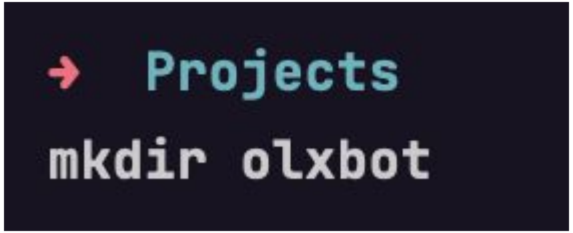
Scraping OLX using Scrapy

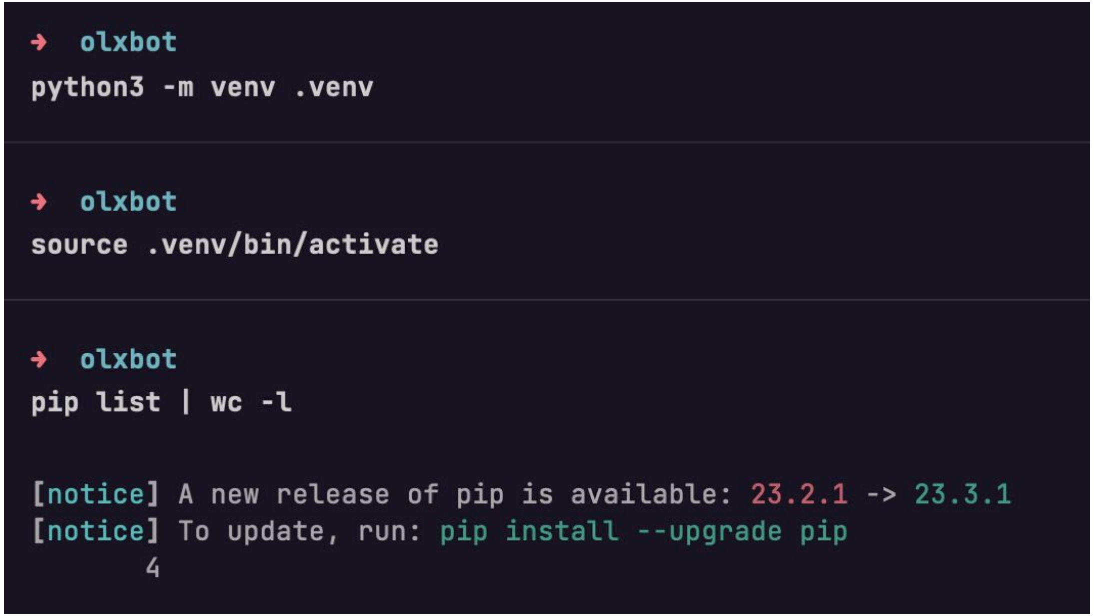
Muhammad Razan Fawwaz
2008107010098

Initialize Project

First of all, we should initialize the Scrapy project. We started it from create a directory and create an virtual environment.



