Hands-on_RegularExpressions

Exercise 1: Check if the string "hello" exists in the sentence "Hello, world!"

➢ Use `re.search()` to check for a match

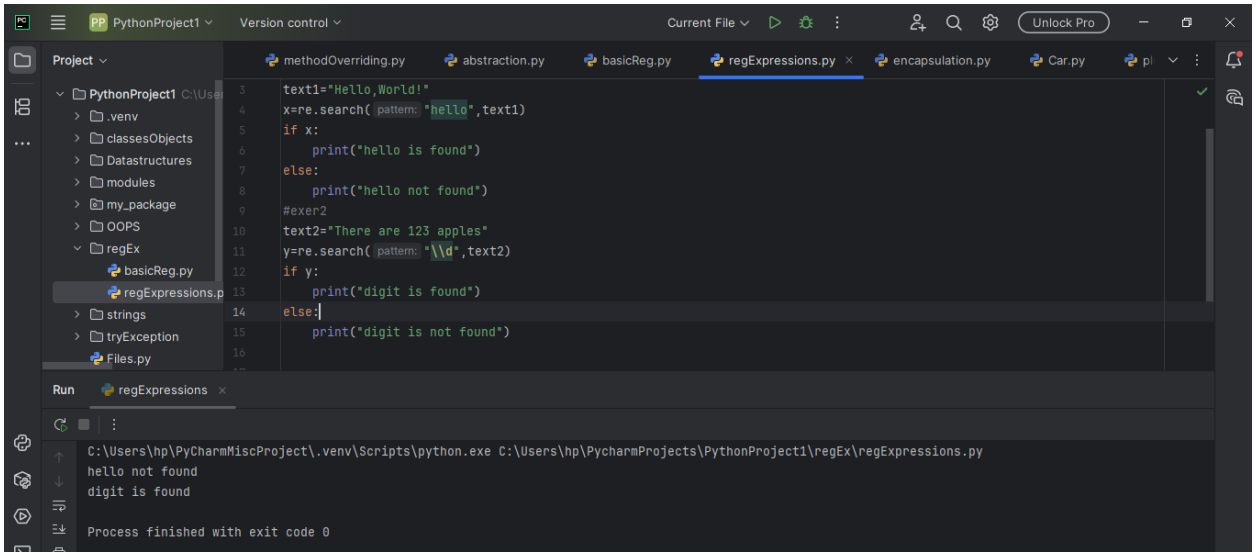
Exercise 2: Search for any number in the string "There are 123 apples"

➢ Use `re.search()` to find the number

Code:

```
import re
#exer1
text1="Hello,World!"
x=re.search("hello",text1)
if x:
    print("hello is found")
else:
    print("hello not found")
#exer2
text2="There are 123 apples"
y=re.search("\\d",text2)
if y:
    print("digit is found")
else:
    print("digit is not found")
```

Output)



Exercise 3: Extract the first email from the text "Contact me at test@example.com for more info"

