

UJIAN AKHIR SEMESTER  
PRAKTIK PEMROGRAMMAN  
PYTHON



**Disusun oleh :**  
Wahyu Bagus Dwi  
Prasetyo

V3922045

**Dosen**  
Yusuf Fadila Rachman, S. Kom., M. Kom

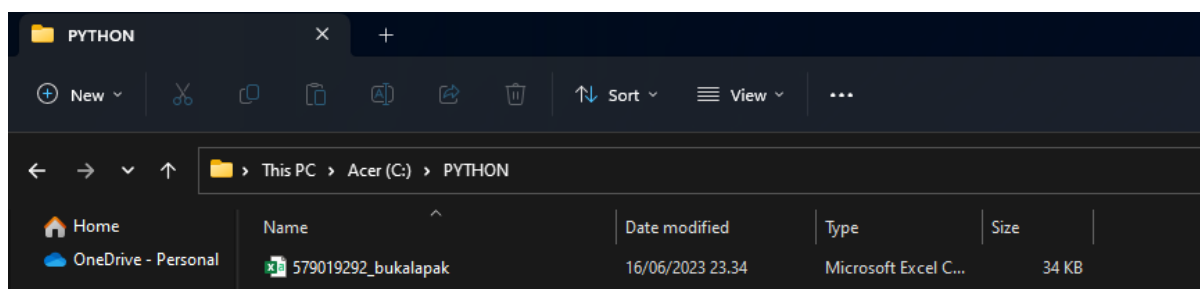
**PS D-III TEKNIK INFORMATIKA  
SEKOLAH VOKASI  
UNIVERSITAS SEBELAS MARET  
2023**

SOAL

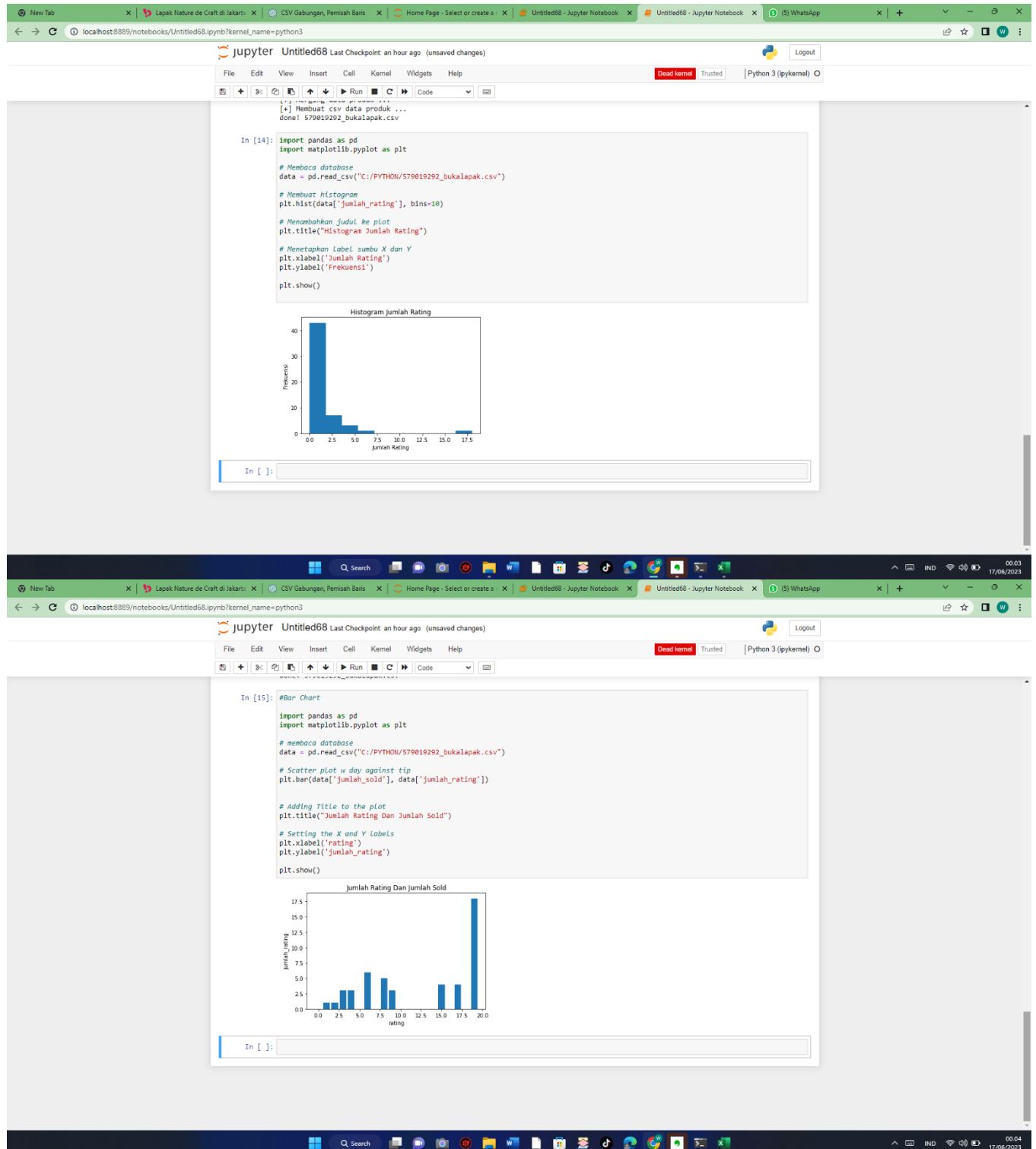
1. Ambil data (Scraping data) di salah satu dari e-commerce berikut (pilih salah satu):
  - a. **Ebay**
  - b. **Zalora**
  - c. **Blibli**
  - d. **Bukalapak**
  - e. **Bhinneka**
2. Deskripsikan data apa yang diambil dan di halaman mana data tersebut diambil sedetail mungkin !  
Misalkan akan mengambil data di Tokopedia pada kategori handphone dan tablet -> aksesoris HP->pada halaman 1 dan 2. Data yang diambil bebas asalkan jelas kronologis dan penjelasannya.
3. Data yang diambil minimal berjumlah 50 buah. Data yang disimpan adalah :
  - a. Wajib : Nama Produk, Harga
  - b. Pilihan : Kategori, Kota/Provinsi (pilih salah 1)
4. Simpan data yang diambil ke dalam bentuk CSV
5. Buatlah 2 grafik menggunakan matplotlib berdasarkan data hasil scraping tadi (pilih 2):
  - a. Line Chart
  - b. Bar Chart
  - c. Histogram
  - d. Pie Chart
  - e. Scatter
6. Jelaskan secara singkat kode program yang anda buat, tuliskan di readme Github atau langsung di file kode python anda dalam bentuk comment. Kemudian link github dikumpulkan ke Spada
7. Dokumen yang dikumpulkan ke github yaitu :
  - a. Kode program scraping python
  - b. File CSV berisi data
  - c. 2 Gambar grafik
  - d. Nama folder/direktori github : UAS Python NIM Nama
8. Deadline pengumpulan tugas UAS : 22 Juni 2023

