Department of Networking and Communications

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC207J

Course Name: Advanced Programming Practice

Experiment No	1
Title of Experiment	To complete all the 20 problems in Jupyter environment
Name of the Student	**Abhijay Rajvansh** (RA2011003010398)
Date of Experiment	18 / 03 / 2022

Staff Signature with date

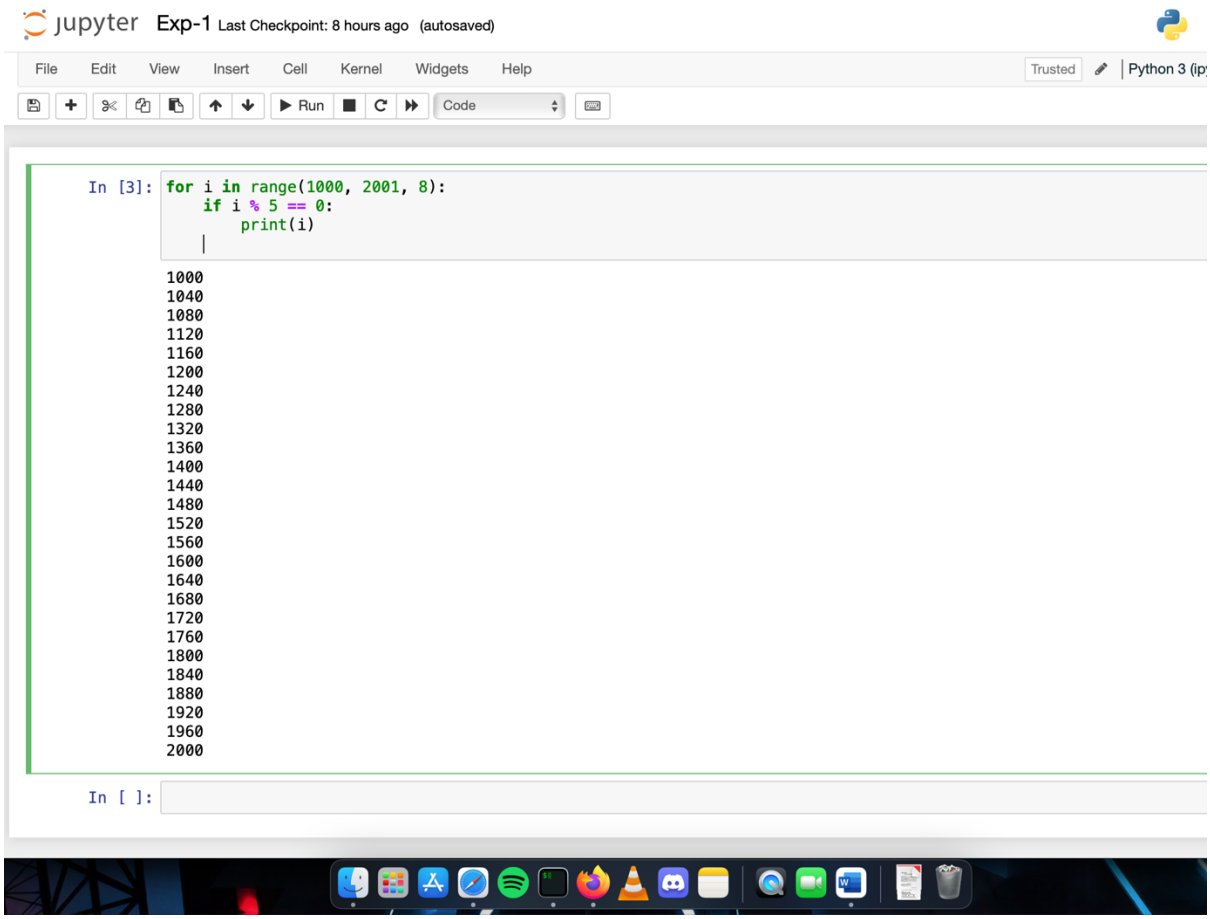
AIM

1. Write a Python program to find those numbers which are divisible

by 8 and multiple of 5, between 1000 and 2000 (both included)

CODE:

```
for i in range(1000, 2001, 8):
    if i % 5 == 0:
        print(i)
```



Result

Python program to find those numbers which are divisible

by 8 and multiple of 5, between 1000 and 2000 (both included) was completed.

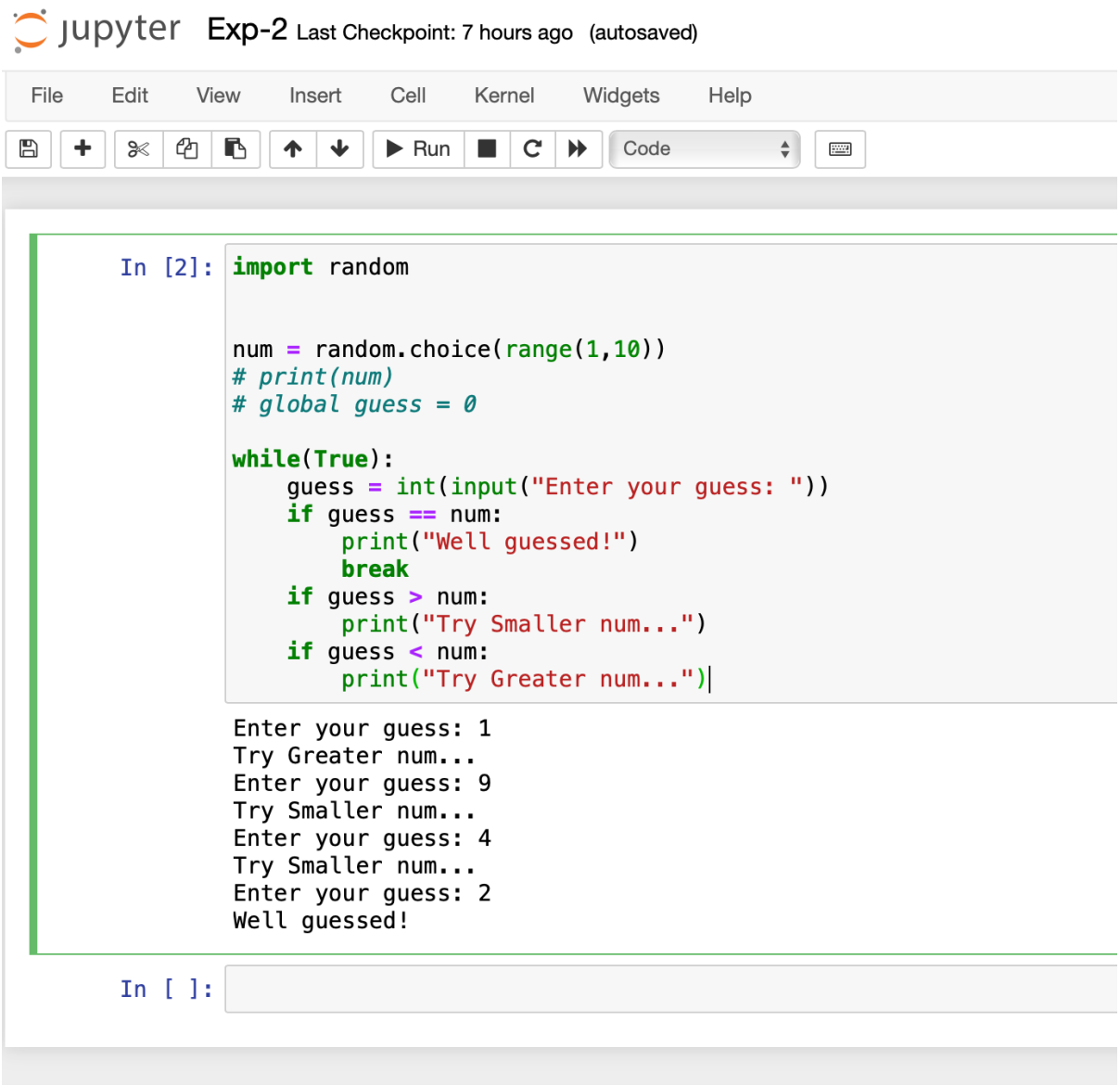
Aim

2. Write a Python program to guess a number between 1 to 9.

Note: User is prompted to enter a guess. If the user guesses wrong then the prompt appears again until the guess is correct, on successful guess, user will get a “Well guessed!” message, and the program will exit.

CODE:

```
import random


num = random.choice(range(1,10))

# print(num)

# global guess = 0

while(True):

    guess = int(input("Enter your guess: "))

    if guess == num:

        print("Well guessed!")

        break

    if guess > num:

        print("Try Smaller num...")

    if guess < num:

        print("Try Greater num...")
```



Result:

Python program to guess the num by the user was completed.

