

# Laporan Penjelasan Cara Kerja Coding

Nama : Muhammad Alfin Rahmadani  
Kelas : S.P. 3.1  
NIM : 1123102125

## 1. Deskripsi Program

Program ini menggunakan Selenium untuk melakukan scraping data produk bertema "Banyuwangi" di sebuah toko online. Data yang diambil meliputi nama produk, harga, dan penjual. Hasil scraping ditampilkan di terminal.

---

## 2. Langkah-Langkah Eksekusi

### 2.1 Membuka Browser

Kode memulai dengan membuka browser menggunakan Selenium WebDriver. Dalam hal ini, WebDriver untuk Chrome digunakan:

```
from selenium import webdriver
driver = webdriver.Chrome()
driver.get("https://www.tokopedia.com/")
```

WebDriver akan membuka halaman utama Tokopedia.

### 2.2 Pencarian Tema "Banyuwangi"

Program mencari elemen kotak pencarian di halaman dengan XPath. Tema "Banyuwangi" dimasukkan ke kotak pencarian, kemudian tombol Enter ditekan:

```
search_box = driver.find_element(By.XPATH, '//input[@placeholder="Cari di Tokopedia"]')
search_box.send_keys("Banyuwangi")
search_box.send_keys(Keys.RETURN)
```

Selenium menunggu hingga hasil pencarian dimuat sepenuhnya.

### 2.3 Mengambil Data Produk

Setelah hasil pencarian muncul, Selenium mencari elemen produk berdasarkan kelas CSS yang relevan. Iterasi dilakukan untuk mengambil nama produk, harga, dan penjual:

```
products = driver.find_elements(By.CLASS_NAME, 'css-1sn1xa2')
for product in products:
    name = product.find_element(By.CLASS_NAME, 'css-1bjwylw').text
    price = product.find_element(By.CLASS_NAME, 'css-o5uqvq').text
    seller = product.find_element(By.CLASS_NAME, 'css-1kr22w3').text
```

```
data.append({"name": name, "price": price, "seller": seller})
```

Setiap elemen produk diproses untuk mendapatkan data yang diperlukan. Data tersebut disimpan dalam list data.

## 2.4 Menutup Browser

Setelah data diambil, browser ditutup menggunakan perintah berikut:

```
driver.quit()
```

---

# 3. Penjelasan Teknik dan Fungsi Utama

## 3.1 Selenium WebDriver

- **Fungsi:** Mengotomasi browser untuk melakukan scraping data.
- **Alur:** WebDriver membuka halaman, menjalankan JavaScript jika diperlukan, lalu memungkinkan interaksi seperti pengguna biasa.

## 3.2 XPath dan Selector

- **XPath:** Digunakan untuk menemukan elemen HTML spesifik, seperti kotak pencarian.
- **CSS Selector:** Digunakan untuk menemukan elemen produk di hasil pencarian.

## 3.3 Pengolahan Data

Data produk disimpan dalam dictionary yang mencakup nama, harga, dan penjual. Hasilnya dicetak di terminal untuk memverifikasi keakuratan scraping.

---

# 4. Hasil dan Output

Setelah menjalankan program:

- Browser akan membuka halaman Tokopedia dan melakukan pencarian "Banyuwangi".
- Data produk berupa nama, harga, dan penjual dicetak di terminal.

Contoh output:

Nama Produk: Kopi Banyuwangi Premium  
Harga: Rp50.000  
Penjual: Toko Kopi Nusantara

Nama Produk: Batik Banyuwangi Motif Modern  
Harga: Rp120.000  
Penjual: Batik Asli Banyuwangi

---

## **5. Kesimpulan**

Program ini berhasil mengotomasi pencarian dan scraping data produk di toko online menggunakan Selenium. Teknik ini dapat dikembangkan lebih lanjut untuk mengekstrak data dari sumber lain atau menyimpan hasil dalam format file seperti CSV untuk analisis lebih lanjut.