_element\_array():** функция поиска по заданному ключу крипто валюты. Выводит ошибку если по заданному ключу ничего не было найдено, если нашел совпадение, создает таблицу и выводит ее на экран (использует библиотеку “PrettyTable”)

**def parsing\_csv():** функция считывания данных из файла с форматом “csv”. Делает проверку, возможно ли открыть такой файл, если нет, выводит ошибку и предлагает пользователю решение проблемы. Если файл найден, то запускает чтение из файла и запись данных по массивам.

**def create\_data():** функция вывода данных о крипто валюте в виде таблицы.

**def gui\_pars():** функция запуска графического интерфейса с использованием библиотеки “BeautifulSoup”. Реализуется простой и

понятный для пользователя интерфейс, в котором есть две функции, вывод крипто валют с сайта и из файла.

**def UI():** самая главная функция в данном практическом задании. Запускает интерфейс в терминале и предлагает пользователю выбрать в каком режиме работать, либо в терминале, либо в графическом режиме. Режим терминала делится на две составляющие. Первая – это вывод данных из файла, вторая – вывод данных с сайта. Везде реализован выход из данного режима.

### 3. Листинг программы

```
import requests
from bs4 import BeautifulSoup
from prettytable import PrettyTable
import PySimpleGUI as sg
import os
import csv

FILE_NAME = "currencies22.csv"
SITE_URL = "https://coinmarketcap.com"

names_crypto = []
market_cap_crypto_long = []
price_crypto = []
struct_crypto = []

def merge_information():
    for i in range(len(names_crypto)):
        struct = {
            "name": names_crypto[i],
            "market_cap": market_cap_crypto_long[i],
            "price": price_crypto[i]
        }
        struct_crypto.append(struct)

def mass_clear():
    names_crypto.clear()
    price_crypto.clear()
    market_cap_crypto_long.clear()
    struct_crypto.clear()

def find_name_crypto(crypto_information_list):
```

```

    crypto_information_list_statistics =
crypto_information_list.find_all('p', class_='sc-1eb5slv-0 iworPT')

    for crypto_information in crypto_information_list_statistics:
        names_crypto.append(str(crypto_information.contents[0].text))


def find_market_cap_crypto(crypto_information_list):

    crypto_information_list_statistics =
crypto_information_list.find_all('span', class_='sc-1ow4cwt-1 ieFnWP')

    for crypto_information in crypto_information_list_statistics:
        market_cap_crypto_long.append(crypto_information.contents[0].text)


def find_price_crypto(crypto_information_list):

    crypto_information_list_statistics =
crypto_information_list.find_all('div', class_='sc-131di3y-0 cLgOOr')

    for crypto_information in crypto_information_list_statistics:
        price_crypto.append(crypto_information.contents[0].text)


def terminal_table_output(length):

    Table_crypto = PrettyTable()

    Table_crypto.field_names = ["№", "Name", "Price", "Marcet Cap"]

    for i in range(length):

        Table_crypto.add_row([i+1, names_crypto[i], price_crypto[i],
market_cap_crypto_long[i]])

    print(Table_crypto)


def searching_element_array():

    Table_crypto = PrettyTable()

    Table_crypto.field_names = ["№", "Name", "Price", "Marcet Cap"]

    myLetter = str(input("Enter the search key: "))

    i =0

    for word in names_crypto:

        if myLetter.lower() in word.lower():

            i = names_crypto.index(word)

```

```

        Table_crypto.add_row([i+1,names_crypto[i], price_crypto[i],
market_cap_crypto_long[i]])

    if i == 0:
        print("Error, invalid key, try again\n")

    else:
        print(Table_crypto)


def Pulling_information_site():
    try:
        page = requests.get(SITE_URL)
        soup = BeautifulSoup(page.text, 'html.parser')
        crypto_information_list = soup.find('tbody')

        find_name_crypto(crypto_information_list)
        find_market_cap_crypto(crypto_information_list)
        find_price_crypto(crypto_information_list)
    except (ConnectionError, Timeout) as exp:
        print("Mistake! The reason for the error: ", exp)
        sys.exit(1)


def parsing_csv():
    try:
        with open(FILE_NAME, encoding='utf-8') as file:
            file_reader = csv.reader(file, delimiter = ";")
            count = 0
            for row in file_reader:
                names_crypto.append(f'{row[0]}')
                price_crypto.append(f'{row[1]}')
                market_cap_crypto_long.append(f'{row[2]}')
    except FileNotFoundError:
        print("Mistake! The file '{}' could not be opened. Check the
correctness of the name, or the presence of it in the
pope.".format(FILE_NAME))
        sys.exit(1)

