Working with files

Agenda

- Text file
- JSON file
- CSV file
- Excel file

Working with text files

Reading Files

name = open("**filename**")
- opens the given file for reading, and returns a file object

name.read() – file's entire contents as a string
name.readline() – next line from file as a string
name.readlines() – file's contents as a list of lines
- the lines from a file object can also be read using a `for` loop

```
>>> f = open("hours.txt")
>>> f.read()
'123 Susan 12.5 8.1 7.6 3.2\n
456 Brad 4.0 11.6 6.5 2.7 12\n
789 Jenn 8.0 8.0 8.0 8.0 7.5\n'
```

File Input Template

- A template for reading files in Python:

```
name = open("filename")
for line in name:
    statements
```

```
>>> input = open("hours.txt")
>>> for line in input:
...     print(line.strip())    # strip() removes \n

123 Susan 12.5 8.1 7.6 3.2
456 Brad 4.0 11.6 6.5 2.7 12
789 Jenn 8.0 8.0 8.0 8.0 7.5
```

Exercise

- Write a function `input_stats` that accepts a file name as a parameter and that reports the longest line in the file.
  - example input file, `carroll.txt`:

```
Beware the Jabberwock, my son,
the jaws that bite, the claws that catch,
Beware the JubJub bird and shun
the frumious bandersnatch.
```

  - expected output:

```
>>> input_stats("carroll.txt")
longest line = 42 characters
the jaws that bite, the claws that catch,
```

Exercise Solution

```
def input_stats(filename):
    input = open(filename)
    longest = ""
    for line in input:
        if len(line) > len(longest):
            longest = line

    print("Longest line =", len(longest))
    print(longest)
```

Recall: String Methods

Java	Python
`length`	`len(`**`str`**`)`
`startsWith, endsWith`	`startswith, endswith`
`toLowerCase, toUpperCase`	`upper, lower, isupper, islower, capitalize, swapcase`
`indexOf`	`find`
`trim`	`strip`
	`ord, chr`

```
>>> name = "Martin Douglas Stepp"
>>> name.upper()
'MARTIN DOUGLAS STEPP'
>>> name.lower().startswith("martin")
True
>>> len(name)
20
```

String Splitting

- `split` breaks a string into tokens that you can loop over.

  name`.split()` `# break by whitespace`

  name`.split(`**delimiter**`)` `# break by delimiter`

- `join` performs the opposite of a `split`

  delimiter`.join(`**list of tokens**`)`

```
>>> name = "Brave Sir Robin"
>>> for word in name.split():
...     print(word)
Brave
Sir
Robin
>>> "LL".join(name.split("r"))
'BLLave SiLL Robin
```

Splitting into Variables

- If you know the number of tokens, you can `split` them directly into a sequence of variables.

  var1, **var2**, …, **varN** = **string**.split()

- may want to convert type of some tokens: **type**(**value**)

```
>>> s = "Jessica 31 647.28"
>>> name, age, money = s.split()
>>> name
'Jessica'
>>> int(age)
31
>>> float(money)
647.28
```

Exercise

- Suppose we have this `hours.txt` data:

```
123 Suzy 9.5 8.1 7.6 3.1 3.2
456 Brad 7.0 9.6 6.5 4.9 8.8
789 Jenn 8.0 8.0 8.0 8.0 7.5
```

- Compute each worker's total hours and hours/day.
  - Assume each worker works exactly five days.

```
Suzy ID 123 worked 31.4 hours: 6.3 / day
Brad ID 456 worked 36.8 hours: 7.36 / day
Jenn ID 789 worked 39.5 hours: 7.9 / day
```

Exercise Answer

hours.py

```
1  input = open("hours.txt")
2  for line in input:
3      id, name, mon, tue, wed, thu, fri = line.split()
4
5      # cumulative sum of this employee's hours
6      hours = float(mon) + float(tue) + float(wed) + \
7              float(thu) + float(fri)
8
9      print(name, "ID", id, "worked", \
10           hours, "hours: ", hours/5, "/ day"
```

