

Project:

Online Shopping Data Synchronization

Contents

| | |
|--|----------|
| Introduction..... | 1 |
| Step 1: Confirm the Required Data Format and Data Attribute | 2 |
| Step 1.1: Get Ebay Upload Docs | 2 |
| Step 1.2: Confirm the Data attributes and its format from wholesales which are required for Ebay Upload | 2 |
| Step 1.3: Confirm the Data attributes from Ebay Platform which are required for Ebay Upload | 3 |
| Step 2: Web Scrape from Online Website -Python (Selenium) | 4 |
| Step 2.1: Import Selenium and Pandas as main libraries for web scraping and CSV File Modifier | 4 |
| Step 2.2: Apply Edge Browser as Webdrive | 5 |
| Step 2.3: Simulate Search, Click, Find, Copy, Filter Actions by Python Selenium | 6 |
| Step 3: Filter and Reformat Scraped Data into Ebay Bulk Upload CSV Format | 7 |
| Step 4: Test Upload | 7 |
| Step 4.1: Test Upload Product in different Categories with different Type into ebay | 7 |
| Step 4.2: Get Error/Warning Feedback Report, and Rectify | 7 |
| Step 4.3: Final Upload..... | 8 |

Introduction

This project is aimed to synchronize data between Wholesalers and its Distributors, as they all have different online shopping platforms, and the Wholesalers reject to provide API which directly connect with their intranet. Hence, a Website Scrape methodology is required.

Tool and Technology applied:

Python Pandas,

Python Selenium,

Ebay Bulk Upload tool,

Excel Formulas and Excel VBA

Step 1: Confirm the Required Data Format and Data Attribute

Step 1.1: Get Ebay Upload Docs

<https://www.ebay.com.au/sh/reports>

Get Template -> Source -> List -> Type -> Create or Schedule new listings -> File Type -> csv -> Categories -> Men



eBay-category-listing-template-Jul-15-2016

Bulk Upload User Manual:



File_Exchange_Advanced_Instructions.pdf

Step 1.2: Confirm the Data attributes and its format from wholesalers which are required for Ebay Upload

SKU, Title, RelationshipDetails, Colour, Size, PicURL

8. Department:
9. Size Type:
10. Vintage:
11. Handmade:
12. GalleryType:
13. Format:
14. Duration:
15. Quantity:
16. Location:
17. ShippingType:
18. ShippingService-1:Option:
19. ShippingService-1:Cost:
20. *ReturnsAcceptedOption:
21. ReturnsWithinOption:
22. RefundOption:
23. ShippingCostPaidByOption:

Step 2: Web Scrape from Online Website -Python (Selenium)

Step 2.1: Import Selenium and Pandas as main libraries for web scraping and CSV File Modifier

```
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC
import pandas as pd
import time
import re
import csv
```

Step 2.2: Apply Edge Browser as Webdrive

```
import Functional

ListFilePath = ' ';
readingColName = 'SKU'
ItemSKU = Functional.read_columns_from_csv(ListFilePath, readingColName)

driver = Functional.webdriver.Edge()

indexer = 0

while indexer < len(ItemSKU):
    results = Functional.Auto_information_Scrape(ItemSKU[indexer],driver)
    indexer += 1

driver.quit()
```