```



```

def create_data():
    new_struct_crypto = []

    for i in struct_crypto:
        new_struct_crypto.append([i["name"], i["market_cap"], i["price"]])

    return new_struct_crypto

def gui_pars():
    sg.theme("DarkPurple")

    head = ["Name of the cryptocurrency", " Marcet Cap ", "Cost in dollars"]

    frame_1 = [
        [sg.Text("Select the data output mode: ")],
        [sg.Button("1. FILE '{}'.format(FILE_NAME) )],
        [sg.Button("2. WEBSITE '{}'.format(SITE_URL) )],
        [sg.Text("Information: ")],
        [sg.Table(values=struct_crypto, headings=head, key="TABLE",
size=(500, 420))]]

    layout = [[sg.Column(frame_1)]]

    sg.set_options(font=("Courier New", 10))

    window = sg.Window("Parser", layout, size=(690, 620))

    gui = []
    while True:
        event, values = window.read()
        if event == sg.WIN_CLOSED:
            break

```

```

        if event == "1. FILE 'currencies22.csv'.":
            mass_clear()

            parsing_csv()
            merge_information()

            gui = create_data()
        if event == "2. WEBSITE 'https://coinmarketcap.com'.":
            mass_clear()

            Pulling_information_site()
            merge_information()

            gui = create_data()
            print(struct_crypto)
            window["TABLE"].update(values=gui)

    window.close()

def UI():
    os.system('cls')
    flag = 1
    flag_two = 1
    while flag == 1:
        mass_clear()
        mode = input("Select the output method ('T' from terminal, 'G' using
graphics, 'E' - end): ")
        if mode == 'T':
            flag_two = 1
            mass_clear()
            while flag_two == 1:
                mode2 = input("Choose the parsing method ('F' from file, 'S'
from the site, 'Q' - end): ")
                if mode2 == 'S':
                    mass_clear()

```

```

        os.system('cls')
        Pulling_information_site()
        terminal_table_output(len(names_crypto))
        searching_element_array()
    elif mode2 == 'F':
        mass_clear()
        os.system('cls')
        parsing_csv()
        terminal_table_output(len(names_crypto))
        searching_element_array()
    elif mode2 == 'Q':
        flag_two = 2
    else:
        os.system('cls')
        print("Invalid argument")

elif mode == 'G':
    mass_clear()
    os.system('cls')
    gui_pars()
elif mode == 'E':
    flag = 2
else:
    os.system('cls')
    print("Invalid argument")
os.system('cls')

def main():
    UI()

if __name__ == '__main__':
    main()

```

## 4. Результаты тестирования

```
Select the output method ('T' from terminal, 'G' using graphics, 'E' - end):
```

Рисунок 1. Главное меню в консольном режиме

```
Select the output method ('T' from terminal, 'G' using graphics, 'E' - end): T
Choose the parsing method ('F' from file, 'S' from the site, 'Q' - end):
```

Рисунок 2. Выбираем метод показа данных

| Nº | Name            | Price        | Marcet Cap        |
|----|-----------------|--------------|-------------------|
| 1  | Bitcoin         | \$39,042.91  | \$740,185,900,096 |
| 2  | Ethereum        | \$2,629.03   | \$314,699,571,974 |
| 3  | Tether          | \$1.00       | \$79,727,654,799  |
| 4  | BNB             | \$375.66     | \$61,923,153,319  |
| 5  | USD Coin        | \$1.00       | \$52,874,805,350  |
| 6  | XRP             | \$0.7222     | \$34,571,861,293  |
| 7  | Terra           | \$83.70      | \$30,924,825,975  |
| 8  | Cardano         | \$0.8485     | \$28,572,471,711  |
| 9  | Solana          | \$88.44      | \$27,996,936,059  |
| 10 | Avalanche       | \$76.12      | \$20,206,218,196  |
| 11 | Binance USD     | \$0.9996     | \$17,945,162,902  |
| 12 | Polkadot        | \$16.74      | \$16,527,777,456  |
| 13 | Dogecoin        | \$0.1229     | \$16,305,497,774  |
| 14 | TerraUSD        | \$1.00       | \$13,551,925,953  |
| 15 | Shiba Inu       | \$0.00002401 | \$13,184,282,172  |
| 16 | Polygon         | \$1.48       | \$11,259,979,784  |
| 17 | Wrapped Bitcoin | \$39,020.00  | \$10,401,835,175  |
| 18 | Cronos          | \$0.4018     | \$10,149,874,833  |
| 19 | Dai             | \$0.9999     | \$9,732,171,809   |
| 20 | Cosmos          | \$30.20      | \$8,642,212,818   |
| 21 | Litecoin        | \$101.93     | \$7,111,312,915   |
| 22 | NEAR Protocol   | \$10.62      | \$6,842,198,633   |
| 23 | Chainlink       | \$13.86      | \$6,457,496,696   |
| 24 | Uniswap         | \$8.92       | \$6,128,650,374   |
| 25 | TRON            | \$0.05834    | \$5,928,496,012   |

