

# Assignment 1 – ICP\_1\_Spring24

Student id: 700763475

Ucm id :sxn34750

Name :Sai Nalla

## Technical Document

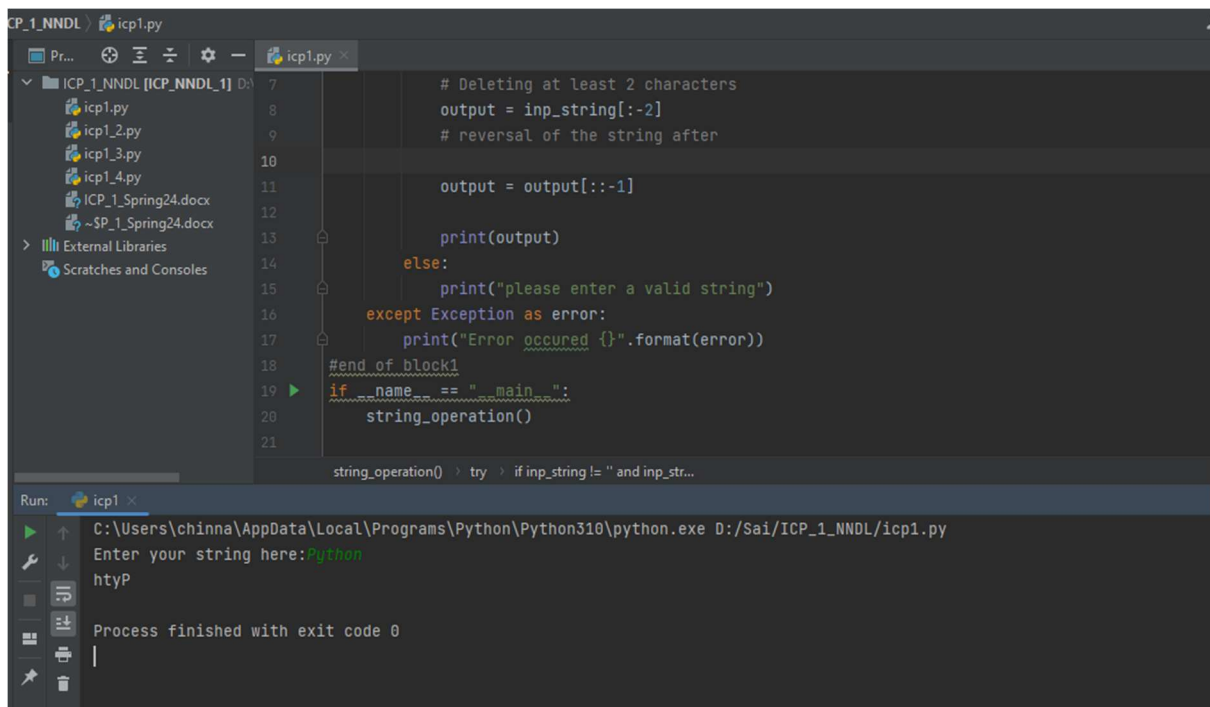
---

1)–Write a python program take the Input string “Python” as a list of characters from console, delete at least 2 characters, reverse the resultant string and print it.

```
icp1.py x
1  #Author : Sai Nalla
2  #start
3  def string_operation():
4      try:
5          inp_string = str(input("Enter your string here:"))
6          if inp_string != '' and inp_string is not None and inp_string.isspace() != True and inp_string.isnumeric() != True:
7              # Deleting at least 2 characters
8              output = inp_string[:-2]
9              # reversal of the string after
10
11              output = output[::-1]
12
13              print(output)
14          else:
15              print("please enter a valid string")
16      except Exception as error:
17          print("Error occurred {}".format(error))
18  #end of block1
19  if __name__ == "__main__":
20      string_operation()
21
22
```