➤ Use `re.search()` with an appropriate email regex pattern

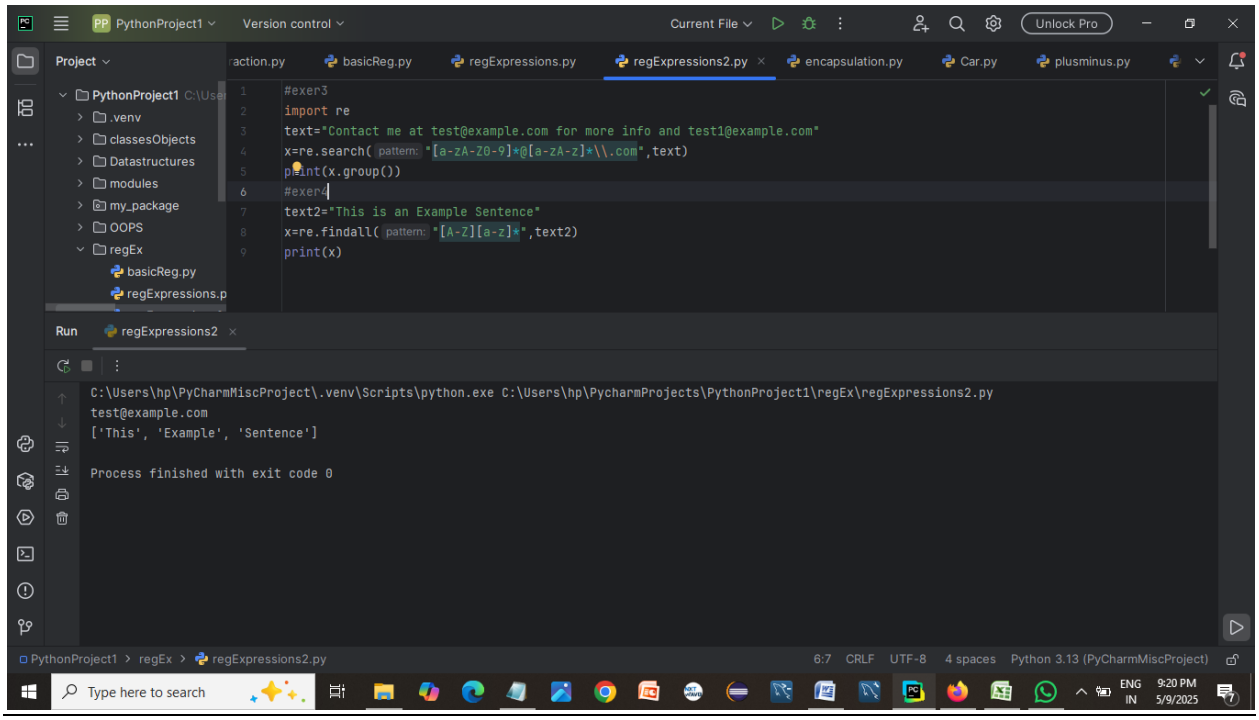
Exercise 4: Extract all words starting with a capital letter from the string "This is an Example Sentence"

➤ Use `re.findall()` to find all such words

Code)

```
#exer3
import re
text="Contact me at test@example.com for more info and test1@example.com"
x=re.search("[a-zA-Z0-9]*@[a-zA-z]*\\.com",text)
print(x.group())
#exer4
text2="This is an Example Sentence"
x=re.findall("[A-Z][a-z]*",text2)
print(x)
```

Output)



Exercise 5: Check if a string contains a valid phone number format (e.g., 123-456-7890)

➤ Use `re.match()` or `re.search()` with a regex pattern for a phone number

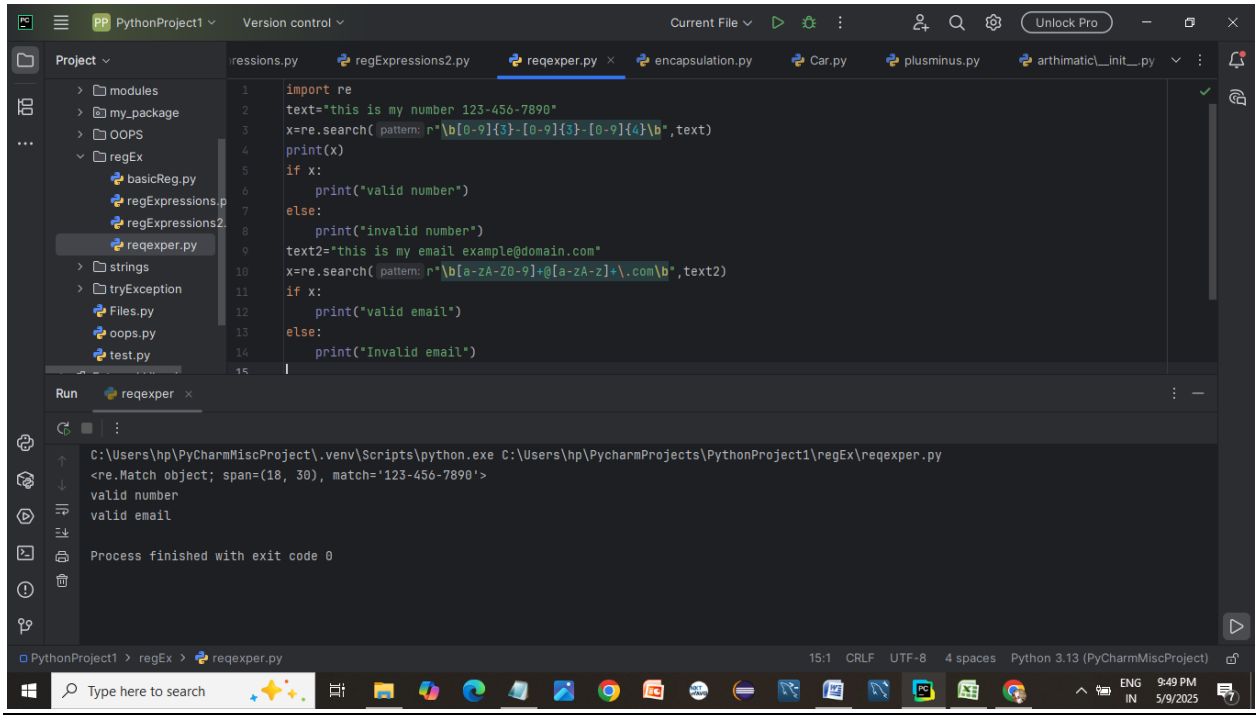
Exercise 6: Validate if a string is a valid email address using a regex pattern

