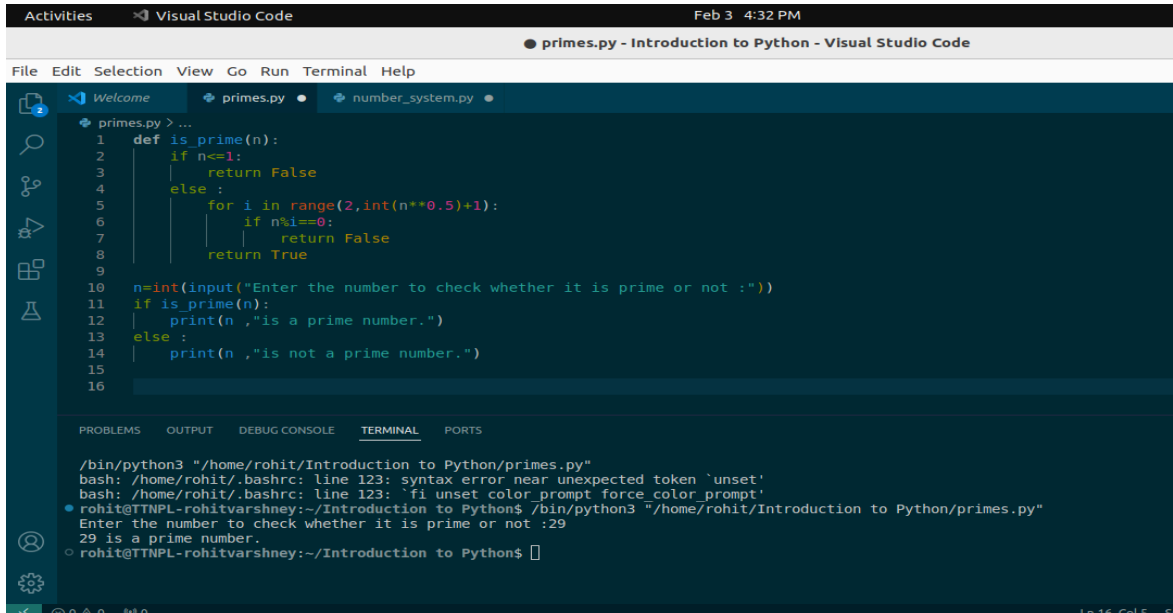


Introduction to Python

1. Write a Python script to test if a number is prime or not? - The Script name: primes.py - Add a function is_prime() which returns boolean True or False - Program should accept a number from console.