Aim

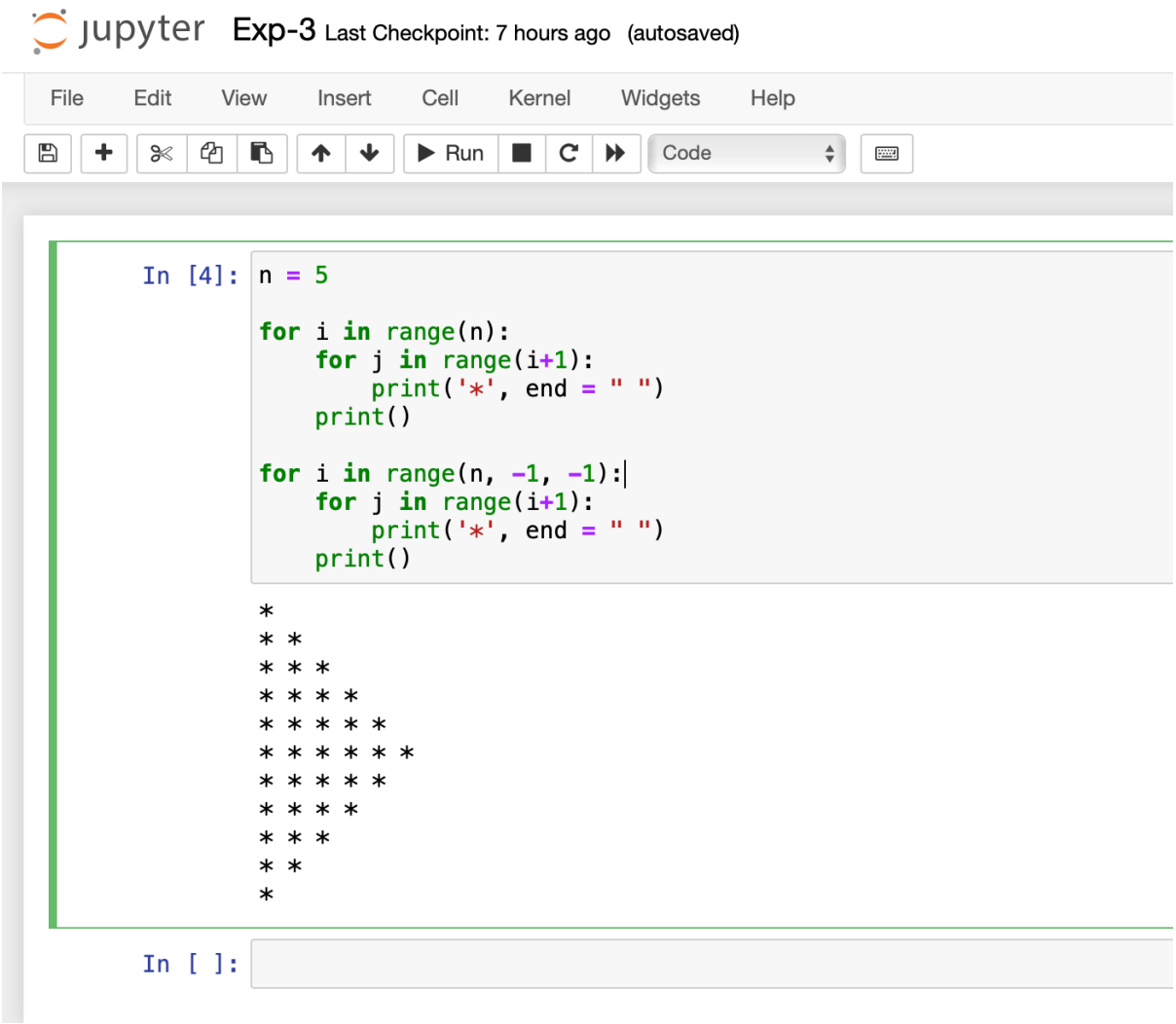
3. Write a Python program to construct the following pattern, using a nested

for loop.

```
*
* *
* * *
* * * *
* * * * *
* * * *
* * *
* *
*
```

CODE:

```
n = 5
for i in range(n):
    for j in range(i+1):
        print('*', end = " ")
    print()
for i in range(n, -1, -1):
    for j in range(i+1):
        print('*', end = " ")
    print()
```



Result:

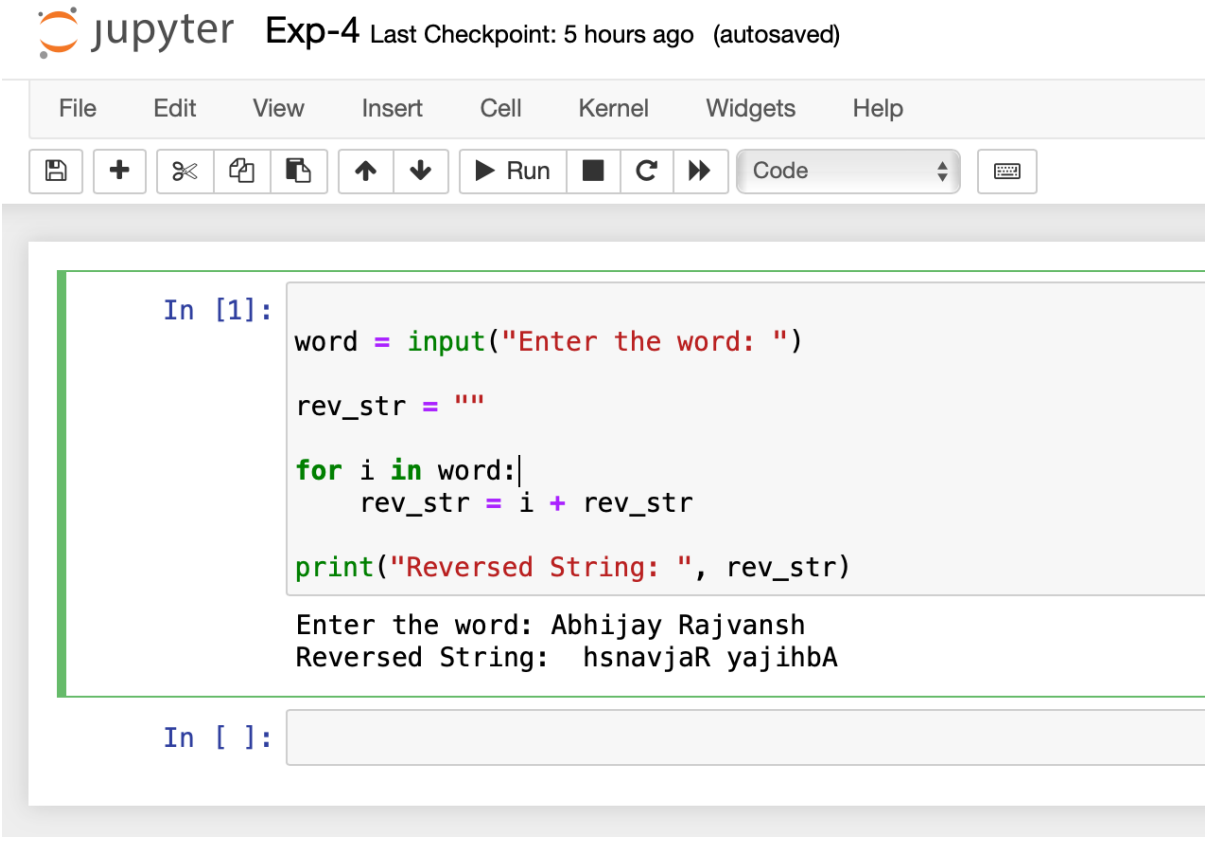
Python program to print the given pattern was completed.

Aim

4. Write a Python program that accepts a word from the user and reverse it. (should not use any functions)

CODE:

```
word = input("Enter the word: ")

rev_str = ""

for i in word:
    rev_str = i + rev_str

print("Reversed String: ", rev_str)
```



Result

Python program that accepts a word from the user and reverse it was completed.

AIM

5. Write a Python program which takes two digits m (row) and n (column) as input and generates a two-dimensional array. The element value in the i-th row and j-th column of the array should be i*j. Note :

i = 0,1.., m-1

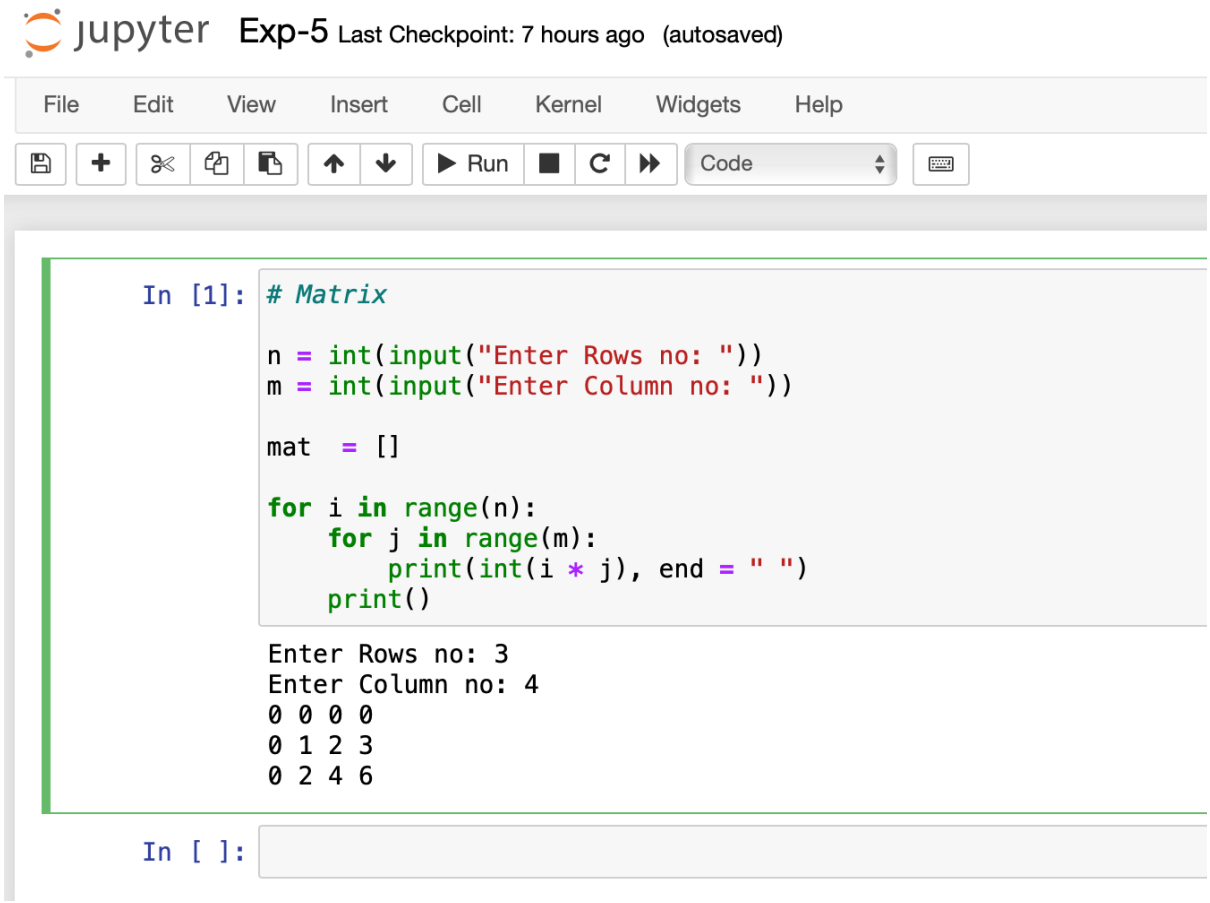
j = 0,1, n-1.