➤ Example: "example@domain.com"

Code)

```
import re
text="this is my number 123-456-7890"
x=re.search(r"\b[0-9]{3}-[0-9]{3}-[0-9]{4}\b",text)
print(x)
if x:
    print("valid number")
else:
    print("invalid number")
text2="this is my email example@domain.com"
x=re.search(r"\b[a-zA-Z0-9]+@[a-zA-z]+\.com\b",text2)
if x:
    print("valid email")
else:
    print("Invalid email")
```

Output)



Exercise 7: Replace all occurrences of "apple" with "orange" in the string "I like apple pie and apple juice"

➤ Use `re.sub()` to replace the text

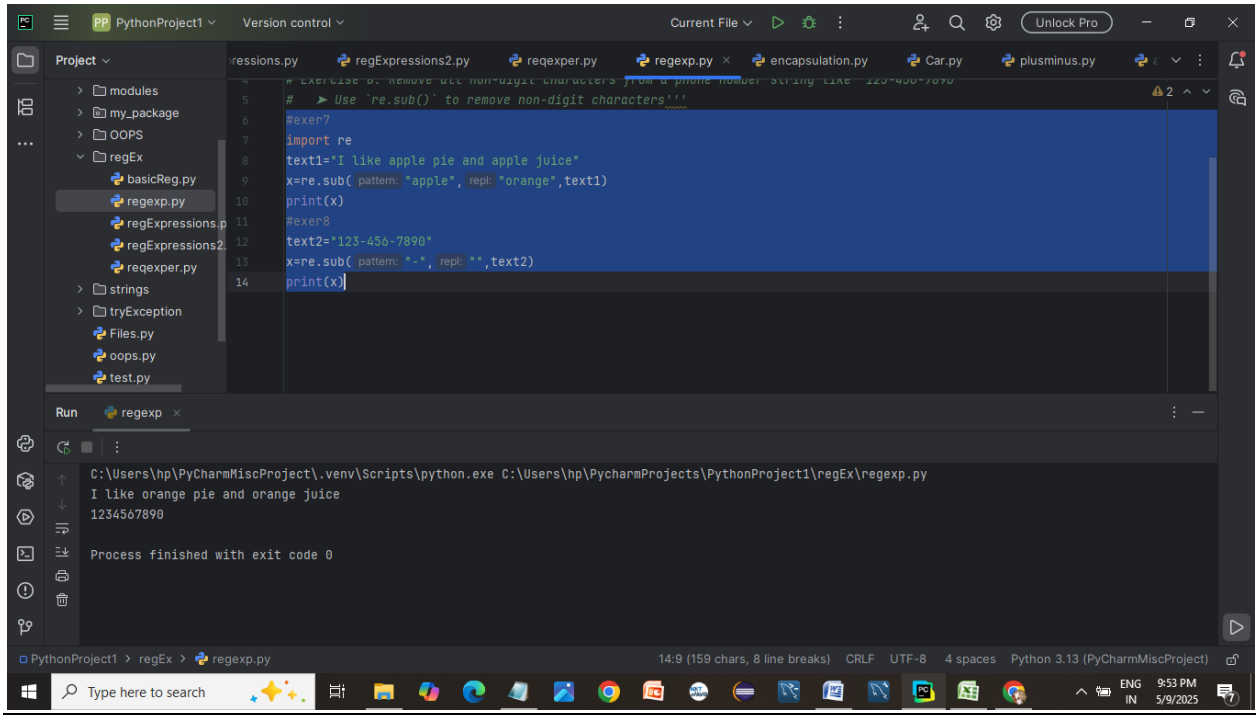
Exercise 8: Remove all non-digit characters from a phone number string like "123-456-7890"