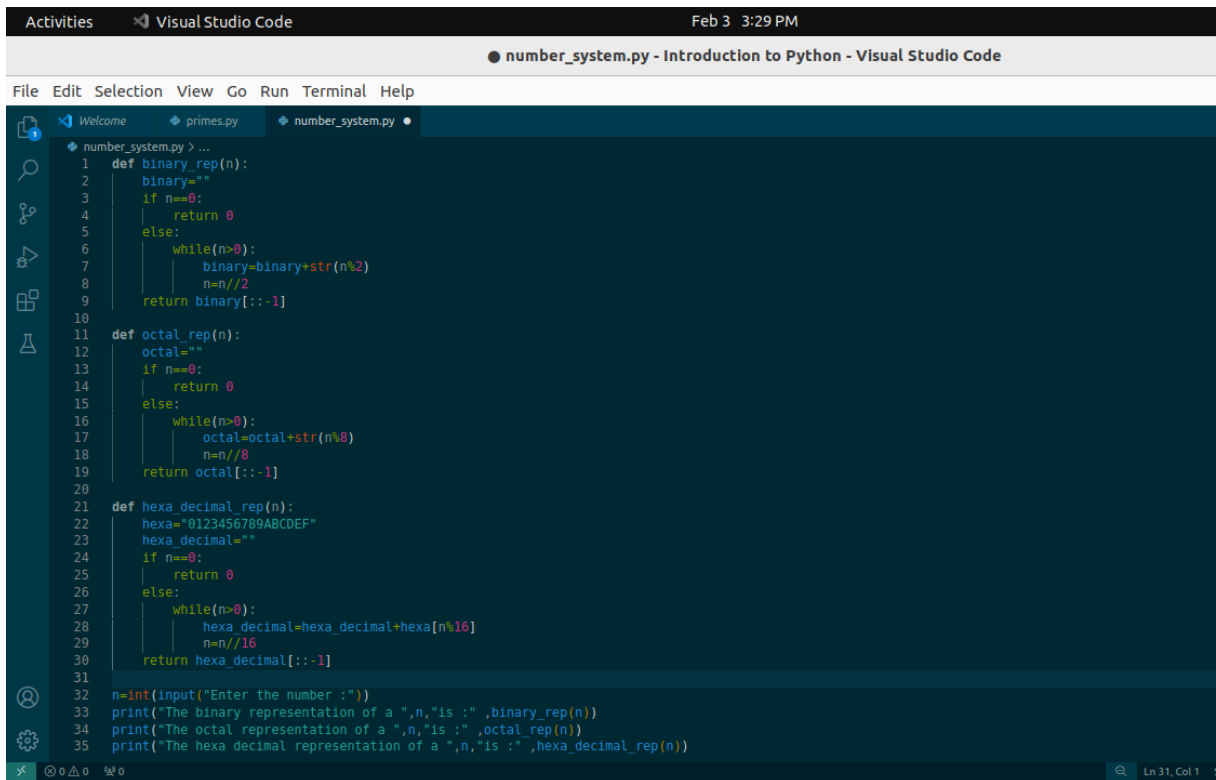
The screenshot shows the Visual Studio Code editor with the file 'primes.py' open. The code defines a function 'is_prime(n)' that checks if a number is prime. It uses a loop to test divisibility from 2 to the square root of n. The main part of the script prompts the user to enter a number and prints the result. The terminal shows the execution of the script, with the user entering '29' and the output '29 is a prime number.'

```
primes.py > ...
1 def is_prime(n):
2     if n<=1:
3         return False
4     else:
5         for i in range(2,int(n**0.5)+1):
6             if n%i==0:
7                 return False
8         return True
9
10 n=int(input("Enter the number to check whether it is prime or not :"))
11 if is_prime(n):
12     print(n,"is a prime number.")
13 else:
14     print(n,"is not a prime number.")
15
16
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
/bin/python3 "/home/rohit/Introduction to Python/primes.py"
bash: /home/rohit/.bashrc: line 123: syntax error near unexpected token `unset'
bash: /home/rohit/.bashrc: line 123: `fi unset color_prompt force_color_prompt'
rohit@TTNPL-rohitvarshney:~/Introduction to Python$ /bin/python3 "/home/rohit/Introduction to Python/primes.py"
Enter the number to check whether it is prime or not :29
29 is a prime number.
rohit@TTNPL-rohitvarshney:~/Introduction to Python$
```

2. Write a code to print binary, octal or hexa-decimal presentation of a number. Do not use any third party library.



The screenshot shows the Visual Studio Code editor with the file 'number_system.py' open. The code defines three functions: 'binary_rep(n)', 'octal_rep(n)', and 'hexa_decimal_rep(n)'. Each function uses a while loop to build the representation of the number in the respective base. The main part of the script prompts the user to enter a number and prints its binary, octal, and hexa-decimal representations.

```
number_system.py > ...
1 def binary_rep(n):
2     binary=""
3     if n==0:
4         return 0
5     else:
6         while(n>0):
7             binary=binary+str(n%2)
8             n=n//2
9         return binary[::-1]
10
11 def octal_rep(n):
12     octal=""
13     if n==0:
14         return 0
15     else:
16         while(n>0):
17             octal=octal+str(n%8)
18             n=n//8
19         return octal[::-1]
20
21 def hexa_decimal_rep(n):
22     hexa="0123456789ABCDEF"
23     hexa_decimal=""
24     if n==0:
25         return 0
26     else:
27         while(n>0):
28             hexa_decimal=hexa_decimal+hexa[n%16]
29             n=n//16
30         return hexa_decimal[::-1]
31
32 n=int(input("Enter the number :"))
33 print("The binary representation of a ",n,"is :",binary_rep(n))
34 print("The octal representation of a ",n,"is :",octal_rep(n))
35 print("The hexa decimal representation of a ",n,"is :",hexa_decimal_rep(n))
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

/bin/python3 "/home/rohit/Introduction to Python/number_system.py"
bash: /home/rohit/.bashrc: line 123: syntax error near unexpected token `unset'
bash: /home/rohit/.bashrc: line 123: `fi unset color_prompt force_color_prompt'
rohit@TTNPL-rohitvarshney:~/Introduction to Python$ /bin/python3 "/home/rohit/Introduction to Python/number_system.py"
Enter the number :15
The binary representation of a 15 is : 1111
The octal representation of a 15 is : 17
The hexa decimal representation of a 15 is : F
rohit@TTNPL-rohitvarshney:~/Introduction to Python$
```

