Hands On – Python

-Sarthak Niranjan Kulkarni (Maverick)

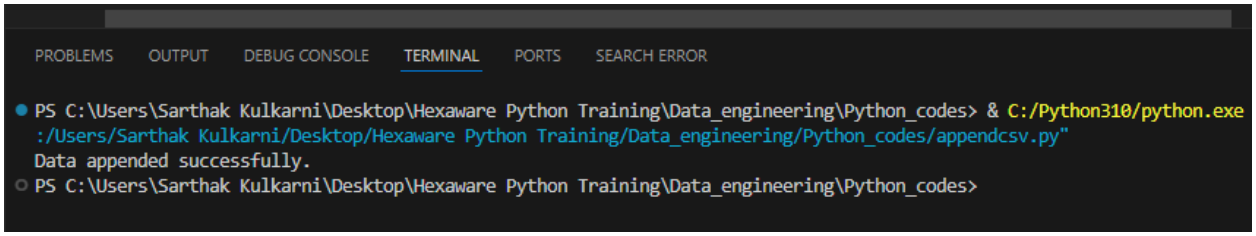
- sarthakkul2311@gmail.com - (+91) 93256 02791

Day 6 – 11/11/2024 (Monday)

1. appendcsv.py

→ import csv

```
new_data = [
    ["Harinya", 32, "Andhra Pradesh"],
    ["Vikas", 21, "Kurnool"]
]
# File path to the CSV file
file_path = "C:/Users/Sarthak Kulkarni/Desktop/Hexaware Python Training/Data_engineering/Day6/example.csv"
# Open the file in append mode and write the new rows
with open(file_path, mode="a", newline="") as file:
    writer = csv.writer(file)
    # Write each row in new_data
    for row in new_data:
        writer.writerow(row)
print("Data appended successfully.")
```



---

2. controlstr.py

→ # if statement

```
x = 10

if x > 5:

    print("x is greater than 5")


# If else statement

x = 3

if x > 5:

    print("x is greater than 5")

else:

    print("x is not greater than 5")


# IF-elif-else statement

x = 5

if x > 10:

    print("x is greater than 10")

elif x > 5:

    print("x is greater than 5")
```

```
else:
    print("x is 5 or less than 5")


# For loop
for i in range(3):
    print("Loop iteration:", i)


# While loop
count = 0
while count < 3:
    print("Count is:", count)
    count += 1


# Nested Loop
for i in range(2):
    for j in range(3):
        print(f"i: {i}, j: {j}")


# Break, Continue and Pass
for i in range(5):
    if i == 4:
        break  # exits loop if i is 4
    elif i == 1:
```

```
        continue  # skips this iteration if i is 1
    print(i)

# Input & Output
name = input("Enter your name: ")
print("Hello,", name)

# Introduction to Lists
fruits = ["apple", "banana", "cherry"]
print(fruits[0])  # Output: apple

# List, Methods and slicing
fruit = ["apple", "banana", "cherry"]
fruit.append("orange")  # Adds "orange"
print(fruit)

# Introduction to Dictionaries & Dictionary Methods
person = {"name": "Sarthak", "age": 22}
print(person["name"])

# Introduction to Set & Set Methods
fruits = {"apple", "banana", "cherry"}
fruits.add("orange")
```

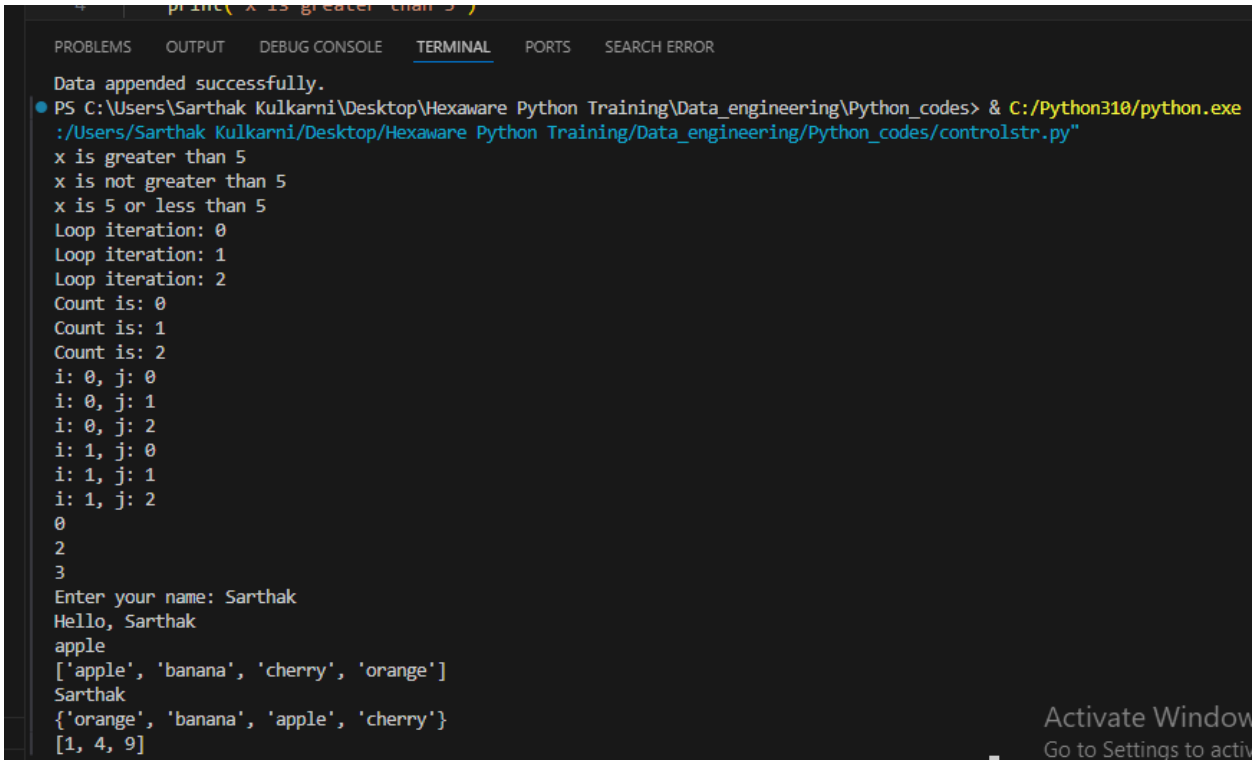
print(fruits)

