

Demo Project: Automation with Python (Spreadsheet)

This project demonstrates how to automate spreadsheet processing using Python. The script reads an Excel file, processes the data, and manipulates it to:

1. **List each supplier with their respective product count**
2. **List products with inventory less than 10**
3. **Calculate total inventory value per supplier**
4. **Write the total inventory value for each product back into the spreadsheet**

Step 1: Prepare the Inventory File

Step 2: Install Required Packages

Step 3: Write the Python Script

Step 4: Execute the Script

Step 5: Review the Output

Step 1: Prepare the Inventory File

Ensure you have an Excel file named `inventory.xlsx` with at least the following structure in **Sheet1**:

Product No	Inventory	Price	Supplier	
1	20	200.4	AAA Company	
2	23	33	BBB Company	
3	30000	234.99	CCC Company	
4	43	21.89	BBB Company	
5	523	55.99	CCC Company	
6	54	1150	BBB Company	
7	352	122.55	CCC Company	
8	2352	111	CCC Company	
9	666	345.99	AAA Company	
10	235	646.95	BBB Company	
11	77	13.99	AAA Company	
12	85	2352.55	AAA Company	
13	458	324.5	AAA Company	
14	234	235.99	BBB Company	
15	535	235.98	AAA Company	
16	352	9.99	BBB Company	
17	688	19.99	AAA Company	
18	378	24.9	BBB Company	
19	54	49.5	AAA Company	
20	457	200.4	BBB Company	
21	346	33	AAA Company	
22	745	234.99	AAA Company	
23	45	21.89	AAA Company	
24	54	55.99	AAA Company	
25	7	1150	AAA Company	
26	57346	122.55	AAA Company	

Step 2: Install Required Packages

1. Install the External Python package to work with spreadsheets

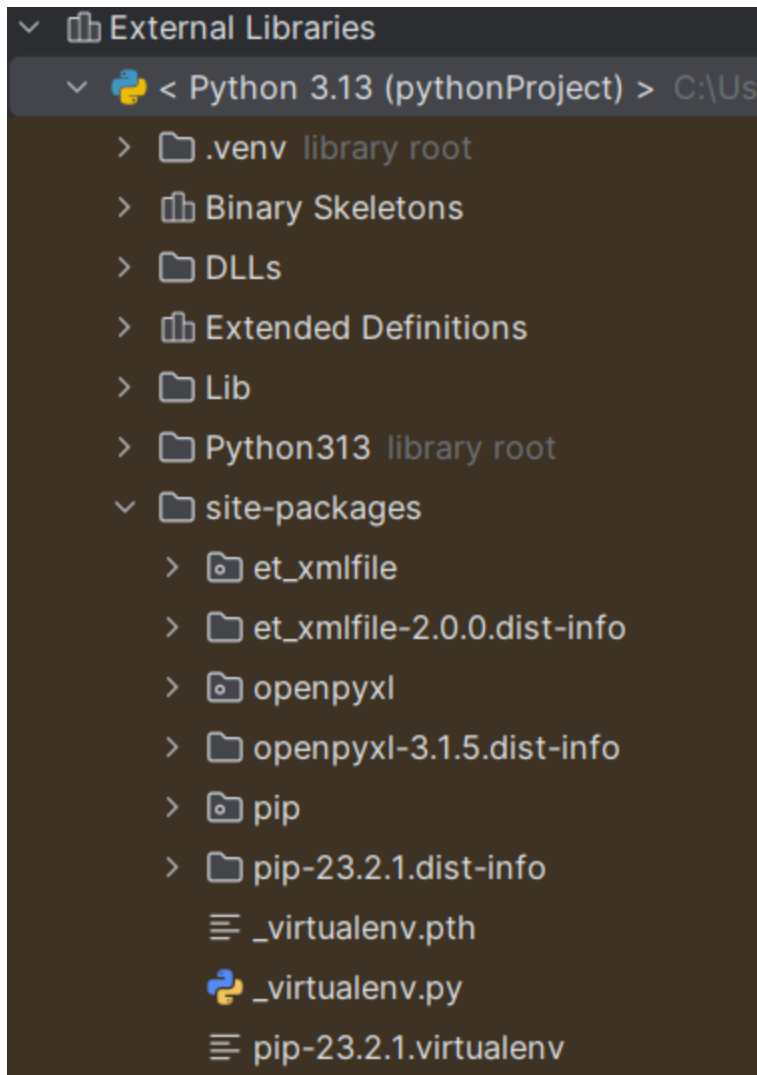
- Visit [pypi.org \(https://pypi.org/project/openpyxl/\)](https://pypi.org/project/openpyxl/) and look for "openpyxl".
- This package allows Python to read, modify, and write Excel files.

