

Python Questions

Q.1) Convert given hrs & mins in second

```
def convert_time(hrs,min):  
    min=hrs*60+min  
    sec=min*60  
    return sec
```

```
convert_time(1,0)
```

```
3600
```

```
convert_time(2,30)
```

```
9000
```

```
convert_time(5,10)
```

```
18600
```

The screenshot shows a Google Colab notebook titled "Python Module.ipynb". The notebook contains a Python function `convert_time(hrs,min):` that calculates the total seconds from hours and minutes. The function is defined as follows:

```
def convert_time(hrs,min):  
    min=hrs*60+min  
    sec=min*60  
    return sec
```

The notebook also shows three test cases with their outputs:

- `convert_time(1,0)` returns `3600`
- `convert_time(2,30)` returns `9000`
- `convert_time(5,10)` returns `18600`

The notebook interface includes a top bar with navigation and sharing options, a left sidebar with file and code views, and a bottom status bar showing the execution time (0s) and completion time (16:11). A video call window is visible in the top right corner.

Q.2) Write code to find the average of 'n' numbers entered by the user to function `avg ()`. Ex: `avg (10, 20, 30, 40)` => average is: 70 `avg (5, 10, 15)` => average is: 10

```
#average function  
def Average(l):  
    avg = sum(l)/len(l)  
    return avg
```

```
my_list1 = [10, 20, 30, 40]  
average = Average(my_list1)  
average  
print("Average is",str(average))
```

```
Average is 25.0
```

```
my_list2=[5, 10, 15]
average = Average(my_list2)
average
print("Average is",str(average))
```

```
Average is 10.0
```

The screenshot shows a Google Colab notebook titled "Python Module.ipynb". The notebook contains a function definition and two code cells. The first code cell defines the function and runs it with a list of four numbers, resulting in an average of 25.0. The second code cell runs the function with a list of three numbers, resulting in an average of 10.0. The notebook interface includes a top bar with tabs for "R Programming - Google Drive", "220340325028 - Google Docs", and "Python Module.ipynb - Colaboratory". A green banner indicates "You are screen sharing". A video feed of a man is visible in the top right corner. The bottom of the screen shows a Windows taskbar with various application icons and a system tray displaying the date and time.

Q.2) Write code to find the average of 'n' numbers entered by the user to function avg (. Ex: avg (10, 20, 30, 40) => average is: 70 avg (5, 10, 15) => average is: 10

```
[19] #average function
def Average(l):
    avg = sum(l)/len(l)
    return avg

my_list1 = [10, 20, 30, 40]
average = Average(my_list1)
average
print("Average is",str(average))

Average is 25.0

my_list2=[5, 10, 15]
average = Average(my_list2)
average
print("Average is",str(average))
```

4s completed at 16:51

R Programming.pdf

Type here to search

40°C Mostly sunny 16:55 07-04-2022

Q.3) Print

```
for i in range(1,6):  
    for j in range(1,i+1):  
        print(j%2,end=" ")  
    print()
```

```
1  
1 0  
1 0 1  
1 0 1 0  
1 0 1 0 1
```

The screenshot shows a Google Colab notebook titled "Python Module.ipynb". The code cell contains the following Python code:

```
for i in range(1,6):  
    for j in range(1,i+1):  
        print(j%2,end=" ")  
    print()
```

The output of the code is displayed below the code cell, showing the pattern of 1s and 0s:

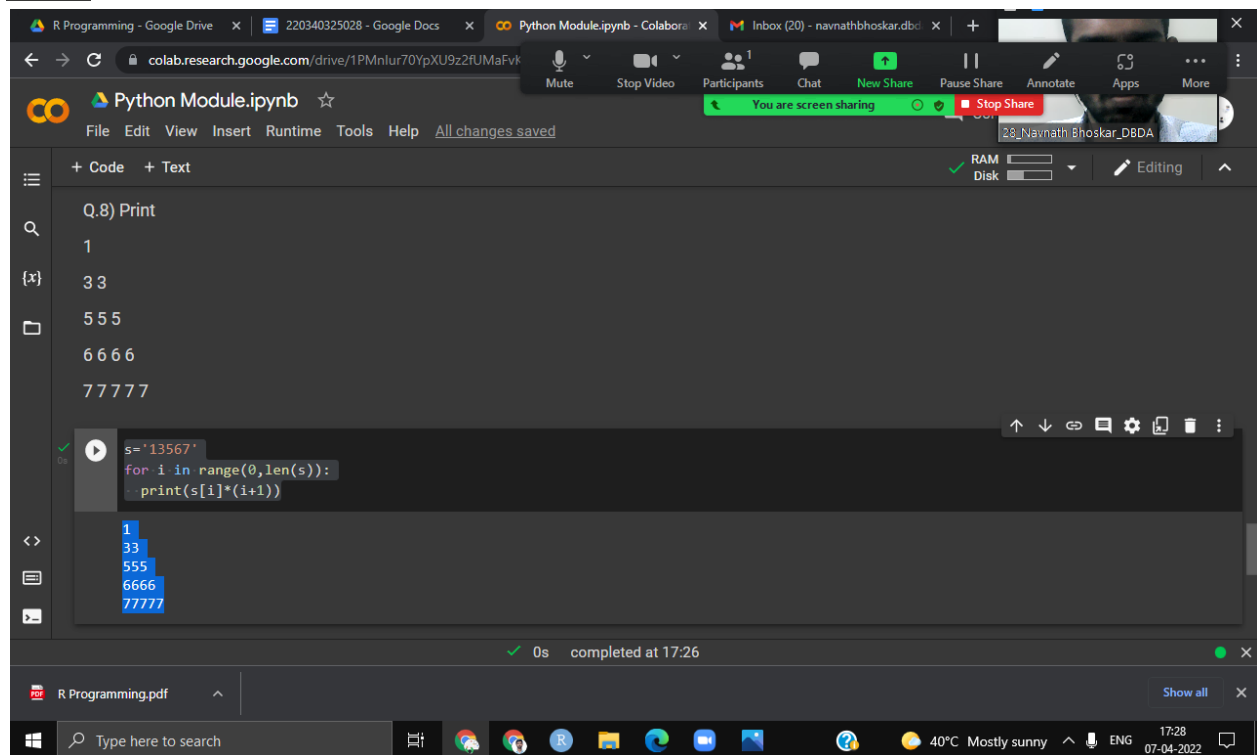
```
1  
1 0  
1 0 1  
1 0 1 0  
1 0 1 0 1
```

The notebook interface includes a top bar with tabs for "R Programming - Google Drive", "220340325028 - Google Docs", and "Python Module.ipynb - Colaboratory". The bottom status bar indicates "0s completed at 17:23".

Q.8) Print

```
s='13567'  
for i in range(0,len(s)):  
    print(s[i]*(i+1))
```

```
1  
33  
555  
6666  
77777
```



The screenshot shows a Google Colab notebook titled "Python Module.ipynb". The code cell contains the following Python code:

```
s='13567'  
for i in range(0,len(s)):  
    print(s[i]*(i+1))
```

The output of the code is displayed in a text area below the code cell:

```
1  
33  
555  
6666  
77777
```

The notebook interface includes a menu bar (File, Edit, View, Insert, Runtime, Tools, Help), a toolbar with icons for running, saving, and sharing, and a status bar at the bottom indicating the execution time (0s) and completion status (completed at 17:26).

Q.5) Create class

```
class Math:
    def __init__(self,a,b):
        self.a=a
        self.b=b
    def Add(self):
        return self.a+self.b

    def Sub(self):
        return self.a-self.b

    def Mul(self):
        return self.a*self.b

    def Div(self):
        return self.a/self.b
```

```
n=Math(50,25)
```

```
print(n.Add())
print(n.Sub())
print(n.Mul())
print(n.Div())
```

```
75
```

```
25
```

```
1250
```

```
75
```

R Programming - Google Drive x 220340325028 - Google Docs x Python Module.ipynb - Colaboratory x Inbox (20) - navnathbhoskar.dbd x