Test Data : Rows = 3, Columns = 4

Expected Result : [[0, 0, 0, 0], [0, 1, 2, 3], [0, 2, 4, 6]]

CODE:

```
# Matrix
n = int(input("Enter Rows no: "))
m = int(input("Enter Column no: "))

mat  = []

for i in range(n):
    for j in range(m):
        print(int(i * j), end = " ")
    print()
```



Result:

Python program which takes two digits m (row) and n (column) as input and generates a two-dimensional array was completed.

Aim

6. Write a Python program that accepts a string and calculate the number of digits and letters.

Sample Data : SRMIST 2022

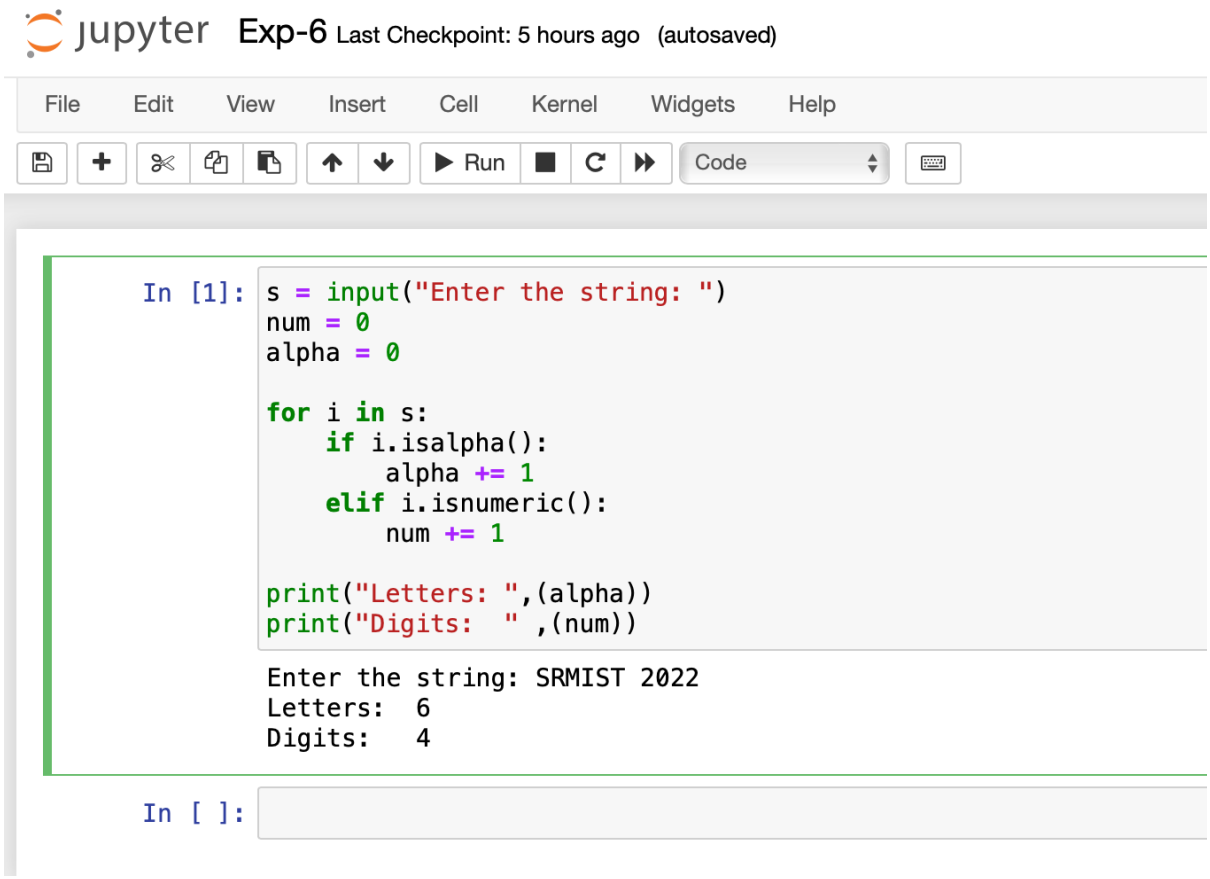
Expected Output :

Letters 6

Digits 4

CODE:

```
s = input("Enter the string: ")

num = 0

alpha = 0


for i in s:

   if i.isalpha():

      alpha += 1

   elif i.isnumeric():

      num += 1


print("Letters: ",(alpha))

print("Digits:  " ,(num))
```



Result:

Python program that accepts a string and calculate the number of digits and letters was completed.

AIM

7. Write a Python program to check the validity of password input by users.

● At least 1 letter between [a-z] and 1 letter between [A-Z].

● At least 1 number between [0-9].

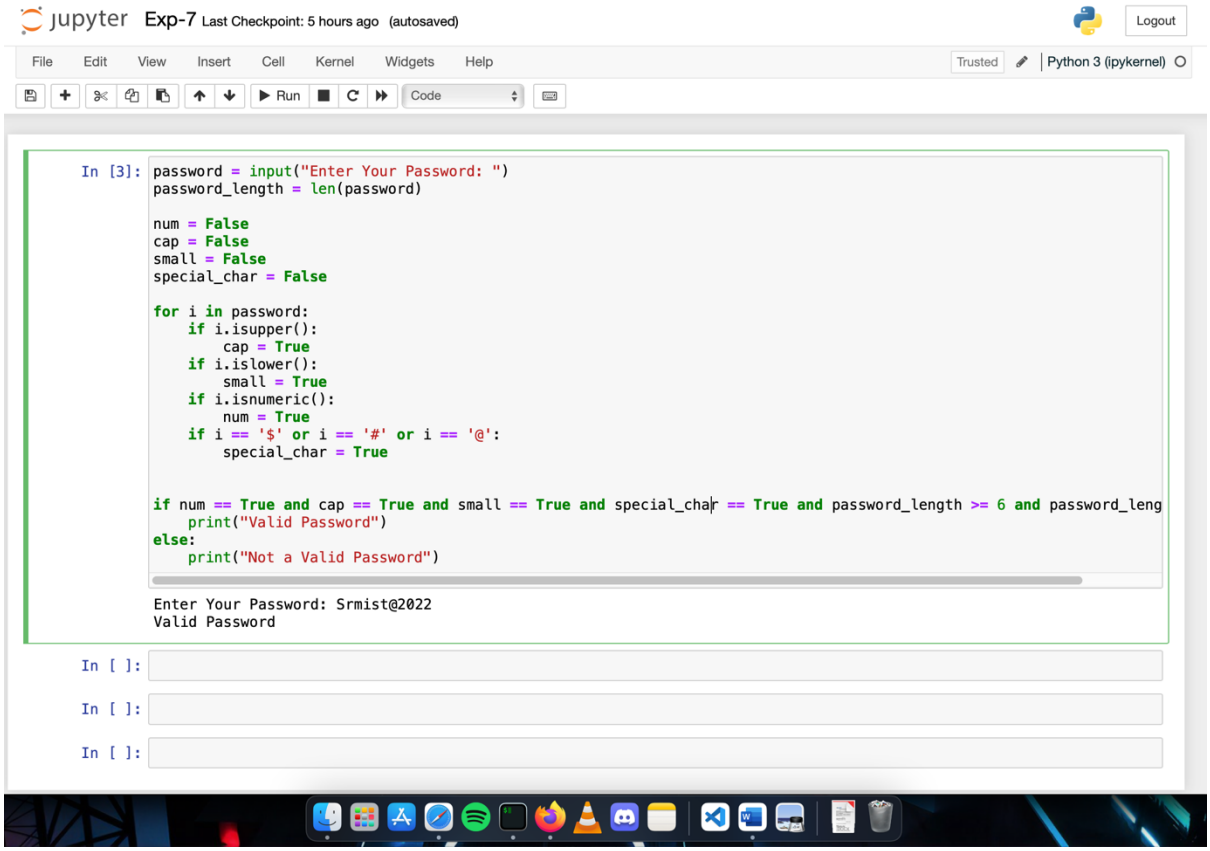
● At least 1 character from [$#@].

● Minimum length 6 characters.

● Maximum length 16 characters.