Step 2.3: Simulate Search, Click, Find, Copy, Filter Actions by Python Selenium

```
# Navigate to the URL where you check the plate number
driver.get("https://www.b12.com.au/" + Website URL)
WebDriverWait(driver, 10).until(EC.presence_of_element_located((By.ID, "search_box"))).send_keys(ItemSKU)
driver.find_element(By.XPATH, "//button[@class='search-button']").click()

# element 1 Product name
WebDriverWait(driver, 10).until(
    EC.presence_of_element_located((By.CLASS_NAME, "product"))
)
time.sleep(2)
# Locate the <div> with id "center-main"
center_main_div = driver.find_element(By.ID, "center-main")

# Get the text content of the <h1> tag
h1_tag = center_main_div.find_element(By.TAG_NAME, "h1")
h1_content = h1_tag.text

# Get the text content of the <h2> tag
h2_tag = center_main_div.find_element(By.TAG_NAME, "h2")
h2_content = h2_tag.text

# Get the text content of the <strong> tag within <h2>
strong_tag = h2_tag.find_element(By.TAG_NAME, "strong")
strong_content = strong_tag.text

# Get the text content of the <span> tag with id "product_price"
product_price_span = center_main_div.find_element(By.ID, "product_price")
product_price_content = product_price_span.text
```

```
import pandas as pd
import numpy as np
import os

# Read the CSV file into a DataFrame
df = pd.read_csv('G:\Data\CSV\AllSizeAllColorAllImage.csv')
csv_file_path = 'G:\Data\CSV\AllSizeAllColorAllImage.csv'

# Convert the DataFrame to a NumPy array
data_array = df.to_numpy()

# Iterate over the rows

SKUArray = []
for row_idx, row in enumerate(data_array):
    if row_idx>0:
        RowData_Variable = ["Add"," "," "," "," "," "," "," "," "," "," "," "," "," "," "," "," "," "," "," "," "," "," "," "," "]
        RowData_Variation =[]
        NotRepeat = False

        #divide Data
        AllSize=""
        AllColor=""
        AllImage=""
        NoRepeatImage = []

        for col_idx, value in enumerate(row):
            if col_idx == 0:
                if value not in SKUArray:
                    NotRepeat = True
                    SKUArray.append(value)
                    RowData_Variable[1] = value
                    RowData_Variable[7] = value
                    RowData_Variable[8] = value
```

| A | B | C | D | E |
|-------------|---------|---------------|-----------|---|
| Line Number | Action | Status | ErrorCode | ErrorMessage |
| 2 | 2 Add | Failure | 21919067 | Error - It looks like this listing is for an item you already have on eBay: BISLEY'S TAPED HI VIS ACTION BACK OVERALL(BAB03597) (296588131741). We don't allow listings for identical items from the same seller to appear on eBay at the same time. If you'd like to list more |
| 3 | 39 Add | Warning | | |
| 4 | 41 Add | Warning | | |
| 5 | 43 Add | Warning | | |
| 6 | 47 Add | Warning | | |
| 7 | 49 Add | Warning | | |
| 8 | 73 Add | Warning | | |
| 9 | 83 Add | Warning | | |
| 10 | 102 Add | Warning | | |
| 11 | 126 Add | Warning | | |
| 12 | 145 Add | Warning | | |
| 13 | 152 Add | Warning | | |
| 14 | 229 Add | Warning | | |
| 15 | 256 Add | Warning | | |
| 16 | 280 Add | Warning | | |
| 17 | 327 Add | Warning | | |
| 18 | 340 Add | Warning | | |
| 19 | 383 Add | Failure | 70 | Error - Listing titles are limited to "80" characters. [0] |
| 20 | 402 Add | Failure | 70 | Error - Listing titles are limited to "80" characters. [0] |
| 21 | 421 Add | Warning | | |
| 22 | 458 Add | Warning | | |
| 23 | 482 Add | Warning | | |
| 24 | 484 Add | Warning | | |
| 25 | 486 Add | Warning | | |
| 26 | 498 Add | Warning | | |
| 27 | 519 Add | Warning | | |
| 28 | 531 Add | Warning | | |
| 29 | 562 Add | Failure | 21919067 | Error - It looks like this listing is for an item you already have on eBay: BISLEY'S TAPED TWO TONE HI VIS 3 IN 1 SOFT SHELL JACKET(BJ6078T) (296588131738). We don't allow listings for identical items from the same seller to appear on eBay at the same time. If you'd like |
| 30 | 583 Add | Failure | 21919067 | Error - It looks like this listing is for an item you already have on eBay: BISLEY'S TAPED HI VIS REVERSIBLE PUFFER JACKET(BJ6350HT) (305690169326). We don't allow listings for identical items from the same seller to appear on eBay at the same time. If you'd like to list r |
| 31 | 604 Add | Warning | | |
| 32 | 615 Add | Warning | | |
| 33 | 626 Add | Warning | | |
| 34 | 637 Add | Warning | | |
| 35 | 648 Add | Warning | | |
| ... | ... | ... | ... | ... |
| | | errorfeedback | + | |

Step 4.3: Final Upload