➤ Use `re.sub()` to remove non-digit characters

Code)

```
#exer7
import re
text1="I like apple pie and apple juice"
x=re.sub("apple","orange",text1)
print(x)
#exer8
text2="123-456-7890"
x=re.sub("-","",text2)
print(x)
```

Output)



Exercise 9: Find all the digits in the string "abc123def456ghi"

➤ Use `re.findall()` to extract all digits

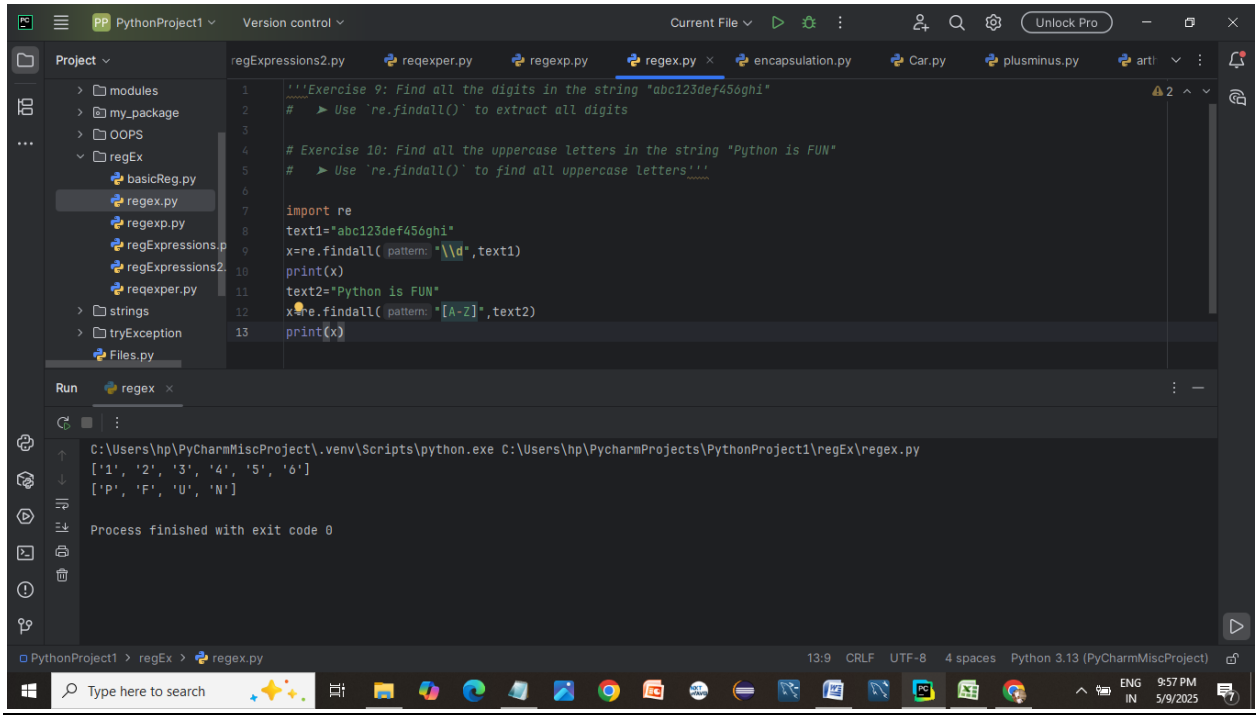
Exercise 10: Find all the uppercase letters in the string "Python is FUN"

➤ Use `re.findall()` to find all uppercase letters

Code)

```
import re
text1="abc123def456ghi"
x=re.findall("\\d",text1)
print(x)
text2="Python is FUN"
x=re.findall("[A-Z]",text2)
print(x)
```