Introduction to Map & Map Methods

numbers = [1, 2, 3]

squared_numbers = list(map(lambda x: x**2, numbers))

print(squared_numbers)



---

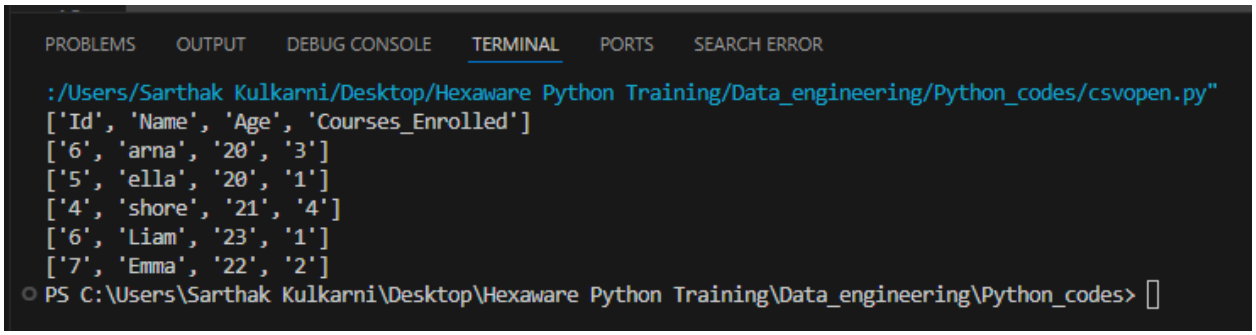
3. csvopen.py

→ import csv

import os

Define the path to the CSV file

```
file_path = "C:/Users/Sarthak Kulkarni/Desktop/Hexaware Python Training/Data_engineering/Python_codes/example2.csv"

# Check if the file exists before opening it

if os.path.exists(file_path):

    # Open the file in read mode

    with open(file_path, mode="r") as file:

        # Create a CSV reader object

        csv_reader = csv.reader(file)



        # Iterate over each row in the CSV file

        for row in csv_reader:

            print(row)  # Print each row

else:

    print("The file does not exist.")
```