CODE:

```
password = input("Enter Your Password: ")

password_length = len(password)

num = False

cap = False

small = False

special_char = False

for i in password:

    if i.isupper():

        cap = True

    if i.islower():

        small = True

    if i.isnumeric():

        num = True

    if i == '$' or i == '#' or i == '@':

        special_char = True




if num == True and cap == True and small == True and special_char == True and password_length >= 6 and
password_length <= 16:

    print("Valid Password")

else:

    print("Not a Valid Password")
```



Result

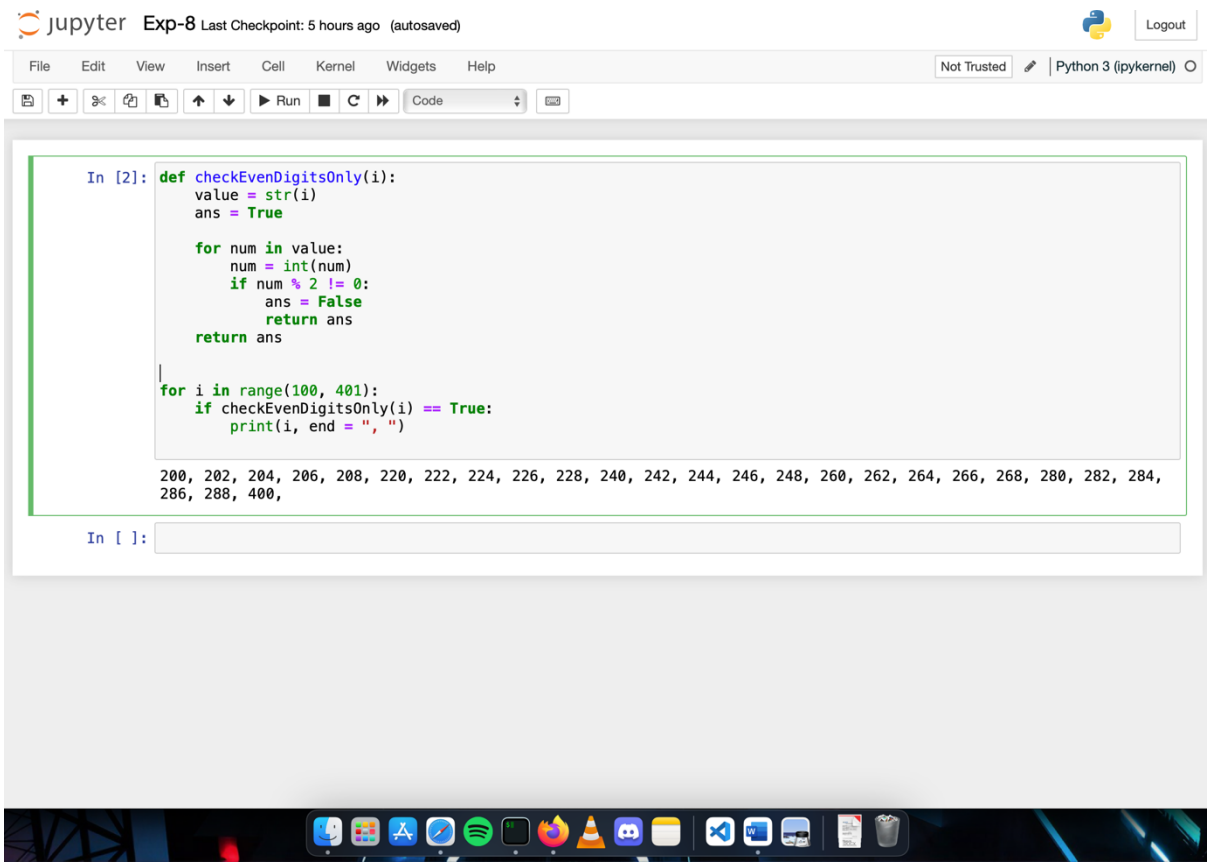
Python program to check the validity of password input by users was completed.

AIM

8. Write a Python program to find numbers between 100 and 400 (both included) where each digit of a number is an even number. The numbers obtained should be printed in a comma-separated sequence.

Code:

```
def checkEvenDigitsOnly(i):
    value = str(i)
    ans = True

    for num in value:
        num = int(num)
        if num % 2 != 0:
            ans = False
            return ans
    return ans


for i in range(100, 401):
    if checkEvenDigitsOnly(i) == True:
        print(i, end = ", ")
```



Result

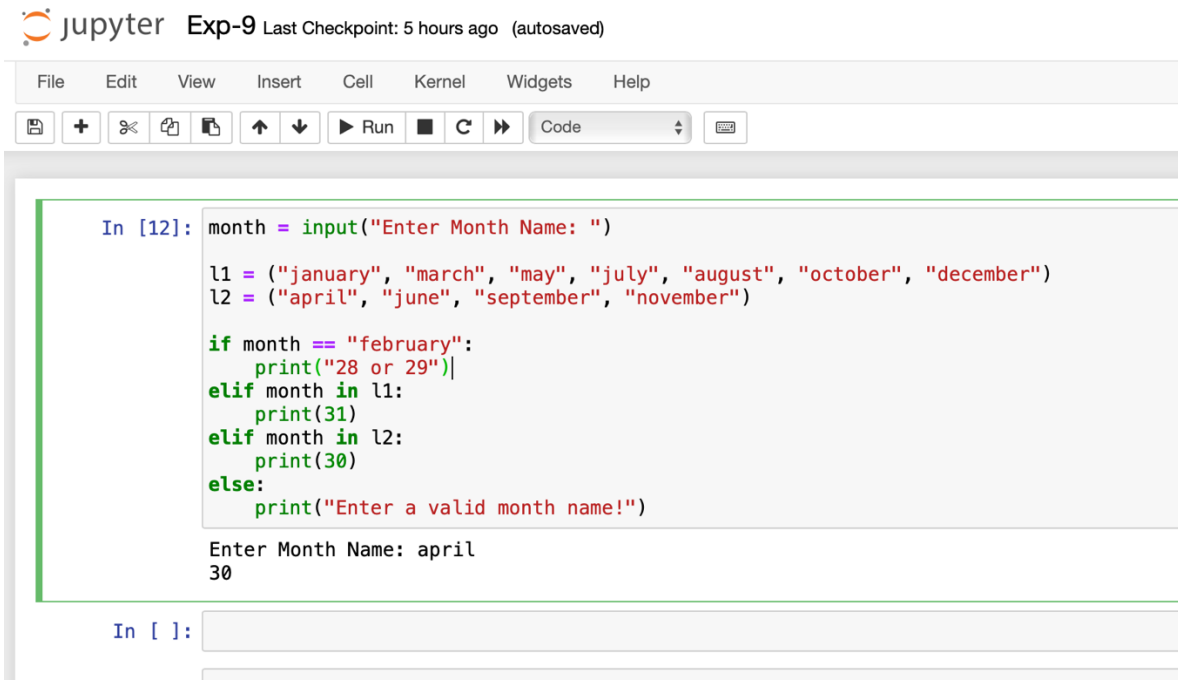
Python program to find numbers between 100 and 400 (both included) where each digit of a number is an even number was completed.

AIM

9. Write a Python program to convert month name to a number of days.

CODE:

```
month = input("Enter Month Name: ")


l1 = ("january", "march", "may", "july", "august", "october", "december")

l2 = ("april", "june", "september", "november")


if month == "february":
    print("28 or 29")
elif month in l1:
    print(31)
elif month in l2:
    print(30)
else:
    print("Enter a valid month name!")
```



Result

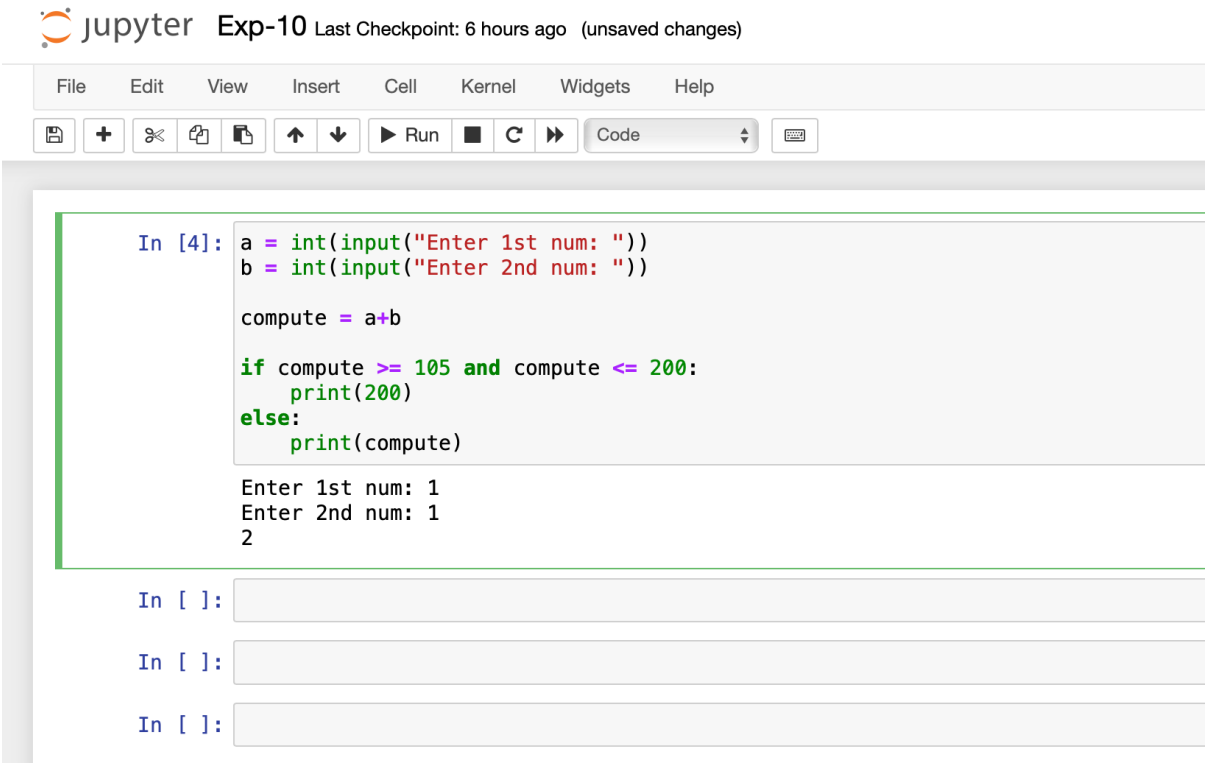
Python program to convert month name to a number of days was completed.