colab.research.google.com/d

Mute Stop Video Participants Chat New Share Pause Share Annotate Apps

You are screen sharing Stop Share

Python Module.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

Q.5) Create class

```
[30] class Math:
      def __init__(self,a,b):
          self.a=a
          self.b=b
      def Add(self):
          return self.a+self.b

      def Sub(self):
          return self.a-self.b

      def Mul(self):
          return self.a*self.b

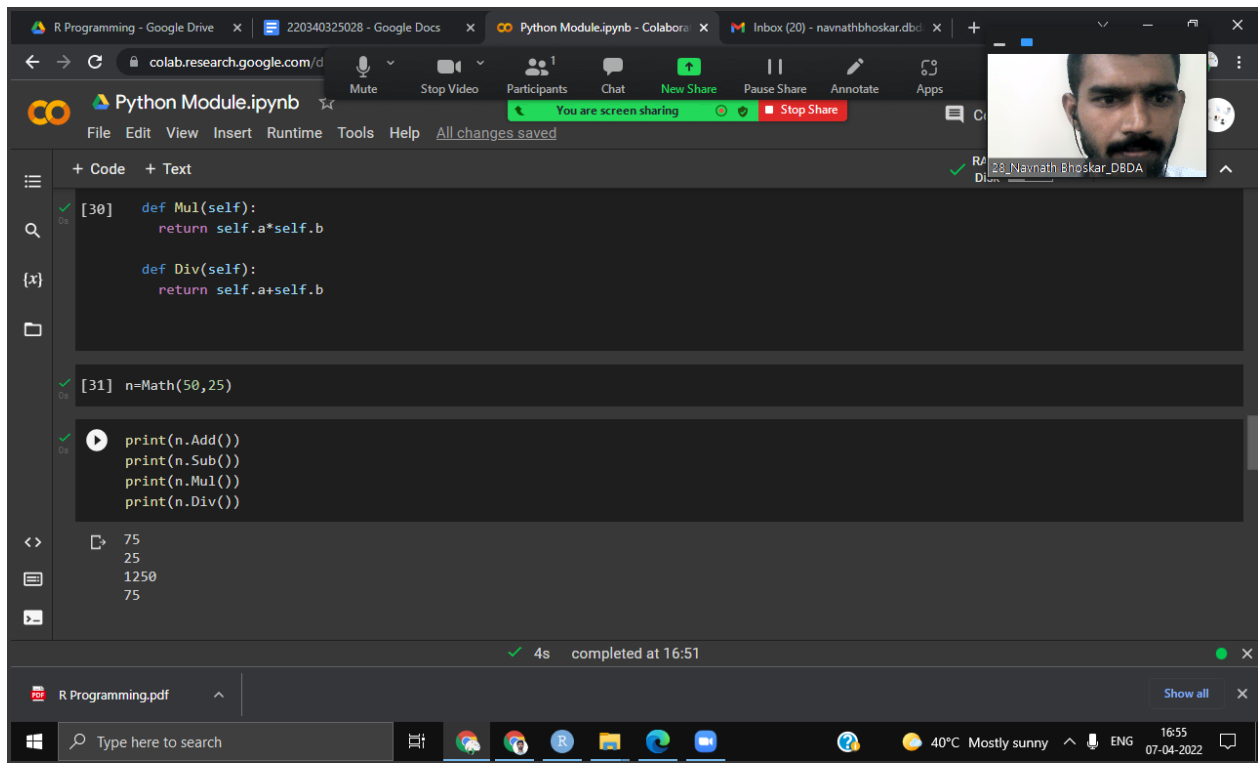
      def Div(self):
          return self.a/self.b
```

4s completed at 16:51

R Programming.pdf Show all

Type here to search

40°C Mostly sunny 16:55 07-04-2022



Q.7) Convert Paise in Rupees & Paises Ex: 350 is 3 Rupees 50 Paise 3.50 print Rupees: 3 Paise: 50

```
amount=int(input("Enter amount in Paise: "))
amount
Rupees = amount // 100
Paises = amount - Rupees*100
print("Rupees:",str(Rupees),"Paise:",str(Paises))
```

```
]

4s

amount=int(input("Enter amount in Paise: "))
amount
Rupees = amount // 100
Paises = amount - Rupees*100
print("Rupees:",str(Rupees),"Paise:",str(Paises))
```

