

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-012

\_\_\_\_\_/\_\_\_\_\_/

ФИО студента

«15» мая 2022г.

Руководитель практики

доцент каф. ПМиК

\_\_\_\_\_/Приставка П.А./

«15» мая 2022г.

Оценка \_\_\_\_\_

Новосибирск 2022 г.

# Оглавление

Условие задачи .....	3
Описание алгоритмов .....	3
Листинг программы .....	5
Результаты тестирования .....	8
Список использованных источников .....	10

## Условие задачи

Разработать программу, реализующую ввод, хранение и обработку данных о не более 25 криптовалютах на основе данных сайта [coinmarketcap.com](https://coinmarketcap.com) в браузере Chrome. Реализовать программу на Python3.10, на операционной системе Windows10, хранящую следующие свойства криптовалют:

- Name – наименование
- Market\_cap – рыночная капитализация
- Price – стоимость 1 ед. в долларах США (USD)

Реализовать ввод данных с сайта в момент запуска программы с помощью библиотек BeautifulSoup и сохранение полученных данных в CSV-таблице.

Реализовать поиск информации о свойствах криптовалюты по её названию.

## Описание алгоритмов

Для реализации автоматического сбора данных с сайта и их структурирования в функции `CryptoParsing()`, которая вызывается в меню в пункте 1), я воспользовался библиотеками Selenium, Webdriver\_Manager и BeautifulSoup. В функции `CryptoParsing(url)` сначала происходит установка нужной версии драйвера и запуск браузера Chrome, после чего браузеру передаётся URL-адрес, по которому открывается сайт [coinmarketcap.com](https://coinmarketcap.com).

Так как страница сайта является динамической – страница сайта формируется по мере просмотра, то необходимо сначала её всю прокрутить, после чего собирать данные о первых криптовалютах. Для этого вызывается скрипт, который возвращает высоту страницы, и вплоть до этого значения происходит постепенный скроллинг страницы с перерывами в 1 одну секунду для подгрузки страницы. Реализована задержка с помощью метода `time.sleep(1)` библиотеки `time`.

После скроллинга страницы в переменную `html` сохраняется её html-код и дальнейшей обработки кода используются BeautifulSoup() с `html.parser`. Исследовав код страницы сайта было обнаружено, что данные внутри тегов 'tr', принадлежавших div с классом 'bFzXgL'.

Поэтому с помощью команды `soup.find(class_='bFzXgL').find_all('tr')` я ищу все вхождения тегов 'tr' в этот div b сохраняю их в отдельный список.