AIM

10. Write a Python program to sum of two given integers. However, if the sum is between 105 to 200 it will return 200.

CODE:

```
a = int(input("Enter 1st num: "))

b = int(input("Enter 2nd num: "))


compute = a+b


if compute >= 105 and compute <= 200:

    print(200)

else:

    print(compute)
```



Result

Python program to sum of two given integers. However, if the sum is between 105 to 200 it will return 200 was completed.

AIM

11. Write a Python program to construct the following pattern, using a

nested loop number.

Expected Output:

999999999

88888888

7777777

666666

55555

4444

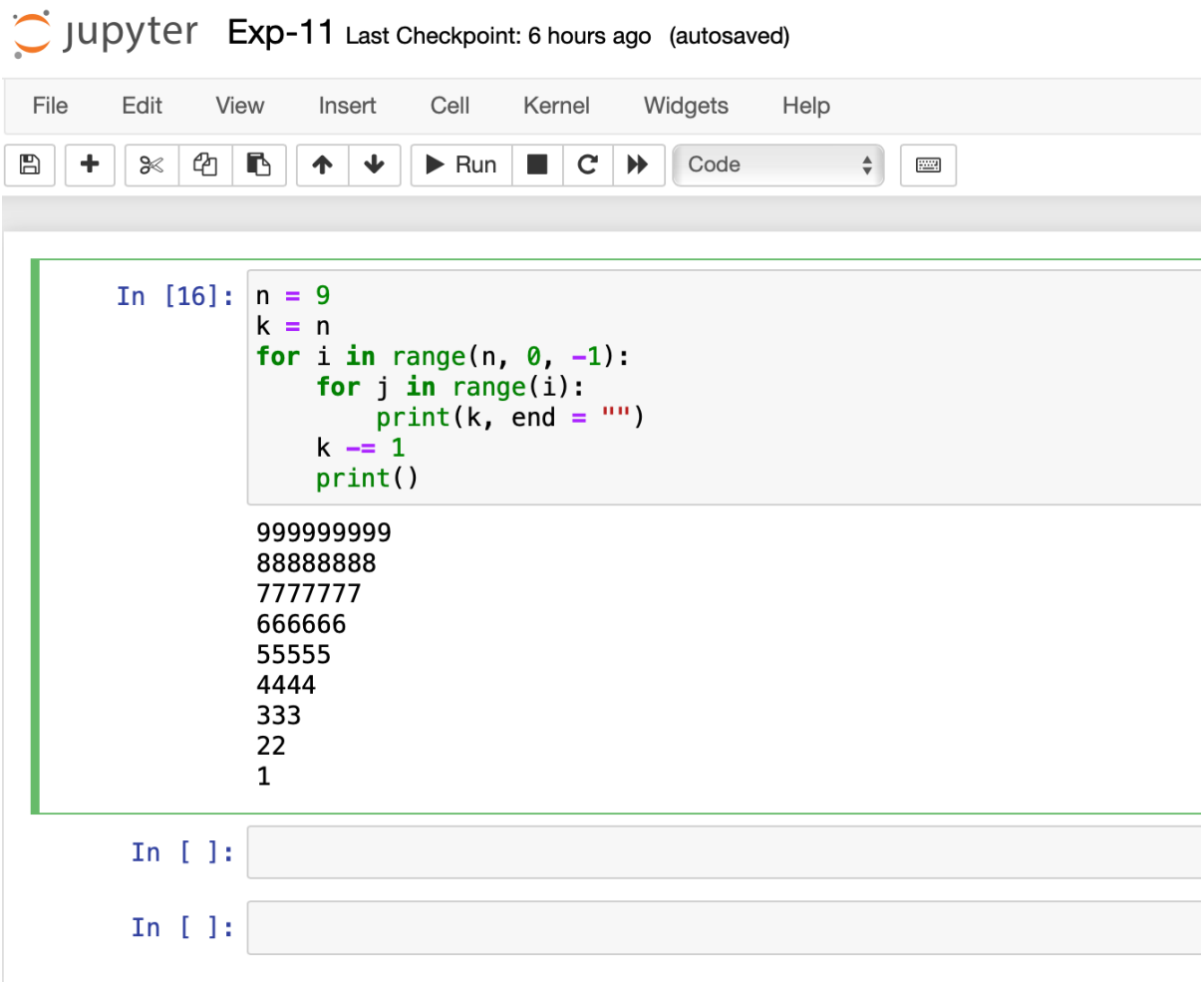
333

22

1

CODE:

```
n = 9
k = n
for i in range(n, 0, -1):
    for j in range(i):
        print(k, end = "")
    k -= 1
    print()
```



Result

Python program to construct the following pattern, using a nested loop number was completed.

AIM

12. Write a Python program to create a histogram from a given list of integers.

CODE:

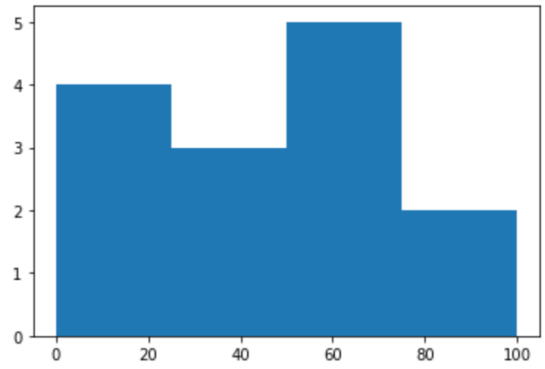
```
from matplotlib import pyplot as plt
import numpy as np
#sample list of integers
a = np.array([87, 5, 43, 56, 73, 55, 54, 11, 20, 51, 5, 79, 31,27])
fig, ax = plt.subplots(figsize =(6, 4))
ax.hist(a, bins = [0, 25, 50, 75, 100])
plt.show()
```

jupyter Exp-12 Last Checkpoint: 2 hours ago (autosaved)

File Edit View Insert Cell Kernel Widgets Help

Run Code

In [7]:
```
from matplotlib import pyplot as plt
import numpy as np


#sample list of integers
a = np.array([87, 5, 43, 56, 73, 55, 54, 11, 20, 51, 5, 79, 31,27])

fig, ax = plt.subplots(figsize =(6, 4))
ax.hist(a, bins = [0, 25, 50, 75, 100])

plt.show()
```



In [ ]:

Result

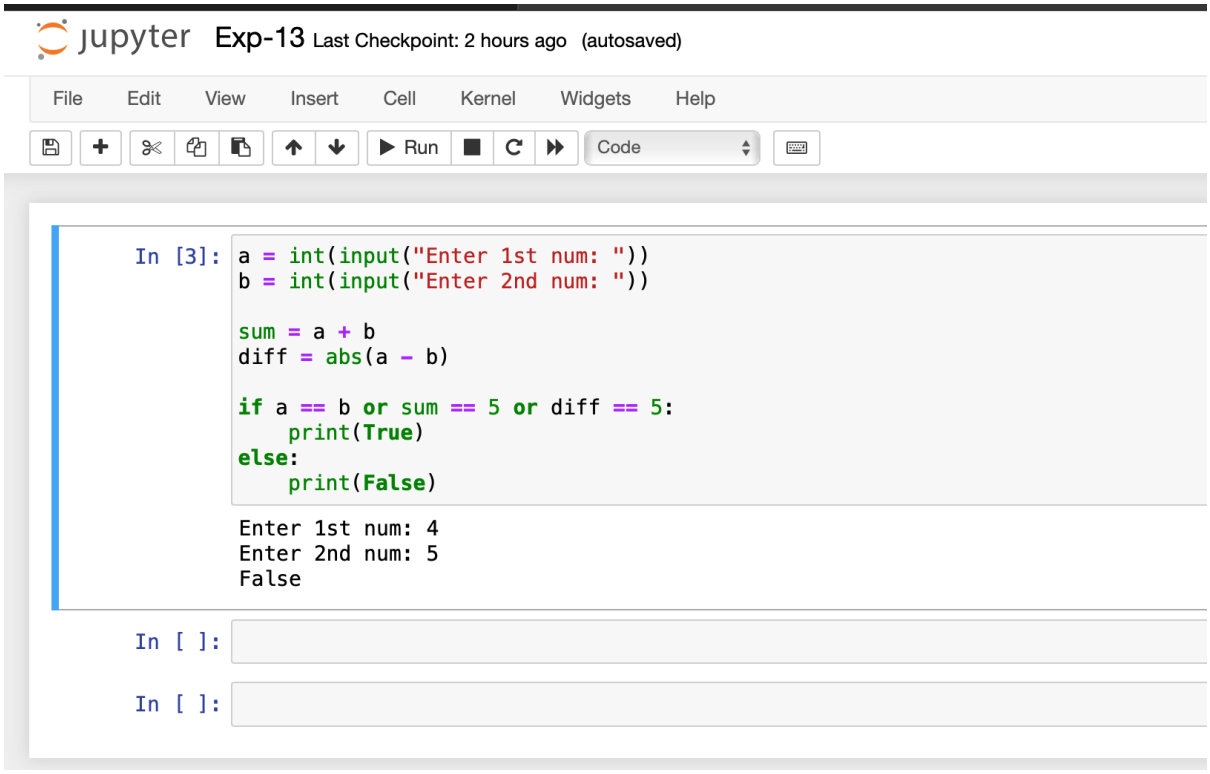
Python program to create a histogram from a given list of integers was completed.