Writing Files

name = open("**filename**", "w")
name = open("**filename**", "a")

- opens file for write (deletes previous contents), or
- opens file for append (new data goes after previous data)

name.write(**str**) – writes the given string to the file

name.close() – saves file once writing is done

```
>>> out = open("output.txt", "w")
>>> out.write("Hello, world!\n")
>>> out.write("How are you?")
>>> out.close()

>>> open("output.txt").read()
'Hello, world!\nHow are you?'
```

Exercise

- Write code to read a file of gas prices in USA and Belgium:

```
8.20    3.81    3/21/11
8.08    3.84    3/28/11
8.38    3.92    4/4/11
...
```

- Output the average gas price for each country to an output file named `gasout.txt`.

python™

Working with JSON files

JSON

JavaScript Object Notation

JSON

- ***JSON*** *(JavaScript Object Notation) is a lightweight data-interchange format. It is easy for humans to read and write. It is easy for machines to parse and generate.*
- Example: (like dict type)

```
{
    "name": “Hien Luong",
    "age": 18,
    "isActive": true
}
```

python™

HIENLTH

Python and JSON

- import json

- JSON to Python

```
jsonData = '{"name": "Hien Luong", "age": 18}'
jsonToPython = json.loads(jsonData)
```

- Python to JSON

```
pythonDictionary = {
    'name':'Hien', 'age':40, 'isEmployed':True
}
dictionaryToJson =
json.dumps(pythonDictionary)
```

Python JSON functions

- Python JSON parsing function
  - obj = **load**(file)
  - obj = **loads**(string)
- Python JSON Serialization Functions
  - **dump**(obj, file)
  - str = **dumps**(obj)

Map Python type to json

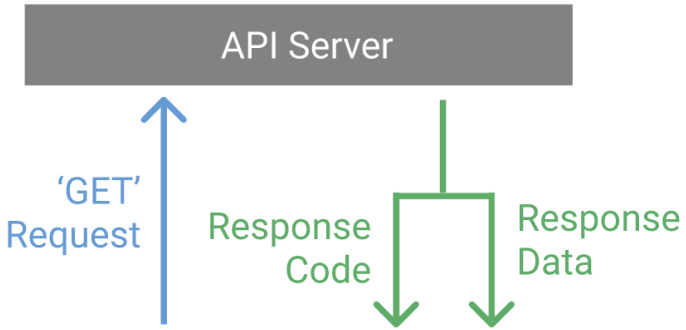
Python	JSON
dict	object
list, tuple	array
str	string
int, float, int- & float-derived Enums	number
True	true
False	false
None	null

Map JSON to python

JSON	Python
object	dict
array	list
string	str
number (int)	int
number (real)	float
true	True
false	False
null	None

What is an API?

- An API, or Application Programming Interface, is a server that you can use to retrieve and send data to using code. APIs are most commonly used to retrieve data.



- Making API Requests in Python
  - `pip install requests`

Working With JSON from an API example

- Call API form https://dummyjson.com/, parse and store in json file

```
import json, requests

result = requests.get('https://dummyjson.com/products')
pythondict = result.json()

print(json.dumps(pythondict, indent=4))
print(list(pythondict.keys()))
```

python™

Working with csv files

(Comma Separated Values)

csv module

Spread sheet and corresponding CSV file

Name	Exam1	Exam2	Final Exam	Overall Grade
Bill	75.00	100.00	50.00	**75.00**
Fred	50.00	50.00	50.00	**50.00**
Irving	0.00	0.00	0.00	**0.00**
Monty	100.00	100.00	100.00	**100.00**
Average				**56.25**

