

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 группы: ИП-011

Мирошниченко Р.О.

«28» мая 2022г.

Руководитель практики

Доцент каф. ПМиК

\_\_\_\_\_/Приставка П.А./

«28» мая 2022г.

Оценка \_\_\_\_\_

Новосибирск 2022 г.

## Содержание

- Условие задачи
- Описание алгоритмов
- Листинг программы
- Результаты тестирования
- Список использованных источников

## **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:
  1. Name – наименование
  2. Market\_cap – рыночная капитализация
  3. Price – стоимость 1 ед. в долларах США (USD)

## **Описание алгоритмов**

С помощью библиотеки “BeautifulSoup” выполняется парсинг страницы – берётся полный код страницы, после чего вытаскиваются определенные теги.

С помощью библиотеки “pandas” создается DataFrame , с помощью которого данные после парсинга записываются в файл “ExcelParsing.csv”.

С помощью “requests” отправляется запрос GET на сайт, после чего html код преобразуется в текст и происходит парсинг.

URL – содержит саму ссылку на сайт.

User-agent – необходим для того, чтобы имитировался вход на сайт пользователем и чтобы запрос requests не был отклонён (взял из вкладки network).

def searchCurrency – функция поиска валюты по имени.

def currencies – сама функция парсинга, в которой происходит нахождение валюты, вывод валюты в Excel, а также вызов функции поиска.

## Листинг программы

Программа была написана в PyCharm 2021.3.2

```
from bs4 import BeautifulSoup
import requests
import pandas as pd

URL = "https://coinmarketcap.com/"

headers = {
    "User-agent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/101.0.4951.54 Safari/537.36"
}

def searchCurrency(coin: str, DF):
    results = DF[DF.eq(coin).any(1)]
    if(not results.empty):
        return results
    else:
        return 'Error: Not found'

def currencies():
    html = requests.get(URL, headers=headers).text
    sup = BeautifulSoup(html, "html.parser")
    body = sup.tbody
    currencies = body.find_all("tr")[0:10]
    currencyList = list()

    for currency in currencies:
        pTags = currency.find_all("p")
        spanTags = currency.find_all("span")
        currencyInfo = {'Name': pTags[1].text, 'price': spanTags[2].text,
'market_cap': spanTags[8].text}
        currencyList.append(currencyInfo)

    frame = pd.DataFrame(currencyList)
    print(frame)
    frame.to_csv('ExcelParsing.csv')

    while True:

        print("\n" 'Enter the title: ', end=" ")
        print(searchCurrency(str(input()), frame))

currencies()
```

```

1  from bs4 import BeautifulSoup
2  import requests
3  import pandas as pd
4
5  URL = "https://coinmarketcap.com/"
6
7  headers = {
8      "User-agent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/
9  }
10
11  def searchCurrency(coin: str, DF):
12      results = DF[DF.eq(coin).any(1)]
13      if(not results.empty):
14          return results
15      else:
16          return 'Error: Not found'
17
18  def currencies():
19      html = requests.get(URL, headers=headers).text
20      sup = BeautifulSoup(html, "html.parser")
21      body = sup.tbody
22      currencies = body.find_all("tr")[0:10]
23      currencyList = list()
24
25      for currency in currencies:
26          pTags = currency.find_all("p")
27          spanTags = currency.find_all("span")
28          currencyInfo = {'Name': pTags[1].text, 'price': spanTags[2].text, 'market_cap': spanTags[8].text}
29          currencyList.append(currencyInfo)
30
31      frame = pd.DataFrame(currencyList)
32      print(frame)
33      frame.to_csv('ExcelParsing.csv')
34
35      while True:
36
37          print("\n" 'Enter the title: ', end=" ")
38          print(searchCurrency(str(input()), frame))
39
40      currencies()

```

## Результат тестирования

При запуске выводятся первые 10 криптовалют с сайта

	Name	price	market_cap
0	Bitcoin	\$29,606.21	\$563,744,896,169
1	Ethereum	\$2,008.53	\$242,634,412,505
2	Tether	\$0.9989	\$75,672,364,752
3	USD Coin	\$1.00	\$51,165,199,429
4	BNB	\$295.43	\$48,236,731,018
5	XRP	\$0.416	\$20,109,095,264
6	Cardano	\$0.5578	\$18,818,919,732
7	Binance USD	\$0.9994	\$17,975,444,518
8	Solana	\$52.51	\$17,718,436,693
9	Dogecoin	\$0.087	\$11,542,374,983

Enter the title:

После чего мы вводим название какой-либо валюты и дальше происходит поиск

1	Ethereum	\$2,008.53	\$242,634,412,505
2	Tether	\$0.9989	\$75,672,364,752
3	USD Coin	\$1.00	\$51,165,199,429
4	BNB	\$295.43	\$48,236,731,018
5	XRP	\$0.416	\$20,109,095,264
6	Cardano	\$0.5578	\$18,818,919,732
7	Binance USD	\$0.9994	\$17,975,444,518
8	Solana	\$52.51	\$17,718,436,693
9	Dogecoin	\$0.087	\$11,542,374,983

Enter the title: **BNB**

	Name	price	market_cap
4	BNB	\$295.43	\$48,236,731,018

Помимо этого на фоне происходит запись в файл ExcelParsing.csv

	A
1	,Name,price,market_cap
2	0,Bitcoin,"\$29,606.21","\$563,744,896,169"
3	1,Ethereum,"\$2,008.53","\$242,634,412,505"
4	2,Tether,\$0.9989,"\$75,672,364,752"
5	3,USD Coin,\$1.00,"\$51,165,199,429"
6	4,BNB,\$295.43,"\$48,236,731,018"
7	5,XRP,\$0.416,"\$20,109,095,264"
8	6,Cardano,\$0.5578,"\$18,818,919,732"
9	7,Binance USD,\$0.9994,"\$17,975,444,518"
10	8,Solana,\$52.51,"\$17,718,436,693"
11	9,Dogecoin,\$0.087,"\$11,542,374,983"
12	
13	
14	
15	

## **Список использованных источников**

1. Документация Python. (<https://www.python.org/doc/>)
2. Документация pandas. (<https://pandas.pydata.org/pandas-docs/stable/>)