AIM

13. Write a Python program that will return true if the two given integer values are equal or their sum or difference is 5.

CODE:

```
a = int(input("Enter 1st num: "))
b = int(input("Enter 2nd num: "))

sum = a + b
diff = abs(a - b)

if a == b or sum == 5 or diff == 5:
    print(True)
else:
    print(False)
```



Result

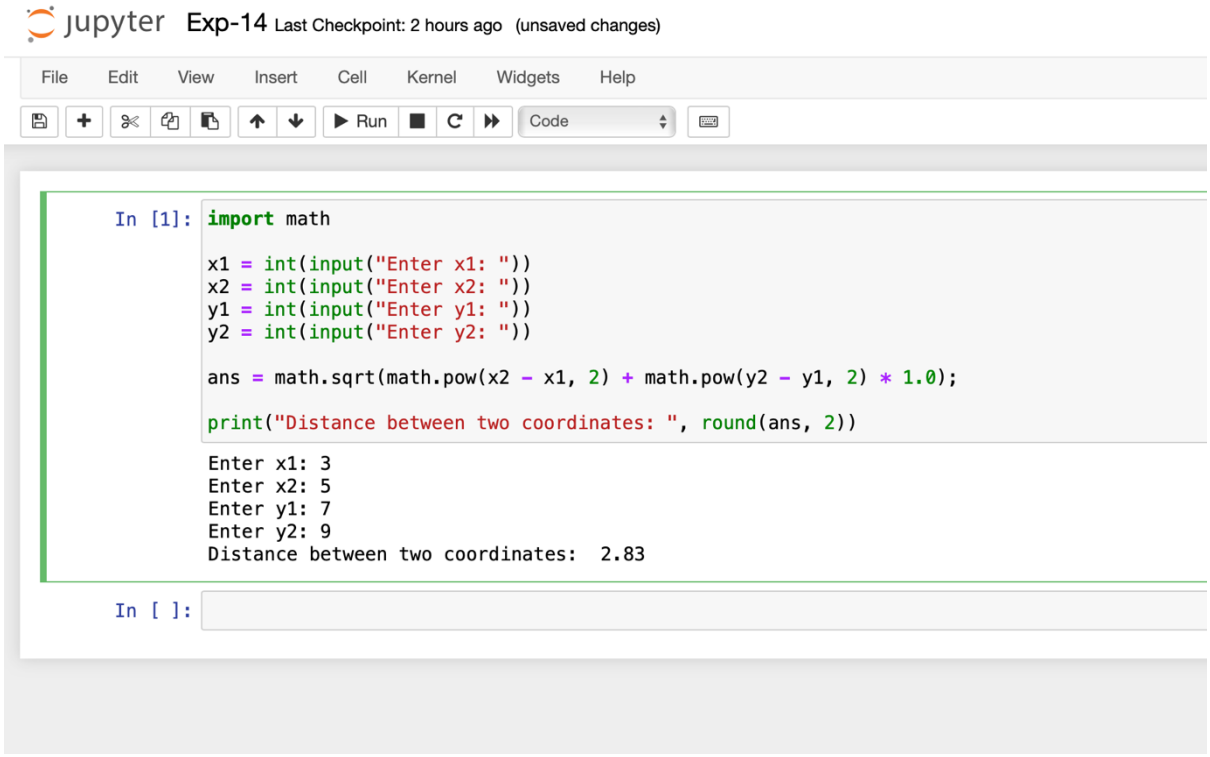
Python program that will return true if the two given integer values are equal or their sum or difference is 5 was completed.

AIM

14. Write a Python program to compute the distance between the points (x1, y1) and (x2, y2).

CODE:

```
import math


x1 = int(input("Enter x1: "))

x2 = int(input("Enter x2: "))

y1 = int(input("Enter y1: "))

y2 = int(input("Enter y2: "))


ans = math.sqrt(math.pow(x2 - x1, 2) + math.pow(y2 - y1, 2) * 1.0);


print("Distance between two coordinates: ", round(ans, 2))
```



Result

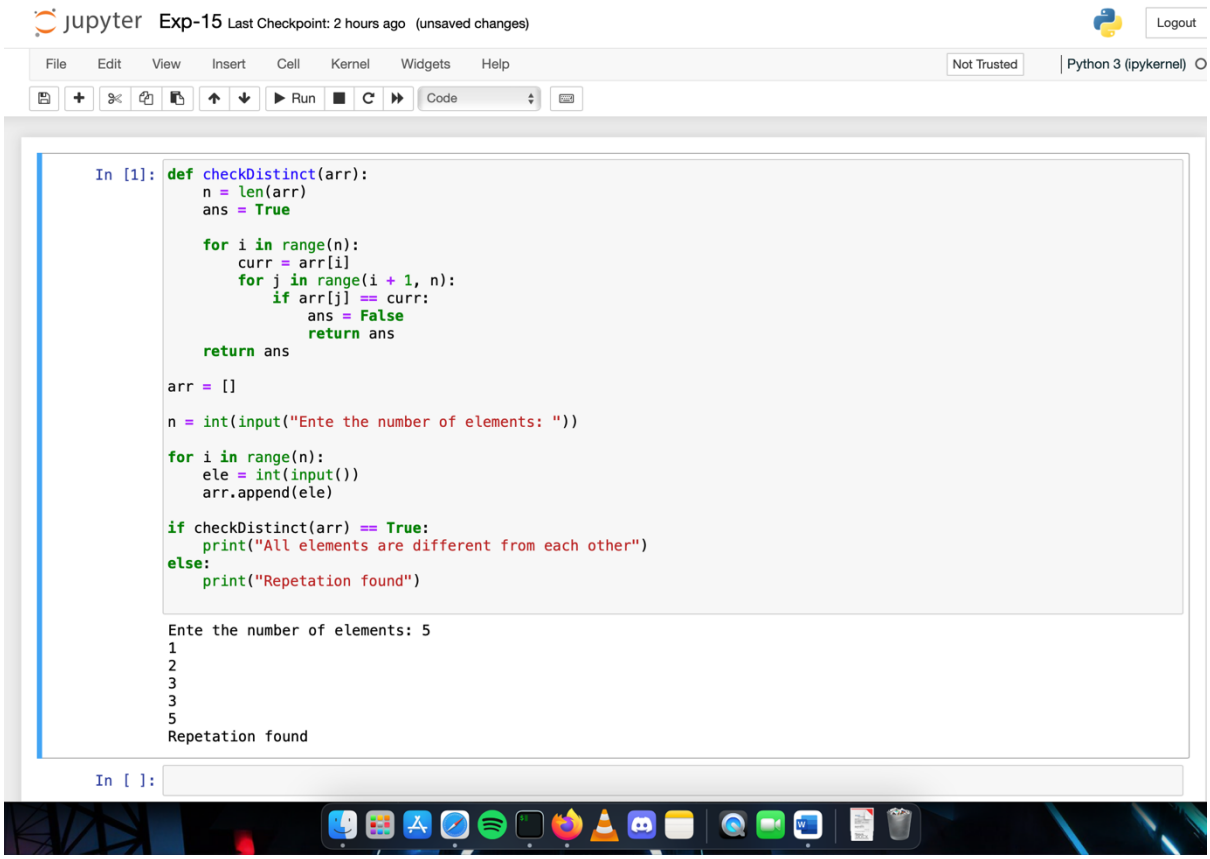
Python program to compute the distance between the points (x1, y1) and (x2, y2) was completed

AIM

15. Function that takes a sequence of numbers and determines whether all are different from each other.

CODE:

```
def checkDistinct(arr):
    n = len(arr)
    ans = True

    for i in range(n):
        curr = arr[i]
        for j in range(i + 1, n):
            if arr[j] == curr:
                ans = False
                return ans
    return ans

arr = []

n = int(input("Ente the number of elements: "))

for i in range(n):
    ele = int(input())
    arr.append(ele)

if checkDistinct(arr) == True:
    print("All elements are different from each other")
else:
    print("Repetation found")
```



Result

Python program to create a function that takes a sequence of numbers and determines whether all are different from each other was completed.