FIGURE 14.2 A simple spreadsheet from Microsoft Excel 2008.

```
Name,Exam1,Exam2,Final Exam,Overall Grade
Bill,75.00,100.00,50.00,75.00
Fred,50.00,50.00,50.00,50.00
Irving,0.00,0.00,0.00,0.00
Monty,100.00,100.00,100.00,100.00

Average,,,,56.25
```

python™

csv file

- https://docs.python.org/3/library/csv.html
- Text file which each rows using (,), (;), or tab (\t) to separate value.
- Python support csv module:
  - csv.field_size_limit – return maximum field size
  - csv.get_dialect – get the dialect which is associated with the name
  - csv.list_dialects – show all registered dialects
  - csv.register_dialect - associate dialect with name
  - csv.unregister_dialect - delete the dialect associated with the name the dialect registry

python™

HIENLTH

csv module

- csv.reader – read data from a csv file
- csv.writer – write data to a csv file
- **csv.QUOTE_ALL** - Quote everything, regardless of type.
- **csv.QUOTE_MINIMAL** - Quote fields with special characters
- **csv.QUOTE_NONNUMERIC** - Quote all fields that aren't numbers value
- **csv.QUOTE_NONE** – Don't quote anything in output

Read csv

```
import csv
workbook_file = open('Workbook1.csv','r')
workbook_reader = csv.reader(workbook_file)

for row in workbook_reader:
    print(row)

workbook_file.close()
```

```
>>>
['Name', 'Exam1', 'Exam2', 'Final Exam', 'Overall Grade']
['Bill', '75.00', '100.00', '50.00', '75.00']
['Fred', '50.00', '50.00', '50.00', '50.00']
['Irving', '0.00', '0.00', '0.00', '0.00']
['Monty', '100.00', '100.00', '100.00', '100.00']
[]
['Average', '', '', '', '56.25']
```

Read file csv into Dictionary

```
import csv
with ('employee_birthday.txt', mode='r') as csv_file:
        csv_reader = csv.DictReader(csv_file)
        for row in csv_reader:
                print(f'\t{row["name"]} - {row[“age"]}.')
```

Write csv

```
import csv
with ('employee_file.csv', mode='w') as employee_file:
        employee_writer = csv.writer(employee_file, delimiter=',',
                                     quotechar='"', quoting=csv.QUOTE_MINIMAL)

        employee_writer.writerow(['John Smith', 'Accounting', 'November’])
        employee_writer.writerow(['Erica Meyers', 'IT', 'March'])
```

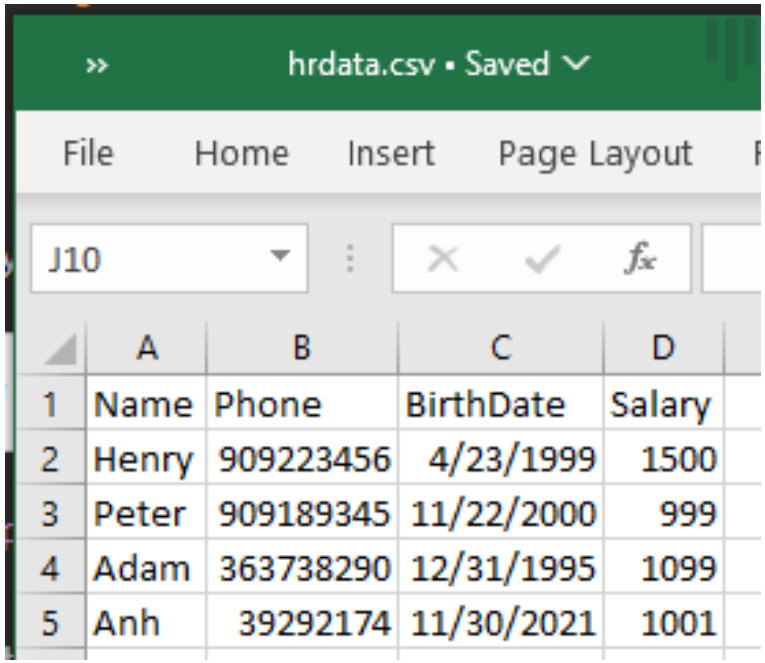
Write csv from Dict

```
import csv
with ('employee_file2.csv', mode='w') as csv_file:
        fieldnames = ['emp_name', 'dept']
        writer = csv.DictWriter(csv_file, fieldnames=fieldnames)
        writer.writeheader()
        writer.writerow({'emp_name': 'John Smith', 'dept': 'Accounting'})
        writer.writerow({'emp_name': 'Erica Meyers', 'dept': 'IT'})
```