---

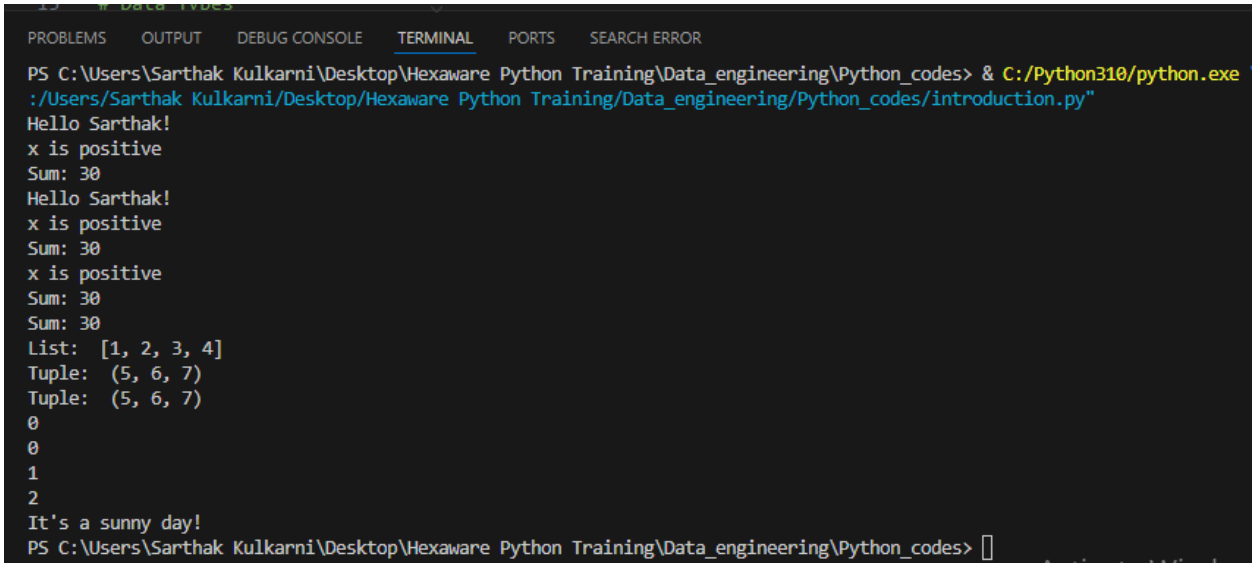
4. introduction.py

→

```
# Writing the first program

print("Hello Sarthak!")
```

```
# Key words and Identifier
x = 5
if x > 0:
    print("x is positive")

# Variables and Operators
a = 10   # Variable assignment
b = 20
sum_result = a + b  # Using the '+' operator
print("Sum:", sum_result)

# Data Types
x = 5           # int
y = 3.14        # float
name = "Python" # str
is_active = True # bool

# Sequence
my_list = [1, 2, 3, 4]   # list
print("List: ",my_list)
my_tuple = (5, 6, 7)     # tuple
print("Tuple: ",my_tuple)
for i in range(3):       # range
```

```
    print(i)


# Boolean
is_sunny = True
if is_sunny:
    print("It's a sunny day!")
else:
    print("It's not sunny.")
```



---

5. practice1.py

→ import csv

```
filepath = "example2.csv"


# Open the file in write mode ('w') which overwrites any existing data
with open(filepath, 'w', newline="") as file:
```

```
csvwriter = csv.writer(file)


# Write header row
csvwriter.writerow(["Id", "Name", "Age", "Courses_Enrolled"])


# Data to write
data = [
    [1, "Aparna", 20, 3],
    [2, "Stella", 20, 1],
    [3, "Kishore", 21, 4]
]


# Write the data rows
csvwriter.writerows(data)


print("Writing has been completed")
```

```
PS C:\Users\Sarthak Kulkarni\Desktop\Hexaware Python Training\Data_engineering\Python_codes> & C:/Pyth
:/Users/Sarthak Kulkarni/Desktop/Hexaware Python Training/Data_engineering/Python_codes/practice1.py"
Writing has been completed
PS C:\Users\Sarthak Kulkarni\Desktop\Hexaware Python Training\Data_engineering\Python_codes>
```

---

6. practice2.py

→ import csv

```
filepath = 'example2.csv'
```

```
# Open the file in write mode ('w'), which overwrites any existing data
with open(filepath, 'w', newline="") as file:
    csvwriter = csv.writer(file)  # Create a writer object

    # Write header row
    csvwriter.writerow(["Id", "Name", "Age", "Courses_Enrolled"])

    # Data to write
    data = [
        [6, "arna", 20, 3],
        [5, "ella", 20, 1],
        [4, "shore", 21, 4]
    ]

    # Write the data rows
    csvwriter.writerows(data)

print("Writing has been completed")
```

```
PS C:\Users\Sarthak Kulkarni\Desktop\Hexaware Python Training\Data_engineering\Python_codes> & C:/Python310/python.exe
:/Users/Sarthak Kulkarni/Desktop/Hexaware Python Training/Data_engineering/Python_codes/practice2.py"
Writing has been completed
PS C:\Users\Sarthak Kulkarni\Desktop\Hexaware Python Training\Data_engineering\Python_codes>
```

7. practice3.py

→ import csv

```
filepath = "example2.csv"


# Open the file in read-and-write mode ('r+')
with open(filepath, 'r+', newline="") as file:
    csvwriter = csv.writer(file)

    # Move the file pointer to the end of the file to append data
    file.seek(0, 2)  # Seek to end of file

    # Data to append
    additional_data = [
        [6, "Liam", 23, 1],
        [7, "Emma", 22, 2]
    ]

    # Append the new data rows
    csvwriter.writerows(additional_data)

print("Data has been appended successfully.")
```

```
PS C:\Users\Sarthak Kulkarni\Desktop\Hexaware Python Training\Data_engineering\Python_codes> & C:/Python310/python.exe
:/Users/Sarthak Kulkarni/Desktop/Hexaware Python Training/Data_engineering/Python_codes/Practice3.py"
Data has been appended successfully.
PS C:\Users\Sarthak Kulkarni\Desktop\Hexaware Python Training\Data_engineering\Python_codes>
```