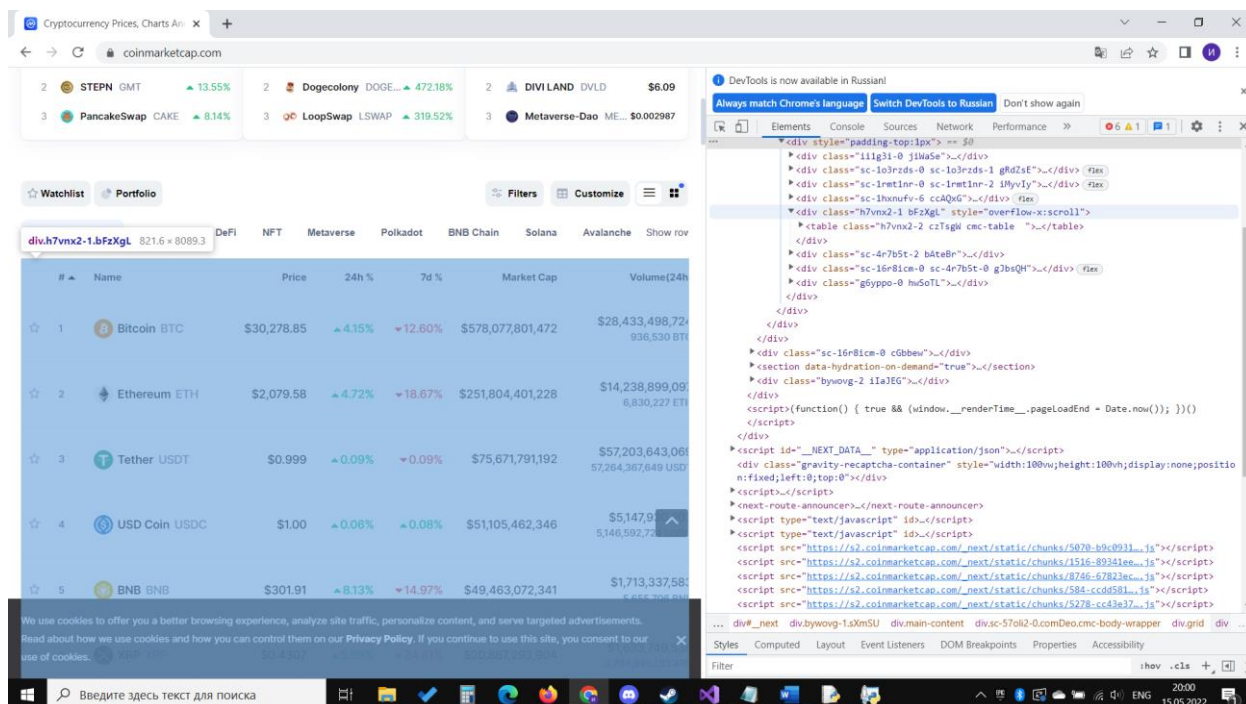


Рис.1. HTML – код страницы сайта coinmarketcap.com

Внутри этого списка поэлементно происходит сбор данных по классам, которые, согласно html-коду, хранят в себе информацию о Name, Price и Market\_cap криптовалют. Так как на одной странице находится 100 криптовалют, а по условию задания их необходимо считать не больше 25, необходимости переключаться между страницами не было, и при достижении этого значения парсинг сайта останавливается. Полученная информация хранится в списке списков, который возвращается как результат выполнения функции.

При необходимости можно посмотреть данные о первых 25 считанных криптовалют, вызвав функцию `PrintCrypto(data)` из меню под пунктом 2). Вывод в виде таблице реализован с помощью библиотеки Pandas, так как с её помощью можно полученный список списков можно преобразовать в DataFrame, который легко можно вывести с помощью функции `print()`.

Также в программе была прописана функция `SearchCrypto(data, key)` в меню под пунктом 3), в которой был реализован поиск криптовалют по первым буквам их наименований без учёта регистра и с последующим их выводом в виде таблицы.

Для сохранения полученных данных в виде CSV-таблицы была подключена библиотека Pandas, которая, после преобразования полученного

списка списков о криптовалютах в `DataFrame`, легко сохраняет его в виде CSV-таблицы. Данный функционал реализован в функции `CreateCSV(data)` в меню под пунктом 4).

В этой функции было дополнительно реализовано сохранение данных в файл с названием, соответствующим текущей дате. Для этого была подключена библиотека `datetime`, с помощью которой производится расчёт системного времени.

Вместо того, чтобы парсить сайт в условиях, если не будет доступа к сети Интернет, была реализована функция `LoadCSV()` в меню под пунктом 5). Эта функция преобразует данные из CSV-таблицы в `DataFrame` и возвращает с помощью функции `dataframe.values.tolist().copy()` список списков, содержащий информацию о всех криптовалютах в этой таблице. В результате можно также, как после парсинга сайта, просмотреть таблицу и искать криптовалюты по первым буквам наименования.

Дополнительно, подключив библиотеку `colorama`, был изменён цвет консольного шрифта на зелёный, а также добавлена возможность очищать терминал с помощью Escape-последовательности перед вызовом каждой функции из меню, а также во время парсинга сайта.

Для выхода из программы есть 0) пункт с названием "Exit".