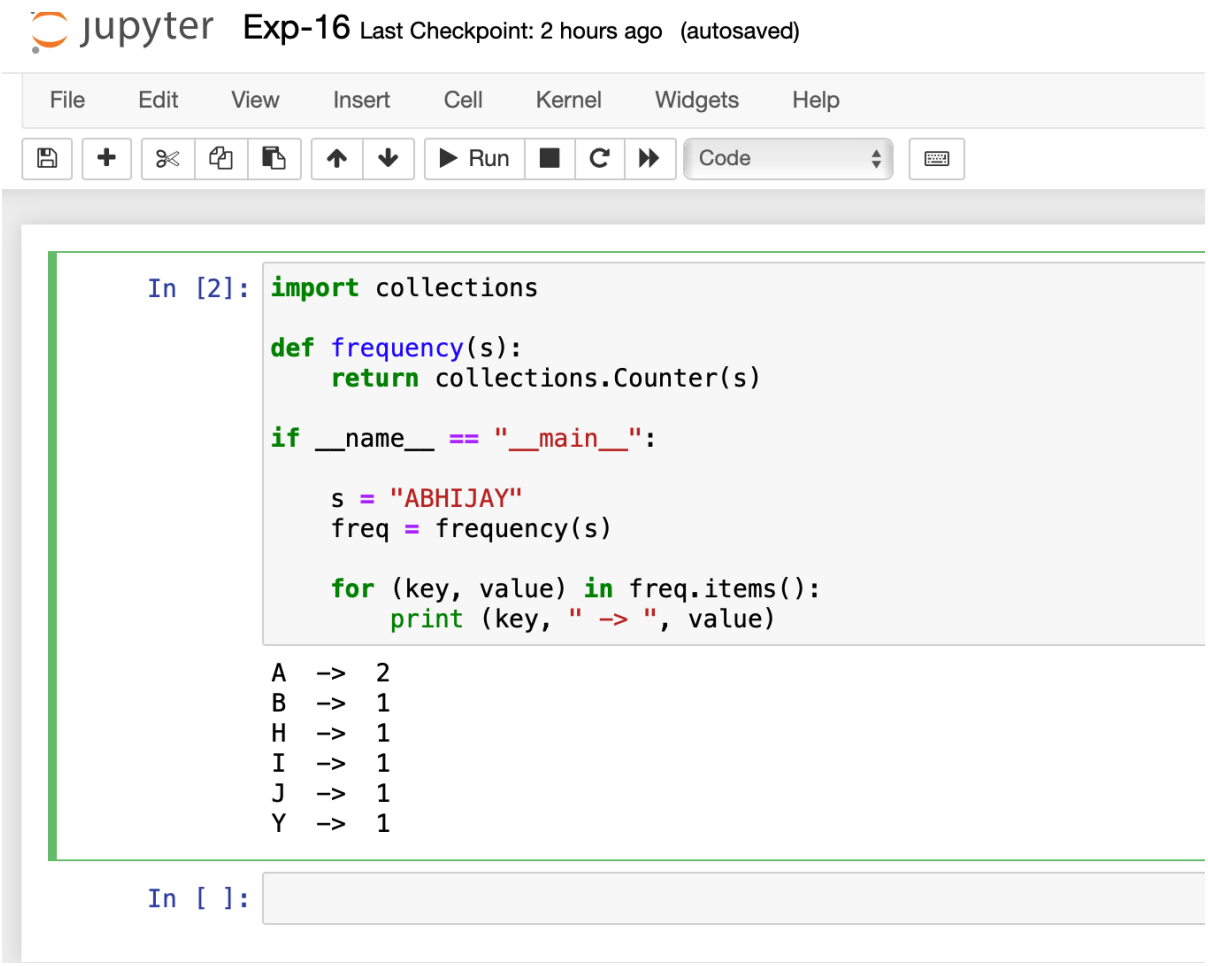
AIM

16. Write a Python program to count the number of each character of a

given text : -

CODE:

```
import collections


def frequency(s):


    return collections.Counter(s)


if __name__ == "__main__":


    s = "ABHIJAY"


    freq = frequency(s)


    for (key, value) in freq.items():

        print (key, " -> ", value)
```



Result

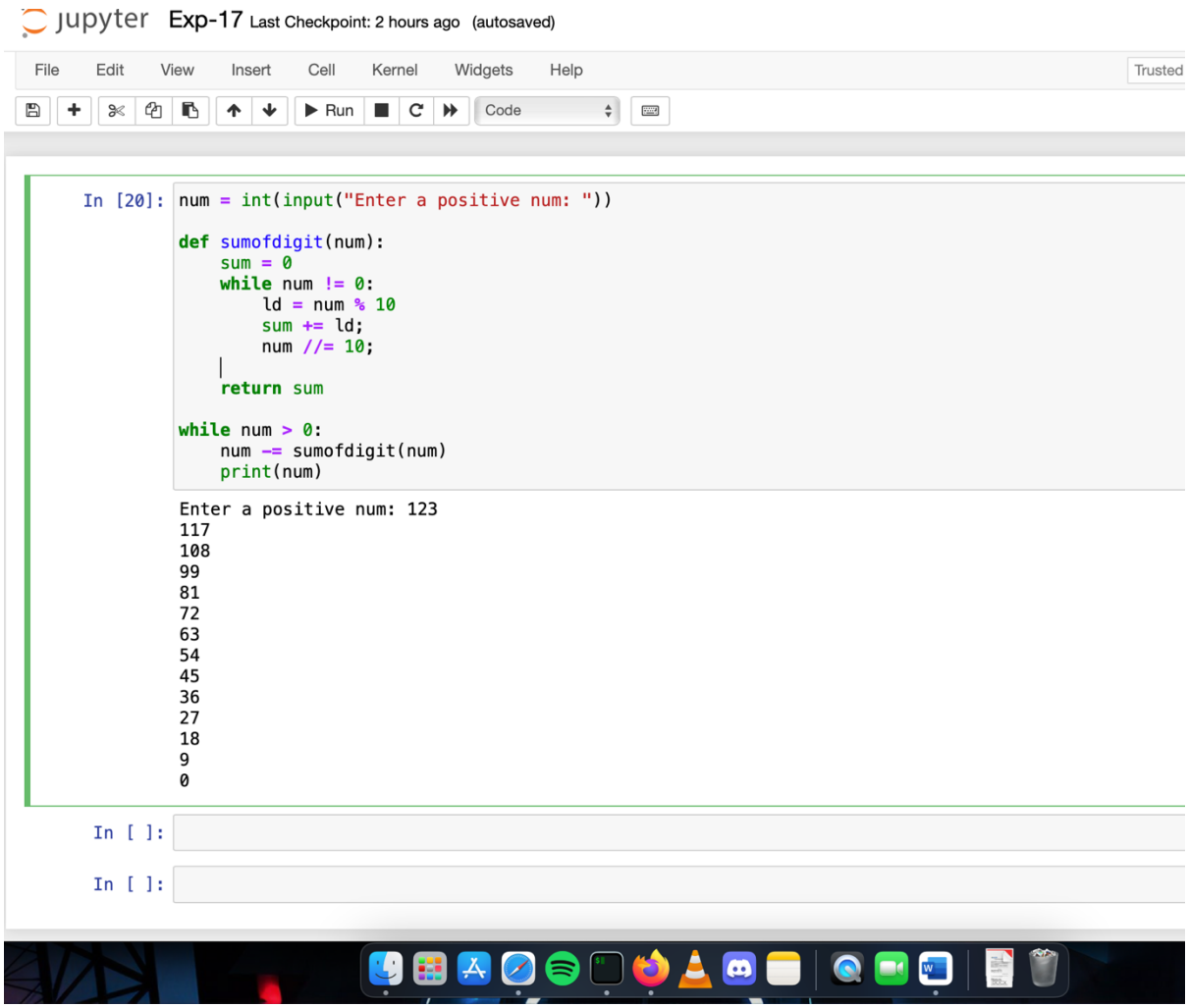
Python program to count the number of each character of a given text was completed.

AIM

17. Write a Python program that accept a positive number and subtract from this number the sum of its digits and so on. Continues this operation until the number is positive.

CODE:

```
num = int(input("Enter a positive num: "))


def sumofdigit(num):
    sum = 0
    while num != 0:
        ld = num % 10
        sum += ld;
        num //= 10;


    return sum


while num > 0:
    num -= sumofdigit(num)
   print(num)
```



Result

Python program that accept a positive number and subtract from this number the sum of its digits and so on. Continues this operation until the number is positive was completed.

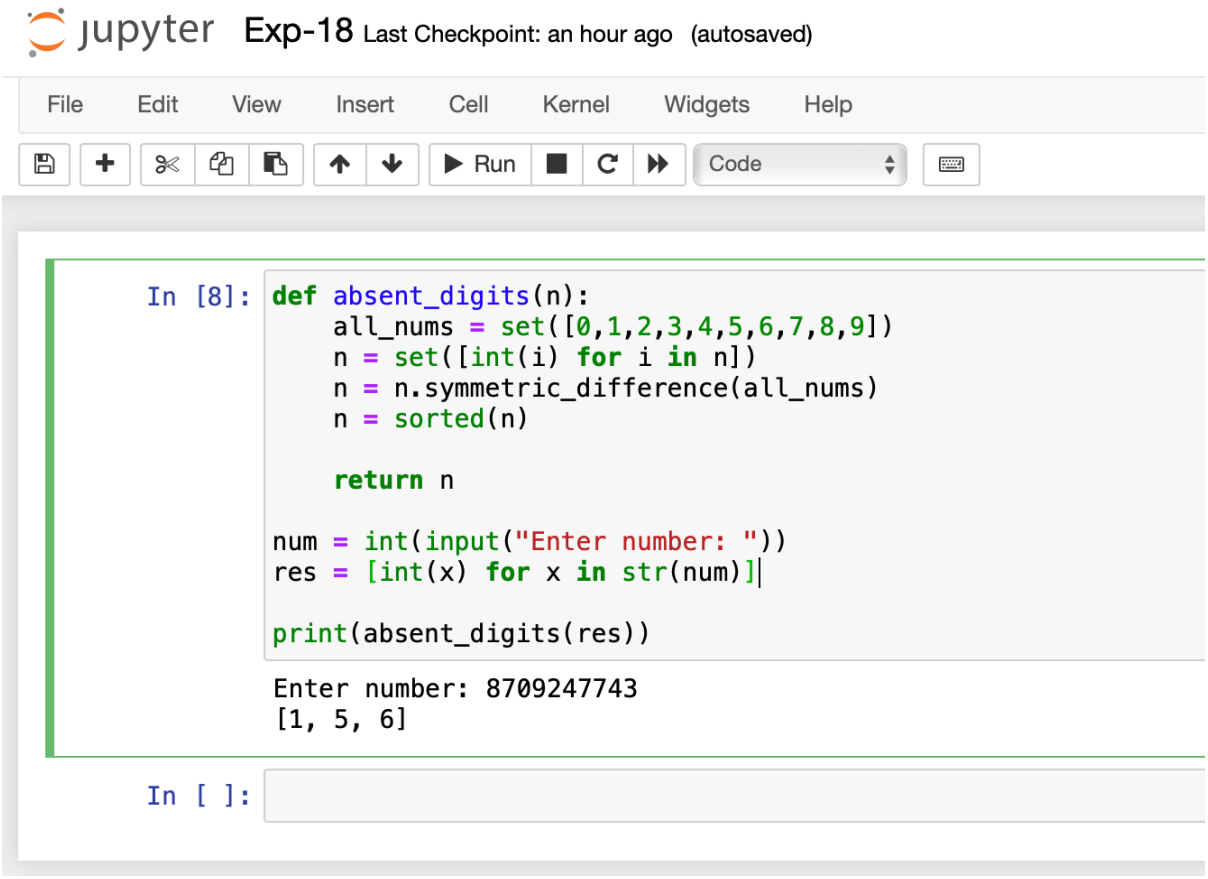
AIM

18. Write a Python program to find the digits which are absent in a

given mobile number.