Enter the search key: B

| Nº | Name            | Price        | Marcet Cap        |
|----|-----------------|--------------|-------------------|
| 1  | Bitcoin         | \$39,042.91  | \$740,185,900,096 |
| 4  | BNB             | \$375.66     | \$61,923,153,319  |
| 11 | Binance USD     | \$0.9996     | \$17,945,162,902  |
| 15 | Shiba Inu       | \$0.00002401 | \$13,184,282,172  |
| 17 | Wrapped Bitcoin | \$39,020.00  | \$10,401,835,175  |

Choose the parsing method ('F' from file, 'S' from the site, 'Q' - end):

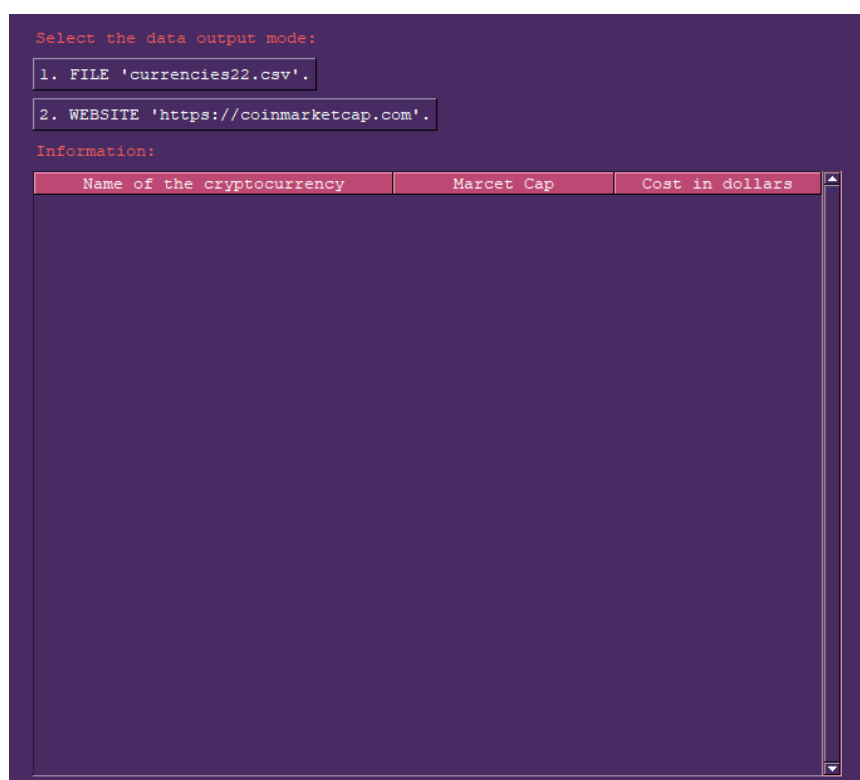
Рисунок 3. Вывод из файла и поиск в файле по ключу

```

+-----+-----+-----+-----+
| № | Name | Price | Marcet Cap |
+-----+-----+-----+-----+
| 1 | Bitcoin | $31,460.93 | $598,913,614,274 |
| 2 | Ethereum | $2,378.71 | $287,181,857,720 |
| 3 | Tether | $0.9998 | $83,128,866,155 |
| 4 | BNB | $315.11 | $51,450,862,068 |
| 5 | USD Coin | $1.00 | $48,371,314,070 |
| 6 | XRP | $0.5172 | $25,002,168,781 |
| 7 | Solana | $66.02 | $22,236,537,699 |
| 8 | Cardano | $0.6278 | $21,180,978,831 |
| 9 | Binance USD | $1 | $17,127,918,257 |
| 10 | Dogecoin | $0.1082 | $14,357,289,365 |
+-----+-----+-----+-----+
Enter the search key: e
+-----+-----+-----+-----+
| № | Name | Price | Marcet Cap |
+-----+-----+-----+-----+
| 2 | Ethereum | $2,378.71 | $287,181,857,720 |
| 3 | Tether | $0.9998 | $83,128,866,155 |
| 9 | Binance USD | $1 | $17,127,918,257 |
| 10 | Dogecoin | $0.1082 | $14,357,289,365 |
+-----+-----+-----+-----+
Choose the parsing method ('F' from file, 'S' from the site, 'Q' - end):

