Essential Python for DA Part2-DSB 11

Key Concepts in Programming

1. Variable
2. Type
3. Structure
4. Control flow
5. Function

sunsun-datateathyme

```
## special topics in Python
```

```
## type hint

name: str = "jay"
age: int = 25
gpa: float = 3.72
netflix: bool = True
```

```
type(name), type(age), type(gpa), type(netflix)
```

```
(str, int, float, bool)
```

```
## type hint in function
## optional
def add(a: int, b: int) -> int:
    return a + b

add(5, 5)
```

```
10
```

```
from typing import List
```

```
friends: List[str] = ["jay", "joe", "jenny"]

for f in friends:
    print(f"Hi! {f}")
```

```
Hi! jay
Hi! joe
Hi! jenny
```

```
## lambda one-liner function
add_lambda = lambda a, b: a + b

add_lambda(5, 5)
```

```
10
```

```
## list comprehension
result = []
original = [1, 2, 3, 4, 5]

for number in original:
    result.append(number * 2)

print(result)
```

```
[2, 4, 6, 8, 10]
```

```
[number * 2 for number in original]
```

```
[2, 4, 6, 8, 10]
```

```
result2 = [number * 2 for number in original]
```

```
print(result2)
```

```
[2, 4, 6, 8, 10]
```

```
## grade
## score >= 80, passed or failed
scores = [88, 75, 72, 90, 95]
grade = ["passed" if score >= 80 else "failed" for score in scores]
```

```
print(grade)
```

```
['passed', 'failed', 'failed', 'passed', 'passed']
```

```
for score in scores:
    if score >= 80:
        print("passed")
    else:
        print("failed")
```

```
passed
failed
failed
passed
passed
```

```
## try - except working with error

try:
    result = 1/0
    print(result)
except:
    print("cannot divide by zero")
```

```
cannot divide by zero
```

```
## try print a variable
datarockie = 500

try:
    print(datarockie)
except:
    print("no variable called 'datarockie'")
else:
    print("the value is printed")
finally:
    print("the end")
```

```
500
the value is printed
the end
```

```
## try print a variable
datarockie = 500

try:
    print(datarockie)
except NameError:
    print("no variable called 'datarockie'")
except ZeroDivisionError:
    print("cannot divide by zero")
except ValueError:
    print("check your value")
else:
    print("the value is printed")
finally:
    print("the end")
```

```
500
the value is printed
the end
```

🎯 How to read csv file in Pyhton

```
## how to read csv file in Pyhton
import csv
```

```
## read file with: context manager
result = []

try:
    with open("customers_arpu.csv", "r") as file:
        reader = csv.reader(file)
        for row in reader:
            result.append(row)
except:
    print("file not found")
```

```
result
```

```
[['\ufeffid', 'name', 'arpu', 'city'],
 ['1', 'john', '500', 'BKK'],
 ['2', 'toy', '250', 'BKK'],
 ['3', 'anne', '300', 'BKK'],
 ['4', 'jessica', '400', 'Lon'],
 ['5', 'joy', '800', 'Lon']]
```

```
result_df = pd.DataFrame(result)
print(result_df)
```

```
      0        1     2     3
0  id     name  arpu  city
1   1     john   500   BKK
2   2      toy   250   BKK
3   3     anne   300   BKK
4   4  jessica   400   Lon
5   5      joy   800   Lon
```

```
## read csv with pandas
import pandas as pd

try:
    df = pd.read_csv("customers_arpu.csv")
    print(df)

except FileNotFoundError:
    print("Error: The file 'customers_arpu.csv' was not found.")
```

```
   id     name  arpu city
0   1     john   500  BKK
1   2      toy   250  BKK
2   3     anne   300  BKK
3   4  jessica   400  Lon
4   5      joy   800  Lon
```

```
df
```

	id	name	arpu	city
0	1	john	500	BKK
1	2	toy	250	BKK
2	3	anne	300	BKK
3	4	jessica	400	Lon
4	5	joy	800	Lon

Next steps: Generate code with df | View recommended plots | New interactive sheet

```
## write csv file
import csv

header = ["id", "student_name", "age"]
body = [
    [1, "jay", 25],
    [2, "ann", 22],
    [3, "joe", 28]
    ]
```

```

with open("new_file.csv", "w") as file:
    writer = csv.writer(file)
    writer.writerow(header)
    writer.writerows(body)
```

```
!cat new_file.csv
```

```
id,student_name,age
1,jay,25
2,ann,22
3,joe,28
```

```
## with + try
try:
    with open("new_file.csv", "r") as file:
        reader = csv.reader(file)
        for row in reader:
            print(row)
except:
    print("file not found")
```

```
['id', 'student_name', 'age']
['1', 'jay', '25']
['2', 'ann', '22']
['3', 'joe', '28']
```

```
Start coding or generate with AI.
```