CODE:

```
def absent_digits(n):
    all_nums = set([0,1,2,3,4,5,6,7,8,9])
    n = set([int(i) for i in n])
    n = n.symmetric_difference(all_nums)
    n = sorted(n)

    return n

num = int(input("Enter number: "))
res = [int(x) for x in str(num)]

print(absent_digits(res))
```



Result

Python program to find the digits which are absent in a given mobile number was completed.

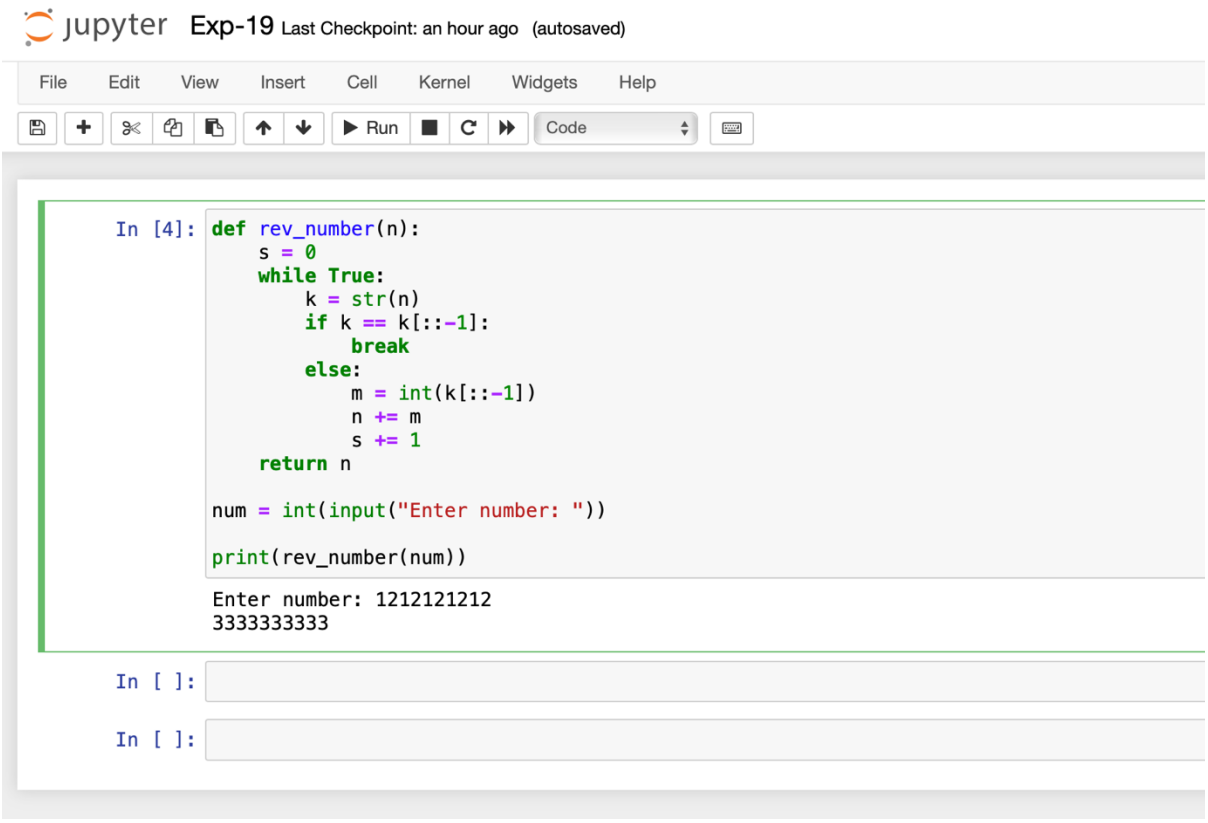
AIM

19. Write a Python program to reverse the digits of a given number and

add it to the original, If the sum is not a palindrome repeat this procedure

CODE:

```
def rev_number(n):
    s = 0
    while True:
        k = str(n)
        if k == k[::-1]:
            break
        else:
            m = int(k[::-1])
            n += m
            s += 1
    return n

num = int(input("Enter number: "))

print(rev_number(num))
```



Result

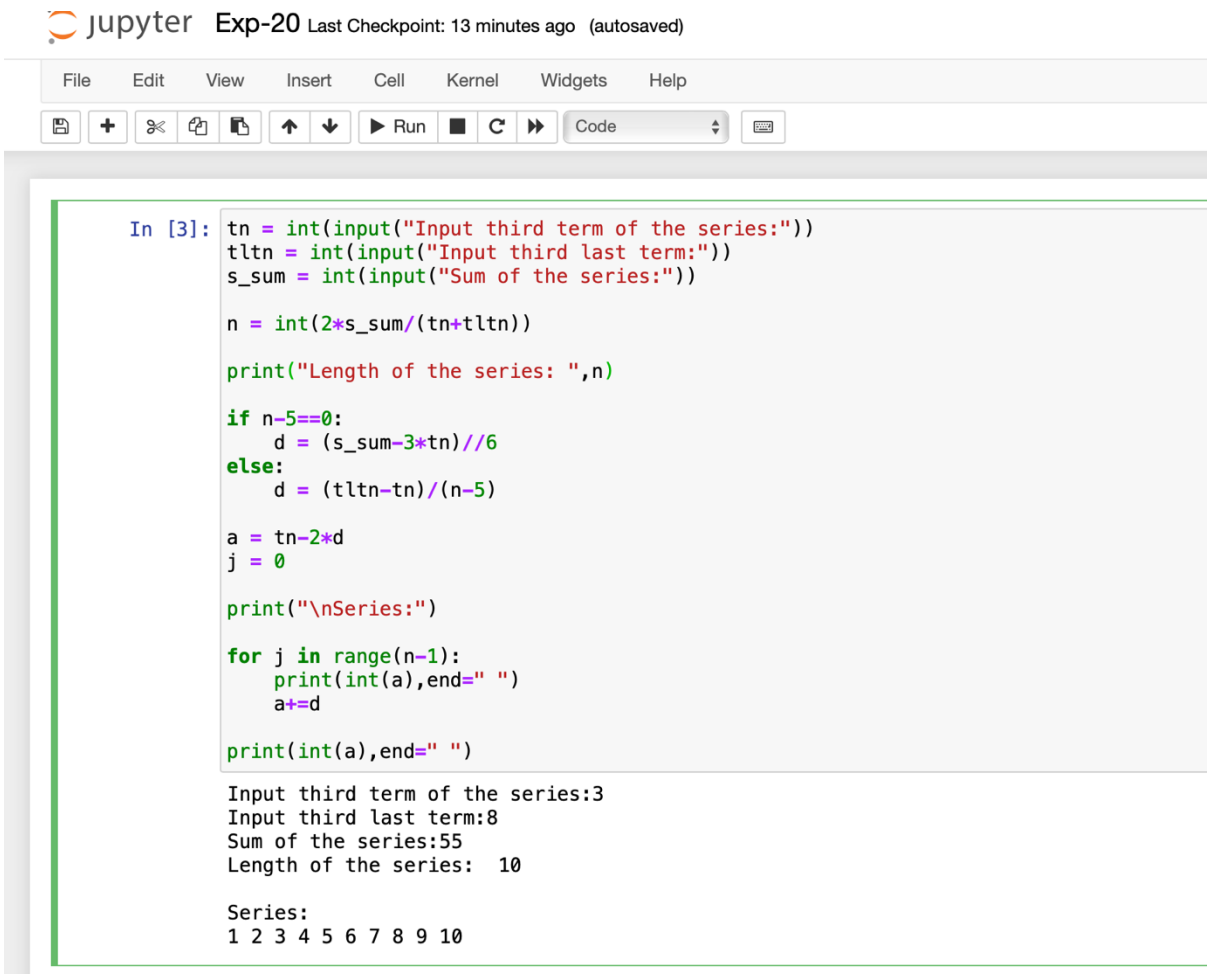
Python program to reverse the digits of a given number and add it to the original, If the sum is not a palindrome repeat this procedure was completed.

Aim

20. Write a Python program to print the length of the series and the series from the given 3rd term, 3rd last term and the sum of a series.

CODE:

```
tn = int(input("Input third term of the series:"))
tltn = int(input("Input third last term:"))
s_sum = int(input("Sum of the series:"))

n = int(2*s_sum/(tn+tltn))

print("Length of the series: ",n)

if n-5==0:
    d = (s_sum-3*tn)//6
else:
    d = (tltn-tn)/(n-5)

a = tn-2*d
j = 0

print("\nSeries:")

for j in range(n-1):
    print(int(a),end=" ")
    a+=d

print(int(a),end=" ")
```



Result:

Python program to print the length of the series and the series from the given 3rd term, 3rd last term and the sum of a series was completed.