## Листинг программы

```
import time
import colorama
import pandas as pd
from datetime import datetime
from selenium import webdriver
from webdriver_manager.chrome import ChromeDriverManager
from colorama import Fore, Back, Style
from bs4 import BeautifulSoup

colorama.init()
URL = "https://coinmarketcap.com/"
FileName = str(datetime.now().date())+'.csv'
ColumnsName = ['name','price','marketcap']

def CryptoParsing():
    LimitData = 25
    Data=[]
    count=0
    driver = webdriver.Chrome(ChromeDriverManager().install())
    print('\033[2J')
    print('\033[H')
```

```

driver.get(URL)
print('\033[2J')
print('\033[H')
for heig_d in range(0, driver.execute_script("return document.body.scrollHeight"), 500):
    driver.execute_script(f"0>window.scrollTo(0, {heig_d});")
    print('\033[2J')
    print('\033[H')
    time.sleep(1)

html = driver.page_source

soup = BeautifulSoup(html, 'html.parser')
items = soup.find(class_="bFzXgL").find_all('tr')
for item in items:
    if (item.find(class_="iworPT") != None):
        item_name = item.find(class_='iworPT').text
        item_price = item.find(class_='cLg00r').text
        item_marketcap = item.find(class_='iosgXe').text
        Data.append([item_name,item_price,item_marketcap])
        count+=1
        if(count==LimitData):
            break
print('\033[2J')
print('\033[H')
return Data

def PrintCrypto(data):
    dataframe = pd.DataFrame(data,columns=ColumnsName)
    print(dataframe.head(25))

def SearchCrypto(data, key):
    items = []
    for item in data:
        if item[0].upper().startswith(key.upper()):
            items.append(item)
    return items

def CreateCSV(data):
    dataframe = pd.DataFrame(data,columns=ColumnsName)
    dataframe.to_csv(FileName, index=False, sep=';')

def LoadCSV():
    dataframe = pd.read_csv(FileName, sep=';')
    data=dataframe.values.tolist().copy()
    return data

def menu():
    CryptoCurrency = []
    while True:
        print("1. Parsing CryptoCurrency")
        print("2. Print CryptoCurrency")

```

```

print("3. Search Cryptocurrency")
print("4. Create CSV")
print("5. Load today's CSV")
print("0. Exit")
sw = input("Select: ")
print('\033[2J')
print('\033[H')

if sw == "1":
    try:
        Cryptocurrency=CryptoParsing()
        print("Parsing was completed")
    except Exception:
        Cryptocurrency.clear()
        print("Please try again")
elif sw == "2":
    PrintCrypto(Cryptocurrency)
elif sw == "3":
    key = input("Enter (name): ")
    Items=SearchCrypto(Cryptocurrency,key)
    if not Items:
        print("Cryptocurrency hasn't been found!!!")
    else:
        print()
        i = 0
        print("{0:>21} {1:>15} {2:>15}".format(Col-
umnsName[0],ColumnsName[1],ColumnsName[2]))
        for item in SearchCrypto(Cryptocurrency,key):
            print("{0}{1:>20} {2:>15} {3:>15}".for-
mat(i,item[0],item[1],item[2]))
            i += 1
elif sw == "4":
    CreateCSV(Cryptocurrency)
    print("File CSV was created")
elif sw == "5":
    try:
        Cryptocurrency=LoadCSV()
        print("Loading was completed")
    except Exception:
        Cryptocurrency.clear()
        print("File CSV not found")
elif sw == "0":
    break
print()
print('\033[2J')
print('\033[H')
print('\033[32m')
menu()

