

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-014

Петровский Владислав Евгеньевич  
ФИО студента

«28» мая 2022г.

Руководитель практики

доцент каф. ПМиК

\_\_\_\_\_/Приставка П.А./

«28» мая 2022г.

Оценка \_\_\_\_\_

Новосибирск 2022 г.

## Условие 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)

## Описание алгоритмов

1. При помощи Selenium WebDriver'а производится парсинг элементов страницы, которые далее хранятся в массиве *coins*.
2. При помощи библиотеки pandas создаём *dataframe*, в который записываем полученный массив и далее сохраняем его в файл *output.csv*.
3. При помощи написанной функции *search\_coin(coin\_name: str)*, производим поиск в *dataframe*.

## Листинг программы

```
from selenium import webdriver
import pandas

class Coin: # Custom class
    name = ""
    short_name = ""
    market_cap = ""
    price = ""

def search_coin(coin_name: str): # Search function
    search_results = dataframe[dataframe.eq(coin_name).any(1)]
    if(not search_results.empty):
        return search_results
    else:
        return 'ERROR: NOT FOUND, try another one'

# Parsing
driver = webdriver.Chrome()
```

```

driver.get("https://coinmarketcap.com/")
driver.execute_script(f"window.scrollTo(0, 1250);")
coins = []
n = 25
tbody = driver.find_element_by_tag_name('tbody')
coins_rows = tbody.find_elements_by_tag_name('tr')[0:n]
for i in range(n):
    new_coin = Coin()
    new_coin.name = coins_rows[i].find_elements_by_tag_name('p')[1].text
    new_coin.short_name = coins_rows[i].find_elements_by_tag_name('p')[2].text
    new_coin.price = coins_rows[i].find_elements_by_tag_name('span')[2].text
    new_coin.market_cap = coins_rows[i].find_elements_by_tag_name('span')[
        8].text
    coins.append(new_coin)
driver.close()

# Making dataframe
dataframe_ = pandas.DataFrame([vars(coin) for coin in coins])
dataframe_.index = dataframe_.index + 1
dataframe_.to_csv('output.csv', sep=';')
print(dataframe_)

# Main cycle
while True:
    print('Input a name or acronym of the cryptocurrency to get info from dataframe
>>', end=" ")
    print(search_coin(str(input()))))

```

## Результаты тестирования

Вывод результата работы программы в консоль:

	name	short_name	price	market_cap
1	Bitcoin	BTC	\$44,704.42	\$849,004,257,189
2	Ethereum	ETH	\$3,149.02	\$378,287,052,739
3	Tether	USDT	\$1.00	\$81,364,498,800
4	BNB	BNB	\$417.41	\$68,921,808,253
5	USD Coin	USDC	\$0.9999	\$51,978,229,638
6	XRP	XRP	\$0.8374	\$40,297,731,902
7	Cardano	ADA	\$1.14	\$38,503,566,390
8	Solana	SOL	\$102.07	\$33,165,591,061
9	Terra	LUNA	\$90.94	\$32,448,214,632
10	Avalanche	AVAX	\$86.25	\$23,052,627,513
11	Polkadot	DOT	\$21.20	\$20,939,257,666
12	Dogecoin	DOGE	\$0.14	\$18,570,874,823
13	Binance USD	BUSD	\$0.9998	\$17,561,635,674
14	TerraUSD	UST	\$1.00	\$16,150,599,893
15	Shiba Inu	SHIB	\$0.00002467	\$13,543,922,496
16	Polygon	MATIC	\$1.62	\$12,495,687,396
17	Wrapped Bitcoin	WBTC	\$44,657.74	\$12,250,974,576
18	Cronos	CRO	\$0.4607	\$11,638,444,070
19	Dai	DAI	\$0.9998	\$9,667,966,434
20	Litecoin	LTC	\$125.99	\$8,810,873,289
21	NEAR Protocol	NEAR	\$12.96	\$8,566,796,661
22	Cosmos	ATOM	\$28.97	\$8,295,521,187
23	Chainlink	LINK	\$16.14	\$7,537,914,705
24	Uniswap	UNI	\$10.74	\$7,402,224,852
25	Bitcoin Cash	BCH	\$362.79	\$6,900,124,788

## Произведённый поиск с успешным результатом и выводом ошибки

```
Input a name or acronym of the cryptocurrency to get info from dataframe >> LTC
      name short_name  price  market_cap
20  Litecoin      LTC  $125.99  $8,810,873,289
Input a name or acronym of the cryptocurrency to get info from dataframe >> wrong_name
ERROR: NOT FOUND, try another one
```

## Содержимое файла output.csv, после работы программы

```
PRACTICE > output.csv
1 ;name;short_name;price;market_cap
2 1;Bitcoin;BTC;$44,704.42;$849,004,257,189
3 2;Ethereum;ETH;$3,149.02;$378,287,052,739
4 3;Tether;USDT;$1.00;$81,364,498,800
5 4;BNB;BNB;$417.41;$68,921,808,253
6 5;USD Coin;USDC;$0.9999;$51,978,229,638
7 6;XRP;XRP;$0.8374;$40,297,731,902
8 7;Cardano;ADA;$1.14;$38,503,566,390
9 8;Solana;SOL;$102.07;$33,165,591,061
10 9;Terra;LUNA;$90.94;$32,448,214,632
11 10;Avalanche;AVAX;$86.25;$23,052,627,513
12 11;Polkadot;DOT;$21.20;$20,939,257,666
13 12;Dogecoin;DOGE;$0.14;$18,570,874,823
14 13;Binance USD;BUSD;$0.9998;$17,561,635,674
15 14;TerraUSD;UST;$1.00;$16,150,599,893
16 15;Shiba Inu;SHIB;$0.00002467;$13,543,922,496
17 16;Polygon;MATIC;$1.62;$12,495,687,396
18 17;Wrapped Bitcoin;WBTC;$44,657.74;$12,250,974,576
19 18;Cronos;CRO;$0.4607;$11,638,444,070
20 19;Dai;DAI;$0.9998;$9,667,966,434
21 20;Litecoin;LTC;$125.99;$8,810,873,289
22 21;NEAR Protocol;NEAR;$12.96;$8,566,796,661
23 22;Cosmos;ATOM;$28.97;$8,295,521,187
24 23;Chainlink;LINK;$16.14;$7,537,914,705
25 24;Uniswap;UNI;$10.74;$7,402,224,852
26 25;Bitcoin Cash;BCH;$362.79;$6,900,124,788
27 |
```

## Приложение

Файлы проекта - [petrovskiy-ve/PRACTICE \(github.com\)](https://github.com/retrovskiy-ve/PRACTICE)

### Список использованных источников

1. Парсинг сайтов при помощи Python и Selenium –YouTube  
([https://www.youtube.com/playlist?list=PLNtbY2SkIByzpcsjmDP\\_3zW52NcDLTY1M](https://www.youtube.com/playlist?list=PLNtbY2SkIByzpcsjmDP_3zW52NcDLTY1M))
2. Введение в pandas: анализ данных на Python  
(<https://khashtamov.com/ru/pandas-introduction/>)