Xử lý csv trong thư viện Pandas

```
import pandas
df = pandas.read_csv('hrdata.csv',
index_col='Name')
print(df)
```

```
import pandas
df = pandas.read_csv('hrdata.csv',
index_col='Name',
parse_dates=[`Birth Date'])
print(df)
```



	A	B	C	D
1	Name	Phone	BirthDate	Salary
2	Henry	909223456	4/23/1999	1500
3	Peter	909189345	11/22/2000	999
4	Adam	363738290	12/31/1995	1099
5	Anh	39292174	11/30/2021	1001

Xử lý csv trong thư viện Pandas

```
df = pandas.read_csv('hrdata.csv',
        index_col='Employee',
        parse_dates=['Hired'],
        header=0,
        names=['Employee', 'Hired', 'Salary', 'Sick Days’])

# Process data

# Write new file
df.to_csv('hrdata_modified.csv')
```

python™

HIENLTH

Excel file

Working excel file with openpyxl

- There are many libraries for working (read/write) with Excel file
- Openpyxl library:
  - pip install openpyxl
  - https://openpyxl.readthedocs.io/en/stable

Write data to excel file excel with openpyxl

```
# File excel (Workbook), trong file sẽ có nhiều Worksheet,
# trong worksheet có nhiều cell
from openpyxl import Workbook

wb = Workbook()

# Tạo worksheet có tên NIIE
ws = wb.create_sheet("NIIE", 1)

ws["A1"] = "Viện Đào tạo Quốc tế"        # Ghi ô "A1"

ws.append([2, 3, 4])                     # Thêm dòng mới

wb.save("DemoOpenpyxl.xlsx")             # Lưu file
```

python™

HIENLTH

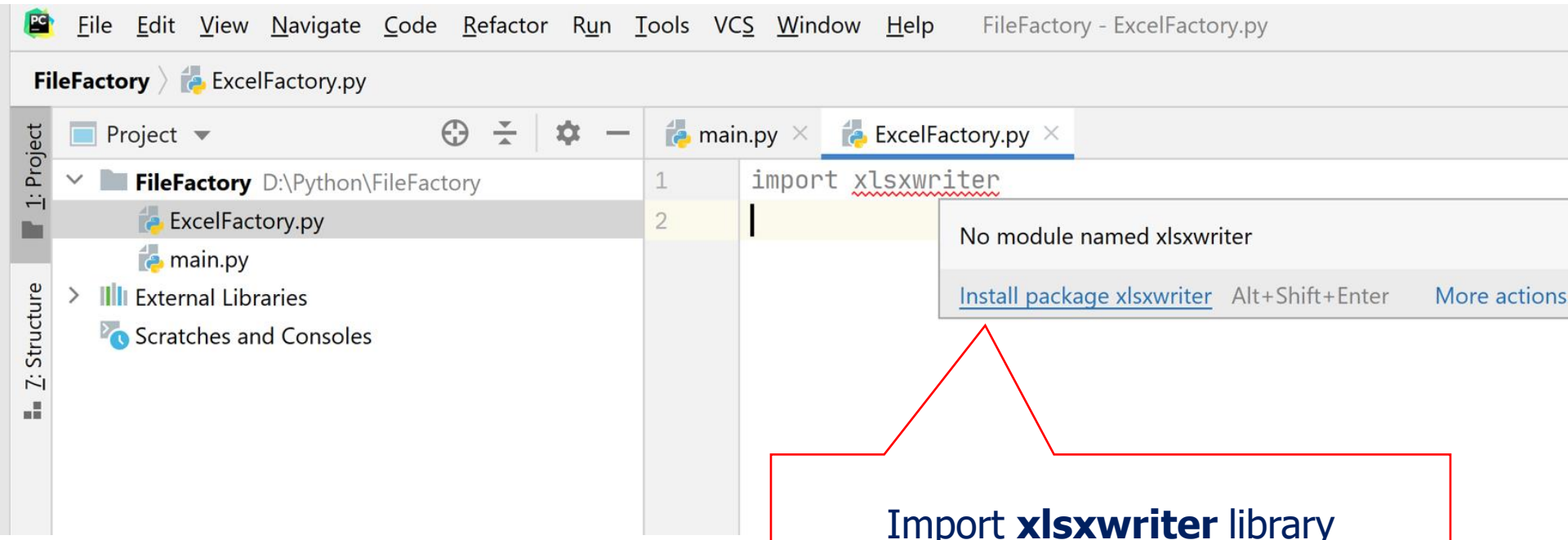
Read data from excel file

```
from openpyxl import load_workbook
wb = load_workbook('demo.xlsx')
print (wb.sheetnames)
ws = wb[wb.sheetnames[0]]
for row in ws.values:
    for value in row:
      print(value,"\t",end='')
    print("")
```