```

**Рисунок 4.** *Результат парсинга сайта и поиска по ключу*



**Рисунок 5.** *Графический режим работы программы*

Select the data output mode:

1. FILE 'currencies22.csv'.

2. WEBSITE 'https://coinmarketcap.com'.

Information:

| Name of the cryptocurrency | Marcet Cap        | Cost in dollars |
|----------------------------|-------------------|-----------------|
| Bitcoin                    | \$740,185,900,096 | \$39,042.91     |
| Ethereum                   | \$314,699,571,974 | \$2,629.03      |
| Tether                     | \$79,727,654,799  | \$1.00          |
| BNB                        | \$61,923,153,319  | \$375.66        |
| USD Coin                   | \$52,874,805,350  | \$1.00          |
| XRP                        | \$34,571,861,293  | \$0.7222        |
| Terra                      | \$30,924,825,975  | \$83.70         |
| Cardano                    | \$28,572,471,711  | \$0.8485        |
| Solana                     | \$27,996,936,059  | \$88.44         |
| Avalanche                  | \$20,206,218,196  | \$76.12         |
| Binance USD                | \$17,945,162,902  | \$0.9996        |
| Polkadot                   | \$16,527,777,456  | \$16.74         |
| Dogecoin                   | \$16,305,497,774  | \$0.1229        |
| TerraUSD                   | \$13,551,925,953  | \$1.00          |
| Shiba Inu                  | \$13,184,282,172  | \$0.00002401    |
| Polygon                    | \$11,259,979,784  | \$1.48          |
| Wrapped Bitcoin            | \$10,401,835,175  | \$39,020.00     |
| Cronos                     | \$10,149,874,833  | \$0.4018        |
| Dai                        | \$9,732,171,809   | \$0.9999        |
| Cosmos                     | \$8,642,212,818   | \$30.20         |
| Litecoin                   | \$7,111,312,915   | \$101.93        |
| NEAR Protocol              | \$6,842,198,633   | \$10.62         |
| Chainlink                  | \$6,457,496,696   | \$13.86         |
| Uniswap                    | \$6,128,650,374   | \$8.92          |
| TRON                       | \$5,928,496,012   | \$0.05834       |

**Рисунок 6. Результат вывода таблицы**

Select the data output mode:

1. FILE 'currencies22.csv'.

2. WEBSITE 'https://coinmarketcap.com'.

Information:

| Name of the cryptocurrency | Marcet Cap        | Cost in dollars |
|----------------------------|-------------------|-----------------|
| Bitcoin                    | \$596,736,343,666 | \$31,346.56     |
| Ethereum                   | \$285,775,163,833 | \$2,367.03      |
| Tether                     | \$83,128,958,474  | \$0.9998        |
| BNB                        | \$51,149,618,456  | \$313.27        |
| USD Coin                   | \$48,353,026,220  | \$0.9997        |
| XRP                        | \$24,864,369,152  | \$0.5143        |
| Solana                     | \$22,118,509,056  | \$65.67         |
| Cardano                    | \$21,054,049,084  | \$0.6238        |
| Binance USD                | \$17,127,814,127  | \$1             |
| Dogecoin                   | \$14,273,102,372  | \$0.1076        |

**Рисунок 7. Результат вывода с сайта**

## **5. Список использованных источников**

1. Лутц М. Изучаем Python, том 1, 5-е издание / М. Лутц. – СПб : Диалектика, 2019. – 832 с. – ISBN 978-5-907144-52-1
2. CSV File Reading and Writing : сайт. – URL: <https://docs.python.org/3/library/csv.html> (дата обращения: 20.04.2022)
3. Guide to Parsing HTML with BeautifulSoup in Python : сайт. – URL: <https://stackabuse.com/guide-to-parsing-html-with-beautifulsoup-in-python/> (дата обращения: 24.04.2022)
4. CoinMarketCap API Documentation : сайт. – URL: <https://coinmarketcap.com/api/documentation/v1/> (дата обращения: 30.04.2022)
5. Python GUIs for Humans : сайт. – URL: <https://pysimplegui.readthedocs.io/en/latest/> (дата обращения: 30.04.2022)