2. Run the following command in the terminal: `pip install openpyxl`

3. Verify Installation

- In PyCharm, navigate to **External Libraries** → **Python 3.x** → **site-packages**

- Ensure `openpyxl` is listed.



Step 3: Write the Python Script

Create a file named `main.py` and add the following code:

```
import openpyxl

inv_file = openpyxl.load_workbook("inventory.xlsx")
product_list = inv_file["Sheet1"]
```

```

products_per_supplier = {}
total_value_per_supplier = {}
products_under_10_inv = {}

for product_row in range(2, product_list.max_row + 1):
    supplier_name = product_list.cell(product_row, 4).value
    inventory = product_list.cell(product_row, 2).value
    price = product_list.cell(product_row, 3).value
    product_num = product_list.cell(product_row, 1).value
    inventory_price = product_list.cell(product_row, 5)

    # calculation number of products per supplier
    if supplier_name in products_per_supplier:
        current_num_products = products_per_supplier.get(supplier_name)
        products_per_supplier[supplier_name] = current_num_products + 1
    else:
        products_per_supplier[supplier_name] = 1

    # calculation total value of inventory per supplier
    if supplier_name in total_value_per_supplier:
        current_total_value = total_value_per_supplier.get(supplier_name)
        total_value_per_supplier[supplier_name] = current_total_value + inventory
    * price
    else:
        total_value_per_supplier[supplier_name] = inventory * price

    # logic products with inventory less than 10
    if inventory < 10:
        products_under_10_inv[int(product_num)] = int(inventory)

    # add value for total inventory price
    inventory_price.value = inventory * price

print(products_per_supplier)

```

```
print(total_value_per_supplier)
print(products_under_10_inv)

inv_file.save("inventory_with_total_value.xlsx")
```

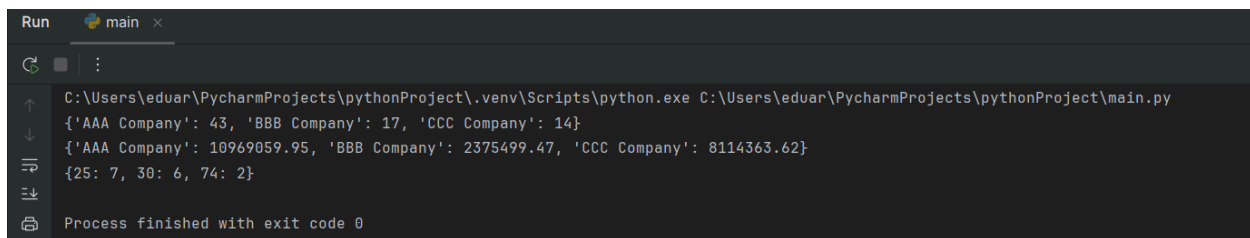
Step 4: Execute the Script

Run in PyCharm

- Open `main.py` in PyCharm.
- Right-click and select **Run 'main'**.

Step 5: Review the Output

- **Terminal Output Example:**



```
Run main x
C:\Users\eduar\PycharmProjects\pythonProject\.venv\Scripts\python.exe C:\Users\eduar\PycharmProjects\pythonProject\main.py
{'AAA Company': 43, 'BBB Company': 17, 'CCC Company': 14}
{'AAA Company': 10969059.95, 'BBB Company': 2375499.47, 'CCC Company': 8114363.62}
{25: 7, 30: 6, 74: 2}
Process finished with exit code 0
```

- **Excel File Changes:**
A new column
Total Value will now contain the calculated values in
inventory_with_total_value.xlsx.

Product No	Inventory	Price	Supplier	
1	20	200.4	AAA Company	4008
2	23	33	BBB Company	759
3	30000	234.99	CCC Company	7049700
4	43	21.89	BBB Company	941.27
5	523	55.99	CCC Company	29282.77
6	54	1150	BBB Company	62100
7	352	122.55	CCC Company	43137.6
8	2352	111	CCC Company	261072
9	666	345.99	AAA Company	230429.34
10	235	646.95	BBB Company	152033.25
11	77	13.99	AAA Company	1077.23
12	85	2352.55	AAA Company	199966.75
13	458	324.5	AAA Company	148621
14	234	235.99	BBB Company	55221.66
15	535	235.98	AAA Company	126249.3
16	352	9.99	BBB Company	3516.48
17	688	19.99	AAA Company	13753.12
18	378	24.9	BBB Company	9412.2
19	54	49.5	AAA Company	2673
20	457	200.4	BBB Company	91582.8
21	346	33	AAA Company	11418
22	745	234.99	AAA Company	175067.55
23	45	21.89	AAA Company	985.05
24	54	55.99	AAA Company	3023.46
25	7	1150	AAA Company	8050