Output)



Exercise 11: Extract the area code from a phone number string like "(123) 456-7890"

➤ Use `re.search()` with capturing groups to extract the area code

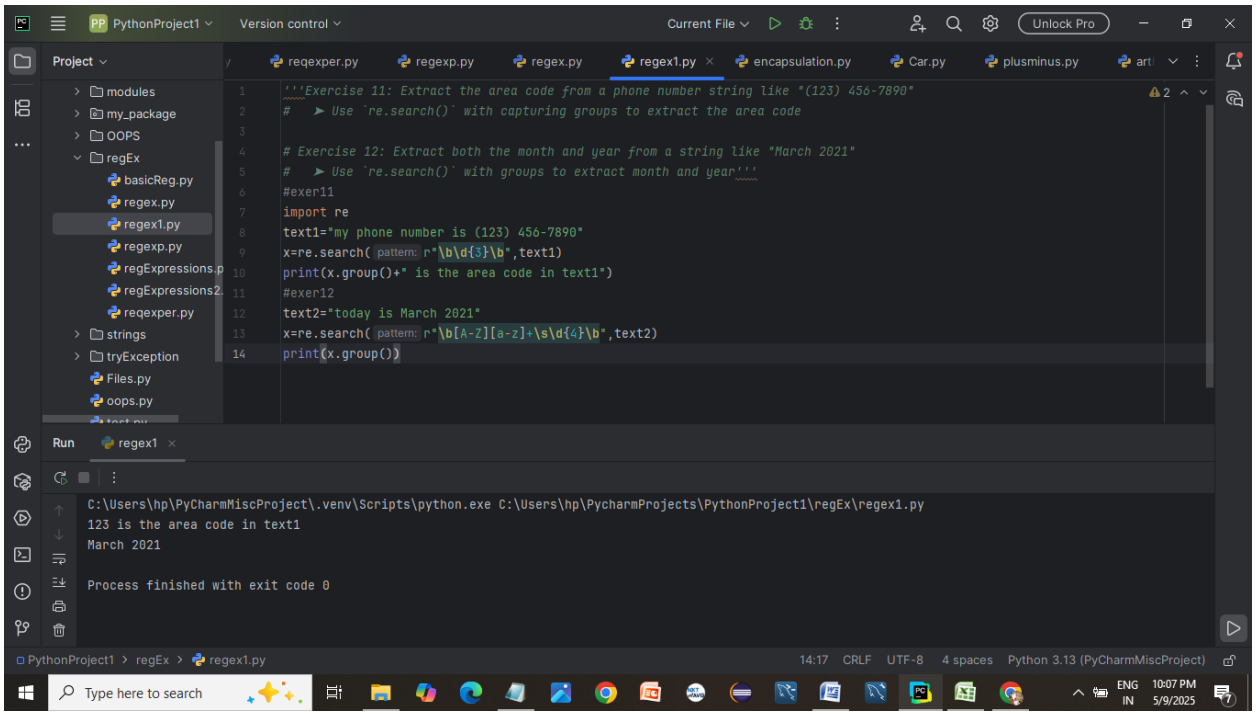
Exercise 12: Extract both the month and year from a string like "March 2021"

➤ Use `re.search()` with groups to extract month and year

Code)

```
#exer11
import re
text1="my phone number is (123) 456-7890"
x=re.search(r"\b\d{3}\b",text1)
print(x.group()+" is the area code in text1")
#exer12
text2="today is March 2021"
x=re.search(r"\b[A-Z][a-z]+\s\d{4}\b",text2)
print(x.group())
```

Output)



Exercise 13: Check if a string starts with "Hello" and ends with "world"

➤ Use `re.match()` with `^` (start) and `$` (end) anchors

Exercise 14: Check if a string contains only digits (using `\d` and `^` for start, `$`

Code)

```
#exer13
import re
text1="Hello boys, we are in a world"
x=re.match("^Hello.*world$",text1)
if x:
    print("text1 starts with Hello and end with world")
else:
    print("donot follow")
#eser14
text2="1234322"
x=re.search("^\\d*$",text2)
if x:
    print("text2 contains only digits")
else:
    print("it contains other chars also")
```

Output)