```

## Результаты тестирования

4	USD Coin	USDC	\$1.00	▼0.01%	▲0.02%	\$51,095,293,745	\$5,022,963,981 5,022,658,343 USDC	51,092,184,697 US
5	BNB	BNB	\$300.13	▲9.22%	▼15.73%	\$49,004,644,808	\$1,690,409,641 5,632,221 BNB	163,276,975 B
6	XRP	XRP	\$0.4269	▲5.89%	▼25.34%	\$20,638,168,410	\$1,606,726,873 3,763,616,919 XRP	48,343,101,197
7	Cardano	ADA	\$0.5515	▲11.18%	▼25.63%	\$18,650,225,590	\$1,026,824,704 1,862,040,806 ADA	33,820,262,544
8	Solana	SOL	\$52.69	▲11.54%	▼32.37%	\$17,773,824,281	\$1,759,509,451 33,393,645 SOL	337,328,554
9	Binance USD	BUSD	\$1.00	▼0.11%	▲0.04%	\$17,752,525,769	\$9,314,644,357 9,305,667,798 BUSD	17,735,417,590 B
10	Dogecoin	DOGE	\$0.08901	▲3.52%	▼29.07%	\$11,809,501,117	\$722,417,920 8,115,815,954 DOGE	132,670,764,3
11	Polkadot	DOT	\$10.94	▲8.08%	▼18.72%	\$10,803,163,239	\$1,155,165,397 105,605,216 DOT	987,579,315

We use cookies to offer you a better browsing experience, analyze site traffic, personalize content, and serve targeted advertisements. Read about how we use cookies and how you can control them on our [Privacy Policy](#). If you continue to use this site, you consent to our use of cookies.

Рис. 2. Парсинг сайта с помощью selenium

```
Командная строка - python CursWork.py

Parsing was completed

1. Parsing Cryptocurrency
2. Print Cryptocurrency
3. Search Cryptocurrency
4. Create CSV
5. Load today's CSV
6. Exit
Select:
```

Рис. 3. Удачное завершение парсинга

```
Командная строка - python CursWork.py

      name      price marketcap
0      Bitcoin  $30,156.03 $574.18B
1      Ethereum $2,067.81  $249.77B
2      Tether    $0.9989   $75.67B
3      USD Coin  $1.00     $51.10B
4      BNB       $298.60   $48.93B
5      XRP       $0.4263   $20.61B
6      Cardano   $0.5489   $18.65B
7      Solana    $52.61    $17.77B
8      Binance USD $1.00     $17.77B
9      Dogecoin  $0.08896  $11.80B
10     Polkadot  $10.80    $10.77B
11     Avalanche $34.09    $9.22B
12     Wrapped Bitcoin $30,132.56 $8.58B
13     TRON      $0.06911  $6.78B
14     Shiba Inu  $0.00001235 $6.78B
15     Dai        $0.9997   $6.42B
16     Polygon    $0.6795   $5.40B
17     Cronos     $0.1922   $4.86B
18     Litecoin   $68.17    $4.79B
19     UNUS SED LEO $4.98     $4.75B
20     NEAR Protocol $6.69     $4.61B
21     FTX Token  $31.67    $4.32B
22     Bitcoin Cash $207.02   $3.95B
23     Uniswap    $5.25     $3.62B
24     Chainlink  $7.46     $3.48B

1. Parsing Cryptocurrency
2. Print Cryptocurrency
```



**Рис. 4.** Список данных о 25 криптовалютах

```

Командная строка - python CursWork.py

Enter (name): bi

      name      price      marketcap
0      Bitcoin   $30,156.03   $574.18B
1      Binance USD   $1.00     $17.77B
2      Bitcoin Cash   $207.02    $3.95B

1. Parsing CryptoCurrency
2. Print CryptoCurrency
3. Search CryptoCurrency
4. Create CSV
5. Load today's CSV
0. Exit
Select:
  