Enter amount in Paise: 350

Rupees: 3 Paise: 50

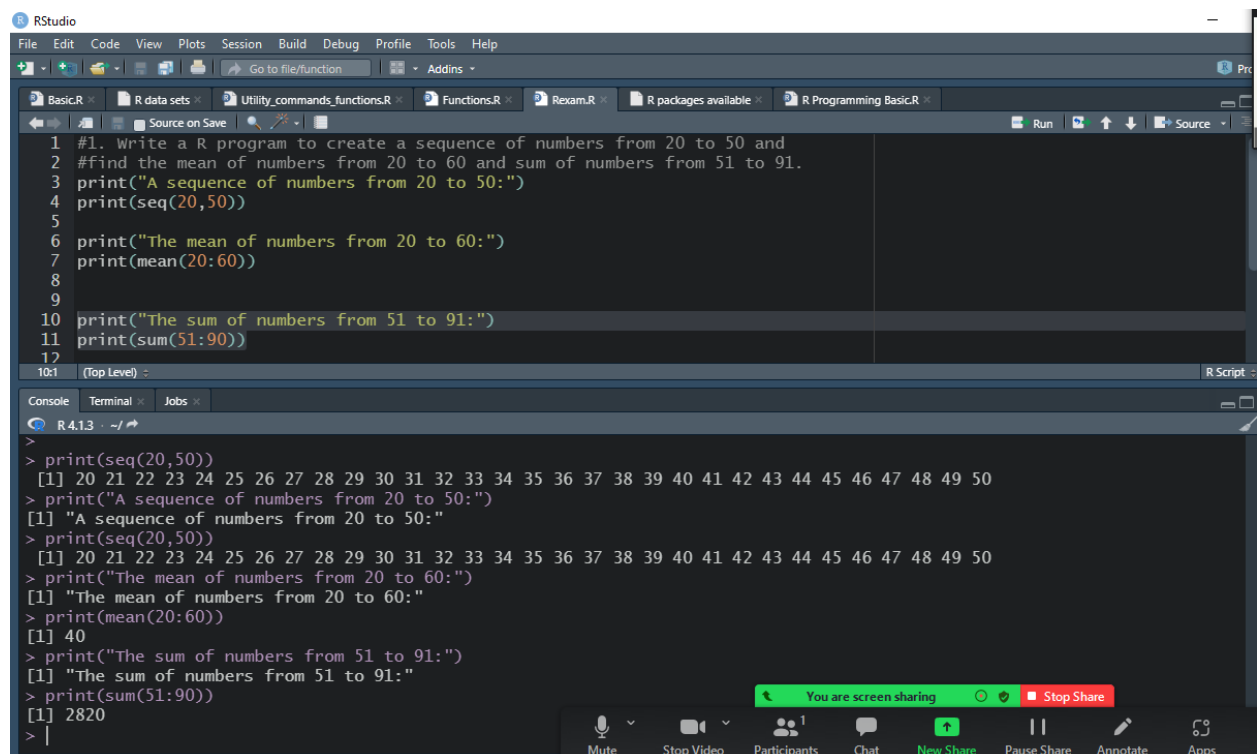
R:-

1. Write a R program to create a sequence of numbers from 20 to 50 and find the mean of numbers from 20 to 60 and sum of numbers from 51 to 91.

```
print("A sequence of numbers from 20 to 50:")  
print(seq(20,50))
```

```
print("The mean of numbers from 20 to 60:")  
print(mean(20:60))
```

```
print("The sum of numbers from 51 to 91:")  
print(sum(51:90))
```

The screenshot shows the RStudio interface. The top pane displays an R script with the following code:

```
1 #1. Write a R program to create a sequence of numbers from 20 to 50 and  
2 #find the mean of numbers from 20 to 60 and sum of numbers from 51 to 91.  
3 print("A sequence of numbers from 20 to 50:")  
4 print(seq(20,50))  
5  
6 print("The mean of numbers from 20 to 60:")  
7 print(mean(20:60))  
8  
9  
10 print("The sum of numbers from 51 to 91:")  
11 print(sum(51:90))  
12
```

The bottom pane shows the console output:

```
>  
> print(seq(20,50))  
[1] 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50  
> print("A sequence of numbers from 20 to 50:")  
[1] "A sequence of numbers from 20 to 50:"  
> print(seq(20,50))  
[1] 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50  
> print("The mean of numbers from 20 to 60:")  
[1] "The mean of numbers from 20 to 60:"  
> print(mean(20:60))  
[1] 40  
> print("The sum of numbers from 51 to 91:")  
[1] "The sum of numbers from 51 to 91:"  
> print(sum(51:90))  
[1] 2820  
>
```

```
> print(seq(20,50))  
[1] 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48  
49 50  
> print("A sequence of numbers from 20 to 50:")
```

```
[1] "A sequence of numbers from 20 to 50:"
```

```
> print(seq(20,50))
```

```
[1] 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48  
49 50
```

```
> print("The mean of numbers from 20 to 60:")
```

```
[1] "The mean of numbers from 20 to 60:"
```

```
> print(mean(20:60))
```

```
[1] 40
```

```
> print("The sum of numbers from 51 to 91:")
```

```
[1] "The sum of numbers from 51 to 91:"
```

```
> print(sum(51:90))
```

```
[1] 2820
```