Jawab:

1. Disini saya mengambil Scrapping Data di Website Blibli
2. Saya mengambil data di blibli pada toko: [https://www.bukalapak.com/u/nature\\_de\\_craft\\_46327](https://www.bukalapak.com/u/nature_de_craft_46327) dan untuk halamannya saya mengambil di halaman 1-4
3. Data yang saya ambil berjumlah 56 buah untuk data wajib seperti: Nama Produk, Harga sudah tersimpan. Lalu Untuk data Pilihan saya Memilih Kategori
4. Disini sudah saya simpan dan saya buat file baru (Untuk file csv nanti saya sertakan di github)



5. Disini saya mengambil barchart dan histogram untuk memvisualisasikan/menampilkan data yang sudah saya simpan sebagai contoh disini saya ingin menampilkan rating dan jumlah sold nya



6. Penjelasan Kode Disini saya cantumkan comment dan file readme untuk readme saya upload digithub:

**A. Berikut kode dengan menggunakan penjelasan comment pada program untuk menyimpan file csv**

```
#!/usr/bin/env python
# coding: utf-8

# In[ ]:

#!/usr/bin/env python
# coding: utf-8

# Import library yang diperlukan
from bs4 import BeautifulSoup as bs
import requests as req
import glob
import json
import os
import csv
import time

# URL dasar
baseurl = 'https://www.bukalapak.com/u/'
apiurl = 'https://api.bukalapak.com/stores/'

# Definisikan kelas Bukalapak
class Bukalapak:
    def __init__(self, username):
        self.username = username
        self.urltoko = baseurl + username
        self.headerbrowser = {
            'User-Agent': 'Mozilla/5.0 (X11; Linux x86_64; rv:74.0) Gecko/20100101 Firefox/74.0'
        }
        pg_data = req.get(self.urltoko, headers=self.headerbrowser, timeout=3000)
        try:
            if pg_data.status_code == 200:
                soup = bs(pg_data.text, 'html.parser')
                # Mengambil id penjual
                print("[+] Mengambil id penjual ...")
                self.idseller = soup.find('a', attrs={'data-user-id': True}).get('data-user-id')
                print("[+] Id penjual: " + self.idseller)
                # Mengambil access token untuk API
                print("[+] Mengambil access token ...")
                token = soup.find_all('script')
                gtoken = str(token[4]).replace("<script>localStorage.setItem('bl_token', '", "'").replace('");</script>',
                ""))
                gtoken = json.loads(str(gtoken))
                self.token = gtoken['access_token']
                print("[+] Access token: " + self.token)
                self.grab_produk()
            else:
                print("[!] Username tidak ditemukan!")
        except Exception as e:
            print(e)
            print("[!] Website down!")
            exit()
```

```

def grab_produk(self):
    print("=== GRABBING PRODUK ===")
    print("[+] Memulai download halaman produk ...")
    print("\x1B[3m" + "(delay 3 detik untuk menghindari anti-spam!)" + "\x1B[0m")
    api = apiurl + str(self.idseller) + '/products?offset=0&limit=50&access_token=' + self.token
    total_product = req.get(api, headers=self.headerbrowser, timeout=3000).json()
    print("[+] Total produk: " + str(total_product['meta']['total']))
    print("[+] Hapus file lama ...")
    if not os.path.exists("data"):
        os.makedirs("data")
    for filename in glob.glob("data/" + str(self.idseller) + "bukalapak*.json"):
        os.remove(filename)
    for filename in glob.glob(str(self.idseller) + "_bukalapak.csv"):
        os.remove(filename)
    # Mengambil halaman produk
    a = 0
    b = 0
    while True:
        print("-> download halaman ke-" + str(b + 1))
        api = apiurl + str(self.idseller) + '/products?offset=' + str(a) + '&limit=50&access_token=' + self.token
        pg_data = req.get(api, headers=self.headerbrowser, timeout=3000).json()
        if len(pg_data['data']) == 0:
            break
        with open("data/" + str(self.idseller) + "bukalapak" + str(b) + '.json', 'w') as json_file:
            json.dump(pg_data['data'], json_file)
        a += 50
        b += 1
        time.sleep(3)
    # Menggabungkan JSON
    print("[+] Menggabungkan data produk ...")
    data = []
    for f in glob.glob("data/" + str(self.idseller) + "bukalapak*.json"):
        with open(f) as infile:
            data.extend(json.load(infile))
    with open("data/" + str(self.idseller) + "_bukalapak_all.json", 'w') as outfile:
        json.dump(data, outfile)
    # Membuat CSV
    print("[+] Membuat csv data produk ...")
    f_data = []
    with open("data/" + str(self.idseller) + "_bukalapak_all.json") as f:
        f_read = json.load(f)
        for i in f_read:
            f_data.append([
                self.idseller,
                self.username,
                i['store']['name'],
                i['store']['level']['name'],
                i['store']['premium_level'],
                i['id'],
                i['name'],
                i['category']['name'],
                i['condition'],
                i['rating']['average_rate'],
                i['rating']['user_count'],
                i['stock'],
                i['stats']['interest_count'],
                i['stats']['sold_count'],
                i['stats']['view_count'],
                i['price'],
            ])

```

```

        i['original_price'],
        i['discount_percentage'],
        i['description']
    ])
    f_header = ['id_seller', 'username', 'nama_toko', 'level_toko', 'premium_toko', 'id_produk', 'nama_produk',
                'kategori', 'kondisi', 'rating', 'jumlah_rating', 'stok', 'jumlah_interest', 'jumlah_sold',
                'jumlah_view', 'harga', 'harga_asli', 'diskon', 'deskripsi']
    with open(str(self.idseller) + '_bukalapak.csv', 'w', newline="", encoding='utf-8') as file:
        writer = csv.writer(file)
        writer.writerow(f_header)
        writer.writerows(f_data)
    print('done! ' + str(self.idseller) + '_bukalapak.csv')
    f.close()

# Main program
print("[+] https://github.com/heryandp/bukalapak-product-scrap")
sname = input("[+] Masukkan username seller: https://www.bukalapak.com/u/")
act = Bukalapak(sname)

```

# In[ ]:

**B. Berikut adalah kode dengan penjelasan menggunakan comment untuk memvisualisasikan dalam bentuk barchart**

```

#!/usr/bin/env python
# coding: utf-8

# In[28]:

#!/usr/bin/env python
# coding: utf-8

# Import library yang diperlukan
import pandas as pd
import matplotlib.pyplot as plt

# Membaca database
data = pd.read_csv("C:/PYTHON/579019292_bukalapak.csv")

# Membuat bar chart dengan jumlah_sold sebagai sumbu X dan jumlah_rating sebagai sumbu Y
plt.bar(data['jumlah_sold'], data['jumlah_rating'])

# Menambahkan judul ke plot
plt.title("Jumlah Rating dan Jumlah Sold")

# Menetapkan label sumbu X dan Y
plt.xlabel('Jumlah Sold')
plt.ylabel('Jumlah Rating')

# Menampilkan plot
plt.show()

# In[ ]:

```

**C. Berikut adalah kode dengan penjelasan menggunakan comment untuk memvisualisasikan dalam bentuk Histogram**

```
#!/usr/bin/env python
# coding: utf-8
```

```
# In[1]:
```

```
#!/usr/bin/env python
# coding: utf-8
```

```
# Import library yang diperlukan
import pandas as pd
import matplotlib.pyplot as plt
```

```
# Membaca database
data = pd.read_csv("C:/PYTHON/579019292_bukalapak.csv")
```

```
# Membuat histogram dengan menggunakan data jumlah_rating dan membaginya ke dalam 10 bins
plt.hist(data['jumlah_rating'], bins=10)
```

```
# Menambahkan judul ke plot
plt.title("Histogram Jumlah Rating")
```

```
# Menetapkan label sumbu X dan Y
plt.xlabel('Jumlah Rating')
plt.ylabel('Frekuensi')
```

```
# Menampilkan plot
plt.show()
```

```
# In[ ]:
```