Working excel file with xlsxwriter

- https://xlsxwriter.readthedocs.io/

```
import xlsxwriter
```



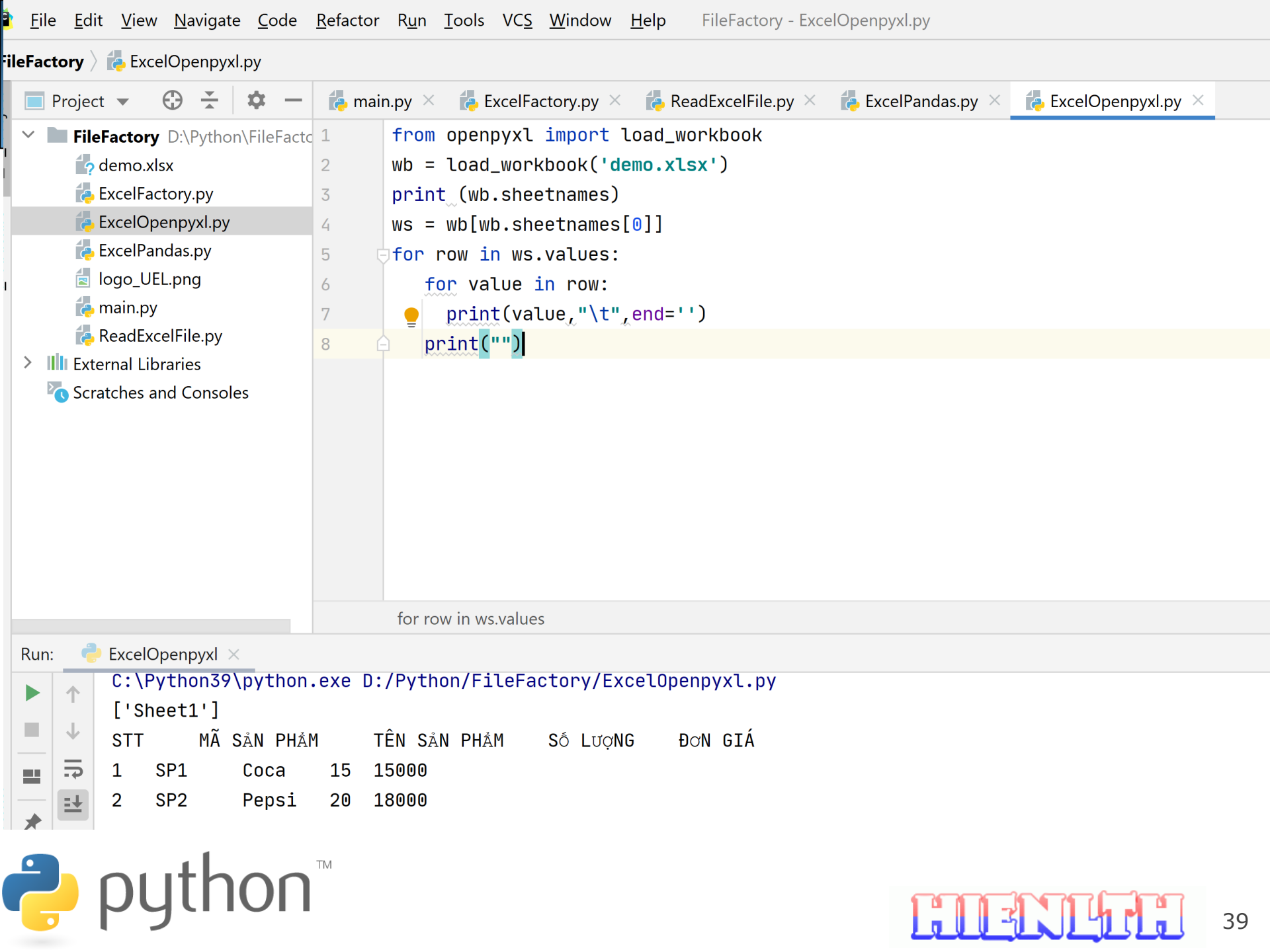
Import **xlsxwriter** library
Hover mouse into line **import**
→Choose Install package xlsxwriter