2. Write a R program to print the numbers from 1 to 100 and print "Fizz" for multiples of 3, print "Buzz" for multiples of 5, and print "FizzBuzz" for multiples of both

```
for (num in 1:100) {  
  if (num %% 3== 0 & num %% 5== 0) {print("FizzBuzz")}  
  else if (num %% 3== 0) {print("Fizz")}  
  else if (num %% 5== 0) {print("Buzz")}  
  else print(num)  
}
```

```
[1] 1  
[1] 2  
[1] "Fizz"  
[1] 4  
[1] "Buzz"  
[1] "Fizz"  
[1] 7  
[1] 8  
[1] "Fizz"  
[1] "Buzz"  
[1] 11  
[1] "Fizz"  
[1] 13  
[1] 14  
[1] "FizzBuzz"  
[1] 16  
[1] 17  
[1] "Fizz"  
[1] 19  
[1] "Buzz"  
[1] "Fizz"  
[1] 22  
[1] 23  
[1] "Fizz"  
[1] "Buzz"  
[1] 26  
[1] "Fizz"  
[1] 28
```

[1] 29
[1] "FizzBuzz"
[1] 31
[1] 32
[1] "Fizz"
[1] 34
[1] "Buzz"
[1] "Fizz"
[1] 37
[1] 38
[1] "Fizz"
[1] "Buzz"
[1] 41
[1] "Fizz"
[1] 43
[1] 44
[1] "FizzBuzz"
[1] 46
[1] 47
[1] "Fizz"
[1] 49
[1] "Buzz"
[1] "Fizz"
[1] 52
[1] 53
[1] "Fizz"
[1] "Buzz"
[1] 56
[1] "Fizz"
[1] 58
[1] 59
[1] "FizzBuzz"
[1] 61
[1] 62
[1] "Fizz"
[1] 64
[1] "Buzz"
[1] "Fizz"
[1] 67
[1] 68
[1] "Fizz"
[1] "Buzz"
[1] 71
[1] "Fizz"

[1] 73
[1] 74
[1] "FizzBuzz"
[1] 76
[1] 77
[1] "Fizz"
[1] 79
[1] "Buzz"
[1] "Fizz"
[1] 82
[1] 83
[1] "Fizz"
[1] "Buzz"
[1] 86
[1] "Fizz"
[1] 88
[1] 89
[1] "FizzBuzz"
[1] 91
[1] 92
[1] "Fizz"
[1] 94
[1] "Buzz"
[1] "Fizz"
[1] 97
[1] 98
[1] "Fizz"
[1] "Buzz"

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function Addins

Basic.R R data sets Utility_commands_functions.R Functions.R Exam.R Exam2.R R packages available R Programming Basic.R

Source on Save Run

```
1 #2. Write a R program to print the numbers from 1 to 100 and print "Fizz" for multiples
2 # of 3, print "Buzz" for multiples of 5, and print "FizzBuzz" for multiples of both
3
4 for (num in 1:100) {
5   if (num %% 3 == 0 & num %% 5 == 0) {print("FizzBuzz")}
6   else if (num %% 3 == 0) {print("Fizz")}
7   else if (num %% 5 == 0) {print("Buzz")}
8   else print(num)
9 }
```

4:1 (Top Level) R Script

Console Terminal Jobs

R 4.1.3 ~/
[1] "Fizz"
[1] 43
[1] 44
[1] "FizzBuzz"
[1] 46
[1] 47
[1] "Fizz"
[1] 49
[1] "Buzz"
[1] "Fizz"
[1] 52
[1] 53
[1] "Fizz"
[1] "Buzz"
[1] 56
[1] "Fizz"
[1] 58
[1] 59

Mute Stop Video Participants Chat New Share Pause Share Annotate Apps More

You are screen sharing Stop Share

28_Navnath_Bhoskar_D8DA

User Library