PPT Exercise Questions :

1. Given string my_string = 'Hello Python!', Reverse the string using slicing, print '!' using indexing.

```
Slicing.py • slicing2.py • format_str.py • dictionary1.py • nested.py • create_set.py • count_i.py • sort_dictionary.py •

Slicing.py > ...
1 def Str_reverse(str):
2     return str[::-1]
3
4 str="Hello Python!"
5 result=Str_reverse(str)
6 print("The reverse of given string is :",result)
7
8 # print the "!"
9 print([result[0]])
10

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

bash: /home/rohit/.bashrc: line 123: syntax error near unexpected token `unset'
bash: /home/rohit/.bashrc: line 123: `fi unset color_prompt force_color_prompt'
rohit@TTNPL-rohitvarshney:~/Introduction to Python$ /bin/python3 "/home/rohit/Introduction to Python/Slicing.py"
The reverse of given string is : !nohtyP olleH
!
rohit@TTNPL-rohitvarshney:~/Introduction to Python$
```

2. Use slicing to get word "frain" from "information".

```
Slicing.py • slicing2.py × format_str.py • dictionary1.py • nested.py • create_set.py • count_i.py • sort_dictionary.py •

slicing2.py > ...
1 str2="information"
2 # Getting word "frain" from "information"
3 result=str2[2::2]
4 print([result])

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

/bin/python3 "/home/rohit/Introduction to Python/slicing2.py"
bash: /home/rohit/.bashrc: line 123: syntax error near unexpected token `unset'
bash: /home/rohit/.bashrc: line 123: `fi unset color_prompt force_color_prompt'
rohit@TTNPL-rohitvarshney:~/Introduction to Python$ /bin/python3 "/home/rohit/Introduction to Python/slicing2.py"
frain
rohit@TTNPL-rohitvarshney:~/Introduction to Python$
```

3. Using examples explain string.format and f-strings

```
format_str.py > ...
1 name="Rohit Varshney"
2 company="To The New"
3 # String.formats
4 print("My name is {} and started a job at {}".format(name,company))
5
6 Python=19
7 DBMS=9
8 # f- strings
9 print(f"{name} got marks {Python + DBMS} out of 20")

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

/bin/python3 "/home/rohit/Introduction to Python/format_str.py"
bash: /home/rohit/.bashrc: line 123: syntax error near unexpected token `unset'
bash: /home/rohit/.bashrc: line 123: `fi unset color_prompt force_color_prompt'
rohit@TTNPL-rohitvarshney:~/Introduction to Python$ /bin/python3 "/home/rohit/Introduction to Python/format_str.py"
My name is Rohit Varshney and started a job at To The New
Rohit Varshney got marks 19 out of 20
rohit@TTNPL-rohitvarshney:~/Introduction to Python$
```

4. Can we sort a dictionary? Why or why not?

```
sort_dictionary.py > ...
1 dict1 = {'Java':9, 'Python':10, 'MySQL':10,'AWS':9}
2 print("Before Sorting : ",dict1)
3
4 Keys = list(dict1.keys())
5 Keys.sort()
6
7 sorted_dict={}
8 for i in Keys:
9     sorted_dict[i]=dict1[i]
10 print("The sorted dictionary : ",sorted_dict)
11

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

/bin/python3 "/home/rohit/Introduction to Python/sort_dictionary.py"
bash: /home/rohit/.bashrc: line 123: syntax error near unexpected token `unset'
bash: /home/rohit/.bashrc: line 123: `fi unset color_prompt force_color_prompt'
rohit@TTNPL-rohitvarshney:~/Introduction to Python$ /bin/python3 "/home/rohit/Introduction to Python/sort_dictionary.py"
Before Sorting : {'Java': 9, 'Python': 10, 'MySQL': 10, 'AWS': 9}
The sorted dictionary : {'AWS': 9, 'Java': 9, 'MySQL': 10, 'Python': 10}
rohit@TTNPL-rohitvarshney:~/Introduction to Python$
```