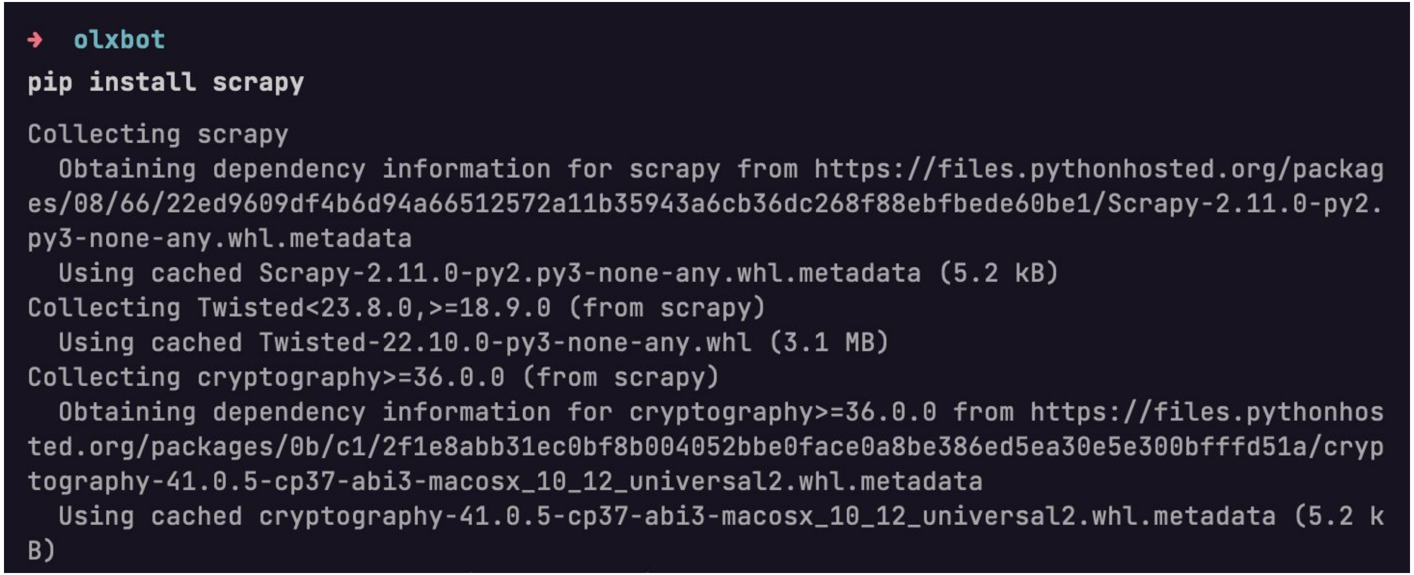
```
➜  Projects
mkdir olxbot
```



```
➜  olxbot
python3 -m venv .venv

➜  olxbot
source .venv/bin/activate

➜  olxbot
pip list | wc -l

[notice] A new release of pip is available: 23.2.1 -> 23.3.1
[notice] To update, run: pip install --upgrade pip
       4
```

Install Scrapy

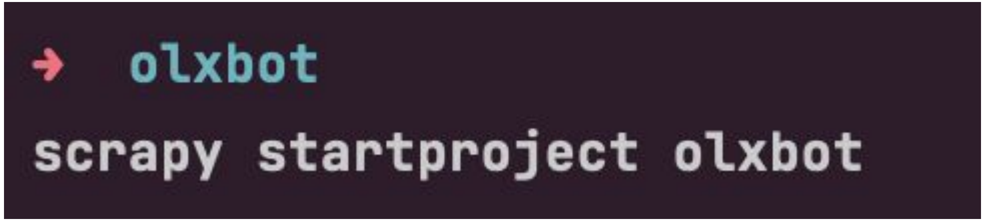
After activating the environment variable, we install the Scrapy by using **pip install scrapy**



```
➜  olxbot
pip install scrapy

Collecting scrapy
  Obtaining dependency information for scrapy from https://files.pythonhosted.org/packag
es/08/66/22ed9609df4b6d94a66512572a11b35943a6cb36dc268f88ebfbede60be1/Scrapy-2.11.0-py2.
py3-none-any.whl.metadata
  Using cached Scrapy-2.11.0-py2.py3-none-any.whl.metadata (5.2 kB)
Collecting Twisted<23.8.0,>=18.9.0 (from scrapy)
  Using cached Twisted-22.10.0-py3-none-any.whl (3.1 MB)
Collecting cryptography>=36.0.0 (from scrapy)
  Obtaining dependency information for cryptography>=36.0.0 from https://files.pythonhos
ted.org/packages/0b/c1/2f1e8abb31ec0bf8b004052bbe0face0a8be386ed5ea30e5e300bfffd51a/cryp
tography-41.0.5-cp37-abi3-macosx_10_12_universal2.whl.metadata
  Using cached cryptography-41.0.5-cp37-abi3-macosx_10_12_universal2.whl.metadata (5.2 k
B)
```

Create Scrapy Project

After the installation finished, we can create a project by using **scrapy startproject** command

