

**MechanicalSoup**  
A Python library for automating website interaction.

# Web scraper using Python

---

Bojan Faletic

# Goal of project


---

- Create web agent to search on Ebay in order to find undervalued items

# Overview of program

```
1  #!/bin/python3
2
3  import mechanicalsoup
4  from collection import collection
5  from evaluate import evaluate
6  from sql import write_results
7  from create_url import create_url
8  from ebay_parser_v2 import ebay_parser
9
10 # Searching for
11 SEARCH_FOR = "FPGA"
12
13 # create collection of ebay items
14 0 = ebay_parser(SEARCH_FOR)
15
16 # get actual price from google
17 evaluated_items = evaluate(0)
18
19
20 # write results to database
21 write_results(evaluated_items.status())
```

# Ebay parser



Nexys 2 spartan - 3e FPGA trainer board-FPGA xilinx spartan - 3e

**span.s-item\_price** 77.19 × 24

**\$98.98**

1h 50m left (Today 02:50 PM)  
From Spain

0 bids  
or Best Offer  
+\$33.36 shipping

♥ Watch

```
<div class="s-item_detail s-item_detail--primary">  
  <span class="s-item_price">$98.98</span> == $0  
</div>
```

```
itm = item_bar.find_all("div", class_="s-item__wrapper clearfix")  
for item in itm:  
    name = item.find("h3", class_="s-item__title").text  
    raw_price = item.find(["span", class_="s-item__price"]).text
```

```
class ebay_items:  
    Names = ""  
    Raw_prices = ""  
    Shipping = ""  
    Img_url = ""  
    Bid_cnt = ""  
    Product_url = ""  
    Item_condition = ""
```

# Get price from google



Nexys 2 Spartan-3E FPGA Trainer Board - FPGA XILINX SPART

## Nexys 2 Spartan-3E FPGA Trainer Board (LIMITED TIME ...

<https://store.digilentinc.com> > nexys-2-spartan-3e-fpga-trai... ▾ Prevedi to stran

349,00 \$

**Nexys 2 Spartan-3E FPGA Trainer Board (LIMITED TIME)** ... The **Nexys 2** is a powerful digital system design platform built around a **Xilinx®** Spartan®-3E **FPGA**. ... state-of-the-art development system for less than the **cost** of a textbook! In

```
def evaluate(ebay_items : collection):  
    ranked_items = result()  
    for item in ebay_items.items():  
        google_value = get_prices_from_google(item.Names)  
        if google_value:  
            ranked_items.append(item.Names, item.Raw_prices, google_value.mean_)  
    return ranked_items
```

# Write to database

Name	Price	Actual_price
Filter	Filter	Filter
Nexys 2 spartan - 3e FPGA trainer board-FPGA xilinx spartan - 3e	\$98.98	220.997575757576
Digilent Arty fpga Dev Board (artix-7 35t eval kit)	\$156.11	129
Digilent Xilinx Spartan-3E FPGA Development Board w/ power supply	\$48.00	148.99

```
def write_results(results):  
    conn = create_connection(DB_NAME)  
    with conn:  
        delete_table(conn)  
        for itm in results:  
            task_1 = (itm[0], itm[1], itm[2])  
            create_task(conn, task_1)
```

```
def create_task(conn, task):  
    sql = ''' INSERT INTO items(Name, Price, Actual_price)  
            VALUES(?, ?, ?)'''  
    cursor = conn.cursor()  
    cursor.execute(sql, task)  
    return cursor.lastrowid
```

# Future work

---

- Create front end application
- Add support for multiple search items (better than just array)
- Automatically enter bidding