5. Using keys and indexing, grab the 'hello' from the following dictionaries:

- d = {'simple_key':'hello'}
- d = {'k1':{'k2':'hello'}}
- d = {'k1':[{'nest_key':['this is deep',['hello']]]}]}
- d = {'k1':[1,2,{'k2':['this is tricky',{'tough':[1,2,['hello']]}]}]}

```
dictionary1.py > ...
1 d = {'simple_key':'hello'}
2 print(d['simple_key'])
3
4 d = {'k1':{'k2':'hello'}}
5 print(d['k1']['k2'])
6
7 d = {'k1':[{'nest_key':['this is deep',['hello']]}]}
8 print(d['k1'][0]['nest_key'][1][0])
9
10 d = {'k1':[1,2,{'k2':['this is tricky',{'tough':[1,2,['hello']]}]}]}
11 print(d['k1'][2]['k2'][1]['tough'][2][0])
12

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

/bin/python3 "/home/rohit/Introduction to Python/dictionary1.py"
bash: /home/rohit/.bashrc: line 123: syntax error near unexpected token `unset'
bash: /home/rohit/.bashrc: line 123: `fi unset color_prompt force_color_prompt'
rohit@TTNPL-rohitvarshney:~/Introduction to Python$ /bin/python3 "/home/rohit/Introduction to Python/dictionary1.py"
hello
hello
hello
hello
rohit@TTNPL-rohitvarshney:~/Introduction to Python$
```

6. Reassign 'hello' in this nested list to say 'goodbye' instead:

```
list3 = [1,2,[3,4,'hello']]
```

```
Slicing.py • slicing2.py • format_str.py • dictionary1.py • nested.py X • create_set.py • count_i.py • sort_dictionary.py •
nested.py > ...
1 list3 = [1,2,[3,4,'hello']]
2 print("Before replacement : ",list3)
3 # Reassign the "hello" in nested list to say "goodbye" instead :
4 list3[2][2]="goodbye"
5 print("After replacement : ",list3)
6
7

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

/bin/python3 "/home/rohit/Introduction to Python/nested.py"
bash: /home/rohit/.bashrc: line 123: syntax error near unexpected token `unset'
bash: /home/rohit/.bashrc: line 123: `fi unset color_prompt force_color_prompt'
rohit@TTNPL-rohitvarshney:~/Introduction to Python$ /bin/python3 "/home/rohit/Introduction to Python/nested.py"
Before replacement :  [1, 2, [3, 4, 'hello']]
After replacement :  [1, 2, [3, 4, 'goodbye']]
rohit@TTNPL-rohitvarshney:~/Introduction to Python$
```

7. From the given list list5 create a set

```
list5 = [1,2,2,33,4,4,11,22,3,3,2]
```

```
Slicing.py • slicing2.py • format_str.py • dictionary1.py • nested.py • create_set.py X • count_i.py • sort_dictionary.py •
create_set.py > ...
1 list5 = [1,2,2,33,4,4,11,22,3,3,2]
2 print("Before converting the list into set : ",list5)
3 # convert the list into set
4 set5=set(list5)
5 print("After Converting the list into set : ",set5)

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

/bin/python3 "/home/rohit/Introduction to Python/create_set.py"
bash: /home/rohit/.bashrc: line 123: syntax error near unexpected token `unset'
bash: /home/rohit/.bashrc: line 123: `fi unset color_prompt force_color_prompt'
rohit@TTNPL-rohitvarshney:~/Introduction to Python$ /bin/python3 "/home/rohit/Introduction to Python/create_set.py"
Before converting the list into set :  [1, 2, 2, 33, 4, 4, 11, 22, 3, 3, 2]
After Converting the list into set :  {1, 2, 33, 4, 3, 11, 22}
rohit@TTNPL-rohitvarshney:~/Introduction to Python$
```

8. In string information count the total number of i.

```
Slicing.py • slicing2.py • format_str.py • dictionary1.py • nested.py • create_set.py • count_i.py X • sort_dictionary.py •
count_i.py > ...
1 str="information"
2 count=0
3 for char in str:
4     if char=="i":
5         count+=1
6 print("The total occurrence of i : ",count)

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

/bin/python3 "/home/rohit/Introduction to Python/cbash: /home/rohit/.bashrc: line 123: syntax error near unexpected token `unset'
bash: /home/rohit/.bashrc: line 123: `fi unset color_prompt force_color_prompt'
ount_i.py"
rohit@TTNPL-rohitvarshney:~/Introduction to Python$ /bin/python3 "/home/rohit/Introduction to Python/count_i.py"
The total occurrence of i :  2
rohit@TTNPL-rohitvarshney:~/Introduction to Python$
```