Working excel file with xlsxwriter

```
import xlsxwriter

# Tạo một file excel cùng 1 sheet
workbook = xlsxwriter.Workbook('demo.xlsx')
worksheet = workbook.add_worksheet()

# thiết lập các cột cho file
worksheet.set_column('A:A', 5)
worksheet.set_column('B:B', 15)
worksheet.set_column('C:C', 20)
worksheet.set_column('D:D', 15)
worksheet.set_column('E:E', 15)

# định dạng tiêu đề cột in đậm
bold = workbook.add_format({'bold': True})

# thêm dòng tiêu đề và định dạng in đậm
worksheet.write('A1', 'STT',bold)
worksheet.write('B1', 'MÃ SẢN PHẨM',bold)
worksheet.write('C1', 'TÊN SẢN PHẨM',bold)
worksheet.write('D1', 'Số LƯỢNG',bold)
worksheet.write('E1', 'ĐƠN GIÁ',bold)
```

```
#thêm một dòng dữ liệu
worksheet.write('A2',1)
worksheet.write('B2','SP1')
worksheet.write('C2', 'Coca')
worksheet.write('D2', '15')
worksheet.write('E2', '15000')
#thêm một dòng dữ liệu
worksheet.write('A3',2)
worksheet.write('B3','SP2')
worksheet.write('C3', 'Pepsi')
worksheet.write('D3', '20')
worksheet.write('E3', '18000')

#Chèn Logo vào
worksheet.insert_image('B5', 'HIENLTH.png')

workbook.close()
```

Chạy phần mềm và vào thư mục phần mềm xem file Excel sẽ có kết quả như mong muốn


File Edit View Navigate Code Refactor Run Tools VCS Window Help
FileFactory - ExcelOpenpyxl.py
FileFactory
ExcelOpenpyxl.py
Project
main.py
ExcelFactory.py
ReadExcelFile.py
ExcelPandas.py
ExcelOpenpyxl.py
FileFactory D:\Python\FileFacto
demo.xlsx
ExcelFactory.py
ExcelOpenpyxl.py
ExcelPandas.py
logo_UEL.png
main.py
ReadExcelFile.py
External Libraries
Scratches and Consoles
1 from openpyxl import load_workbook
2 wb = load_workbook('demo.xlsx')
3 print (wb.sheetnames)
4 ws = wb[wb.sheetnames[0]]
5 for row in ws.values:
6 for value in row:
7 print(value,"\t",end='')
8 print("")
for row in ws.values
Run: ExcelOpenpyxl
C:\Python39\python.exe D:/Python/FileFactory/ExcelOpenpyxl.py
['Sheet1']
STT MÃ SẢN PHẨM TÊN SẢN PHẨM SỐ LƯỢNG ĐƠN GIÁ
1 SP1 Coca 15 15000
2 SP2 Pepsi 20 18000