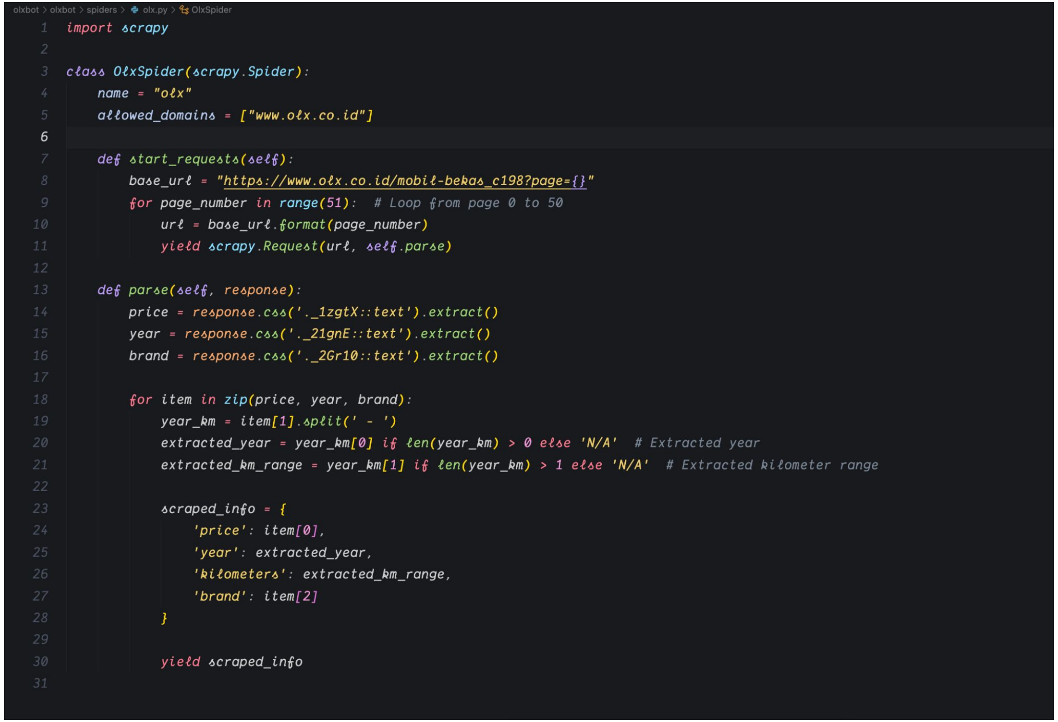


```
➜  olxbot
scrapy startproject olxbot
```

Scraping

After create a project, we go to the spiders directory and create a file called olx.py and put our code

Here I crawl the data from page 0 to 50. Also get the class name by inspect the element of the site.

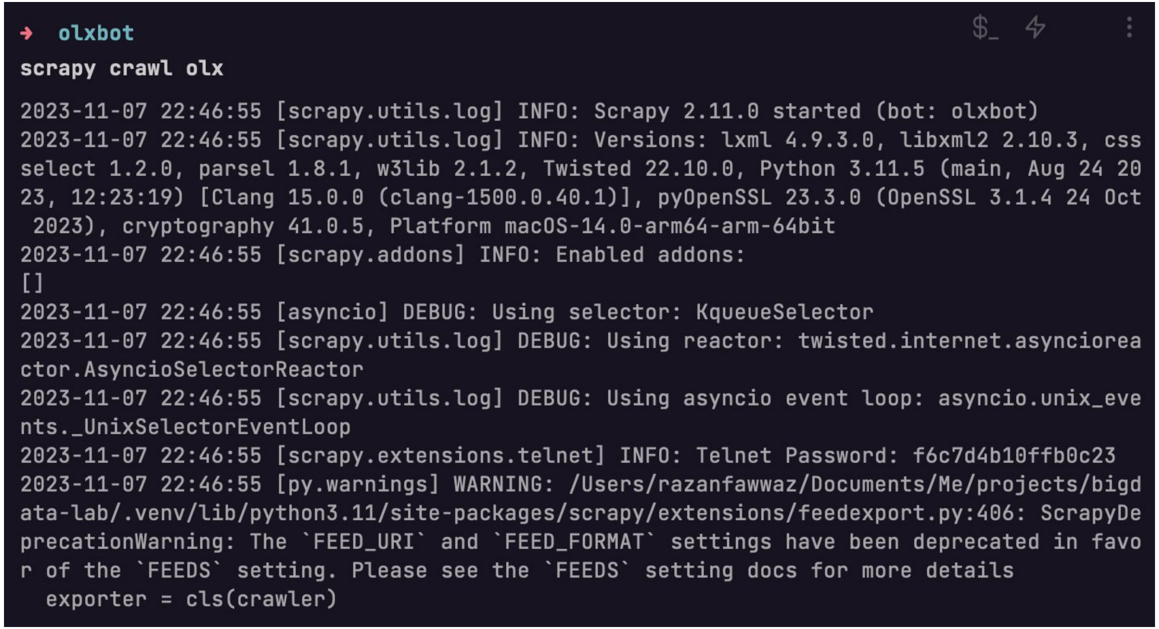


```python
import scrapy

class OlxSpider(scrapy.Spider):
    name = "olx"
    allowed_domains = ["www.olx.co.id"]

    def start_requests(self):
        base_url = "https://www.olx.co.id/mobil-bekas_c198?page={}"
        for page_number in range(51):  # Loop from page 0 to 50
            url = base_url.format(page_number)
            yield scrapy.Request(url, self.parse)

    def parse(self, response):
        price = response.css('._1zgtX::text').extract()
        year = response.css('._21gnE::text').extract()
        brand = response.css('._2Gr10::text').extract()

        for item in zip(price, year, brand):
            year_km = item[1].split(' - ')
            extracted_year = year_km[0] if len(year_km) > 0 else 'N/A'  # Extracted year
            extracted_km_range = year_km[1] if len(year_km) > 1 else 'N/A'  # Extracted kilometer range

            scraped_info = {
                'price': item[0],
                'year': extracted_year,
                'kilometers': extracted_km_range,
                'brand': item[2]
            }

            yield scraped_info