```

**Рис. 5.** Результат поиска криптовалют по первым буквам в списке

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
1	name	price	marketcap																			
2	Bitcoin	\$30,156.03	\$574.18B																			
3	Ethereum	\$2,067.81	\$249.77B																			
4	Tether	\$0.9989	\$75.67B																			
5	USD Coin	\$1.00	\$51.10B																			
6	BNB	\$298.60	\$48.93B																			
7	XRP	\$0.4263	\$20.61B																			
8	Cardano	\$0.5489	\$18.65B																			
9	Solana	\$52.61	\$17.77B																			
10	Binance U	\$1.00	\$17.77B																			
11	Dogecoin	\$0.08896	\$11.80B																			
12	Polkadot	\$10.80	\$10.77B																			
13	Avalanche	\$34.09	\$9.22B																			
14	Wrapped f	\$30,132.51	\$8.58B																			
15	TRON	\$0.06911	\$6.78B																			
16	Shiba Inu	\$0.000012	\$6.78B																			
17	Dai	\$0.9997	\$6.42B																			
18	Polygon	\$0.6795	\$5.40B																			
19	Cronos	\$0.1922	\$4.86B																			
20	Litecoin	\$68.17	\$4.79B																			
21	UNUS SED	\$4.98	\$4.75B																			
22	NEAR Prot	\$6.69	\$4.61B																			
23	FTX Token	\$31.67	\$4.32B																			
24	Bitcoin Cai	\$207.02	\$3.95B																			
25	Uniswap	\$5.25	\$3.62B																			
26	Chainlink	\$7.46	\$3.48B																			
27																						
28																						

**Рис. 6.** CSV-таблица с результатами парсинга сайта

```

Командная строка - python CursWork.py

Loading was completed

1. Parsing CryptoCurrency
2. Print CryptoCurrency
3. Search CryptoCurrency
4. Create CSV
5. Load today's CSV
0. Exit
Select:
  
```

**Рис. 7.** Удачное завершение загрузки данных из CSV-таблицы

```
Командная строка - python CursWork.py

name price marketcap
0 Bitcoin $30,156.03 $574.18B
1 Ethereum $2,067.81 $249.77B
2 Tether $0.9989 $75.67B
3 USD Coin $1.00 $51.10B
4 BNB $298.60 $48.93B
5 XRP $0.4263 $20.61B
6 Cardano $0.5489 $18.65B
7 Solana $52.61 $17.77B
8 Binance USD $1.00 $17.77B
9 Dogecoin $0.08896 $11.80B
10 Polkadot $10.80 $10.77B
11 Avalanche $34.09 $9.22B
12 Wrapped Bitcoin $30,132.56 $8.58B
13 TRON $0.06911 $6.78B
14 Shiba Inu $0.00001235 $6.78B
15 Dai $0.9997 $6.42B
16 Polygon $0.6795 $5.40B
17 Cronos $0.1922 $4.86B
18 Litecoin $68.17 $4.79B
19 UNUS SED LEO $4.98 $4.75B
20 NEAR Protocol $6.69 $4.61B
21 FTX Token $31.67 $4.32B
22 Bitcoin Cash $207.02 $3.95B
23 Uniswap $5.25 $3.62B
24 Chainlink $7.46 $3.48B

1. Parsing Cryptocurrency
Print Cryptocurrency
```

Рис. 8. Список криптовалют, загруженных из CSV-таблицы

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1) Васильев А.Н. Программирование на Python [Электронный ресурс]  
URL: <https://www.litres.ru/aleksey-nikolaevich/programmirovanie-na-python-v-primerah-i-zada-63455717/> (дата обращения: 15.05.2022)
- 2) Чтение и запись CSV-файлов в Python с Pandas [Электронный ресурс]  
URL: <https://pythobyte.com/reading-and-writing-csv-files-in-python-with-pandas-4e74976e/>
- 3) Webdriver Manager for Python [Электронный ресурс] URL:  
<https://pypi.org/project/webdriver-manager/>
- 4) BeautifulSoup Documentation [Электронный ресурс] URL:  
<https://www.crummy.com/software/BeautifulSoup/bs4/doc/>
- 5) Цветной вывод текста в Python [Электронный ресурс] URL: <https://all-python.ru/osnovy/tsvetnoj-vyvod-teksta.html>