```

Scraping

We start the process by using **scrapy crawl olx** command.

The data will be exported as CSV because we set the format on the settings.

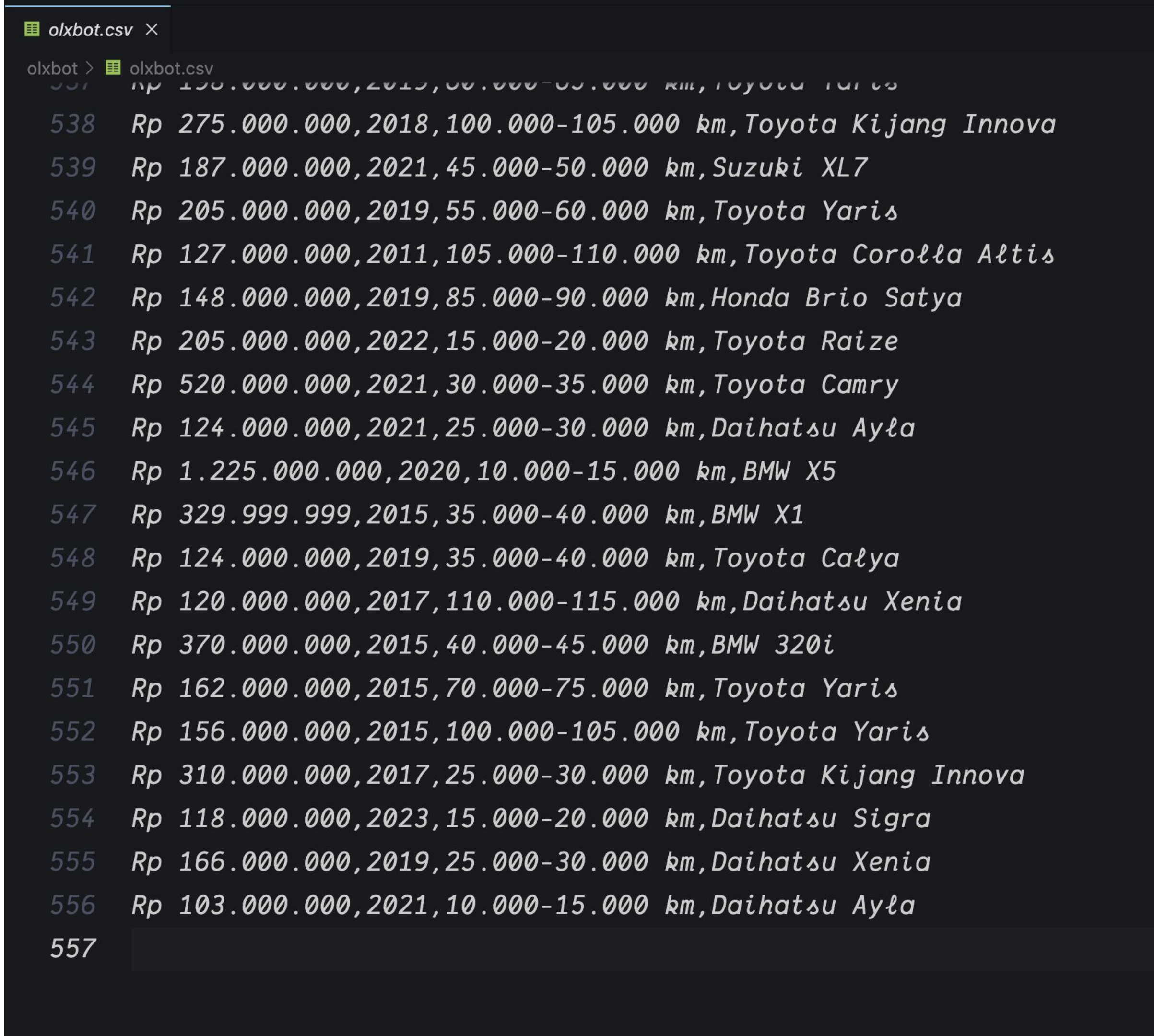


```
➜  olxbot
scrapy crawl olx
2023-11-07 22:46:55 [scrapy.utils.log] INFO: Scrapy 2.11.0 started (bot: olxbot)
2023-11-07 22:46:55 [scrapy.utils.log] INFO: Versions: lxml 4.9.3.0, libxml2 2.10.3, css
select 1.2.0, parsel 1.8.1, w3lib 2.1.2, Twisted 22.10.0, Python 3.11.5 (main, Aug 24 20
23, 12:23:19) [Clang 15.0.0 (clang-1500.0.40.1)], pyOpenSSL 23.3.0 (OpenSSL 3.1.4 24 Oct
 2023), cryptography 41.0.5, Platform macOS-14.0-arm64-arm-64bit
2023-11-07 22:46:55 [scrapy.addons] INFO: Enabled addons:
[]
2023-11-07 22:46:55 [asyncio] DEBUG: Using selector: KqueueSelector
2023-11-07 22:46:55 [scrapy.utils.log] DEBUG: Using reactor: twisted.internet.asynciorea
ctor.AsyncioSelectorReactor
2023-11-07 22:46:55 [scrapy.utils.log] DEBUG: Using asyncio event loop: asyncio.unix_eve
nts._UnixSelectorEventLoop
2023-11-07 22:46:55 [scrapy.extensions.telnet] INFO: Telnet Password: f6c7d4b10ffb0c23
2023-11-07 22:46:55 [py.warnings] WARNING: /Users/razanfawwaz/Documents/Me/projects/bigd
ata-lab/.venv/lib/python3.11/site-packages/scrapy/extensions/feedexport.py:406: ScrapyDe
precationWarning: The `FEED_URI` and `FEED_FORMAT` settings have been deprecated in favo
r of the `FEEDS` setting. Please see the `FEEDS` setting docs for more details
  exporter = cls(crawler)
```

Result

This is the data that we have.

We have price, year, range, and brand



```
Rp 275.000.000,2018,100.000-105.000 km,Toyota Kijang Innova
Rp 187.000.000,2021,45.000-50.000 km,Suzuki XL7
Rp 205.000.000,2019,55.000-60.000 km,Toyota Yaris
Rp 127.000.000,2011,105.000-110.000 km,Toyota Corolla Altis
Rp 148.000.000,2019,85.000-90.000 km,Honda Brio Satya
Rp 205.000.000,2022,15.000-20.000 km,Toyota Raize
Rp 520.000.000,2021,30.000-35.000 km,Toyota Camry
Rp 124.000.000,2021,25.000-30.000 km,Daihatsu Ayla
Rp 1.225.000.000,2020,10.000-15.000 km,BMW X5
Rp 329.999.999,2015,35.000-40.000 km,BMW X1
Rp 124.000.000,2019,35.000-40.000 km,Toyota Calya
Rp 120.000.000,2017,110.000-115.000 km,Daihatsu Xenia
Rp 370.000.000,2015,40.000-45.000 km,BMW 320i
Rp 162.000.000,2015,70.000-75.000 km,Toyota Yaris
Rp 156.000.000,2015,100.000-105.000 km,Toyota Yaris
Rp 310.000.000,2017,25.000-30.000 km,Toyota Kijang Innova
Rp 118.000.000,2023,15.000-20.000 km,Daihatsu Sigra
Rp 166.000.000,2019,25.000-30.000 km,Daihatsu Xenia
Rp 103.000.000,2021,10.000-15.000 km,Daihatsu Ayla
```