## Output

=====

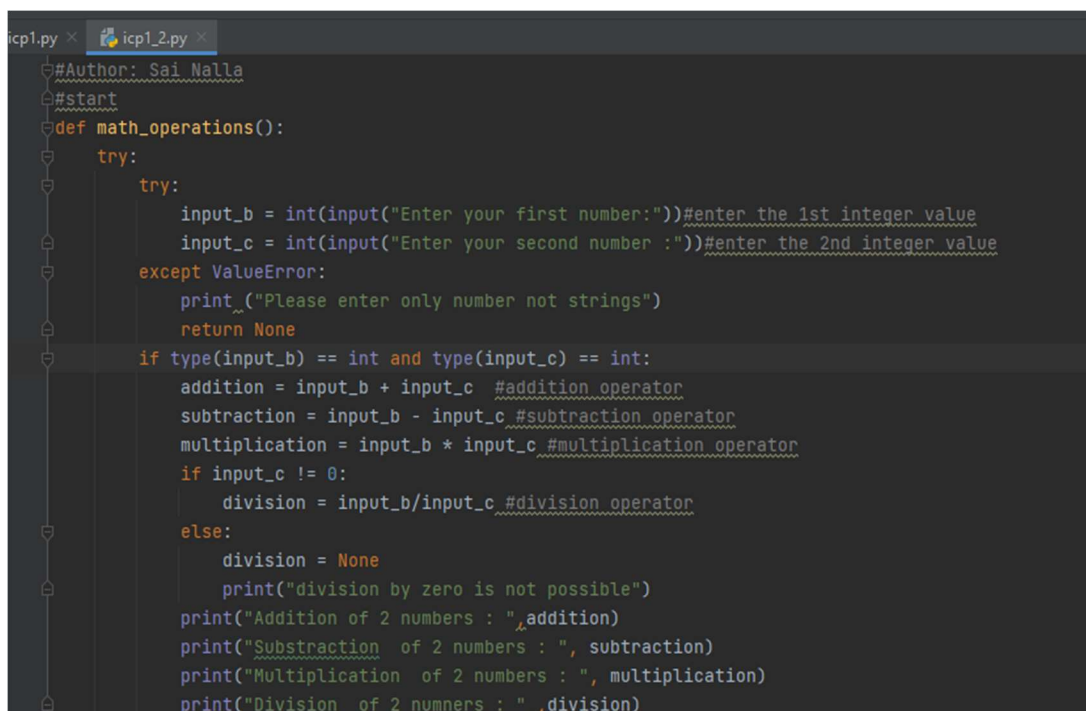


The screenshot shows a Python IDE with a file explorer on the left displaying a project named 'ICP\_1\_NNDL'. The main editor window shows the code for 'icp1.py'. The code defines a function 'string\_operation()' that takes an input string and performs two operations: deleting the last two characters and reversing the string. It includes error handling for invalid inputs. The script is executed, and the output shows the input 'Python' being transformed into 'htyP'.

```
7 # Deleting at least 2 characters
8 output = inp_string[::-2]
9 # reversal of the string after
10
11 output = output[::-1]
12
13 print(output)
14 else:
15     print("please enter a valid string")
16 except Exception as error:
17     print("Error occured {}".format(error))
18 #end of block1
19 if __name__ == "__main__":
20     string_operation()
21
```

Run: icp1 x  
C:\Users\chinna\AppData\Local\Programs\Python\Python310\python.exe D:/Sai/ICP\_1\_NNDL/icp1.py  
Enter your string here: Python  
htyP  
Process finished with exit code 0

2) Take two numbers from user and perform at least 4 arithmetic operations on them



The screenshot shows a Python IDE with a file explorer on the left displaying a project named 'icp1'. The main editor window shows the code for 'icp1\_2.py'. The code defines a function 'math\_operations()' that takes two integers as input and performs addition, subtraction, multiplication, and division. It includes error handling for non-integer inputs and division by zero. The script is executed, and the output shows the input numbers 10 and 2 being transformed into 'Addition of 2 numbers : 12', 'Subtraction of 2 numbers : 8', 'Multiplication of 2 numbers : 20', and 'Division of 2 numners : 5'.

```
#Author: Sai Nalla
#start
def math_operations():
    try:
        try:
            input_b = int(input("Enter your first number:"))#enter the 1st integer value
            input_c = int(input("Enter your second number :"))#enter the 2nd integer value
        except ValueError:
            print("Please enter only number not strings")
            return None
        if type(input_b) == int and type(input_c) == int:
            addition = input_b + input_c #addition operator
            subtraction = input_b - input_c #subtraction operator
            multiplication = input_b * input_c #multiplication operator
            if input_c != 0:
                division = input_b/input_c #division operator
            else:
                division = None
                print("division by zero is not possible")
            print("Addition of 2 numbers : ", addition)
            print("Subtraction of 2 numbers : ", subtraction)
            print("Multiplication of 2 numbers : ", multiplication)
            print("Division of 2 numners : ", division)
```

```

    else:
        print("please enter a valid number")
    except Exception as error:
        print("Error occured {}".format(error))
#end

if __name__ == "__main__":
    math_operations()

```

## Output

The screenshot displays a Python IDE with a file explorer on the left showing a project named 'ICP\_1\_NNDL'. The main editor window shows the code for 'icp1\_2.py'. The code defines a function 'math\_operations()' that takes two inputs, 'input\_b' and 'input\_c', and performs four operations: addition, subtraction, multiplication, and division. It includes error handling for division by zero. The 'Run' console at the bottom shows the execution of the script, where the user enters '12' for the first number and '4' for the second number. The output displays the results of these operations: addition (16), subtraction (8), multiplication (48), and division (3.0). The process finished with exit code 0.

```

12 addition = input_b + input_c #addition operator
13 subtraction = input_b - input_c #subtraction operator
14 multiplication = input_b * input_c #multiplication operator
15 if input_c != 0:
16     division = input_b/input_c #division operator
17 else:
18     division = None
19     print("division by zero is not possible")
20 print("Addition of 2 numbers : ", addition)
21 print("Substraction of 2 numbers : ", subtraction)
22 print("Multiplication of 2 numbers : ", multiplication)
23 print("Division of 2 numners : ", division)
24 else:
25     print("please enter a valid number")

```

Run: icp1\_2

```

C:\Users\chinna\AppData\Local\Programs\Python\Python310\python.exe D:/Sai/ICP_1_NNDL/icp1_2.py
Enter your first number:12
Enter your second number :4
Addition of 2 numbers : 16
Substraction of 2 numbers : 8
Multiplication of 2 numbers : 48
Division of 2 numners : 3.0

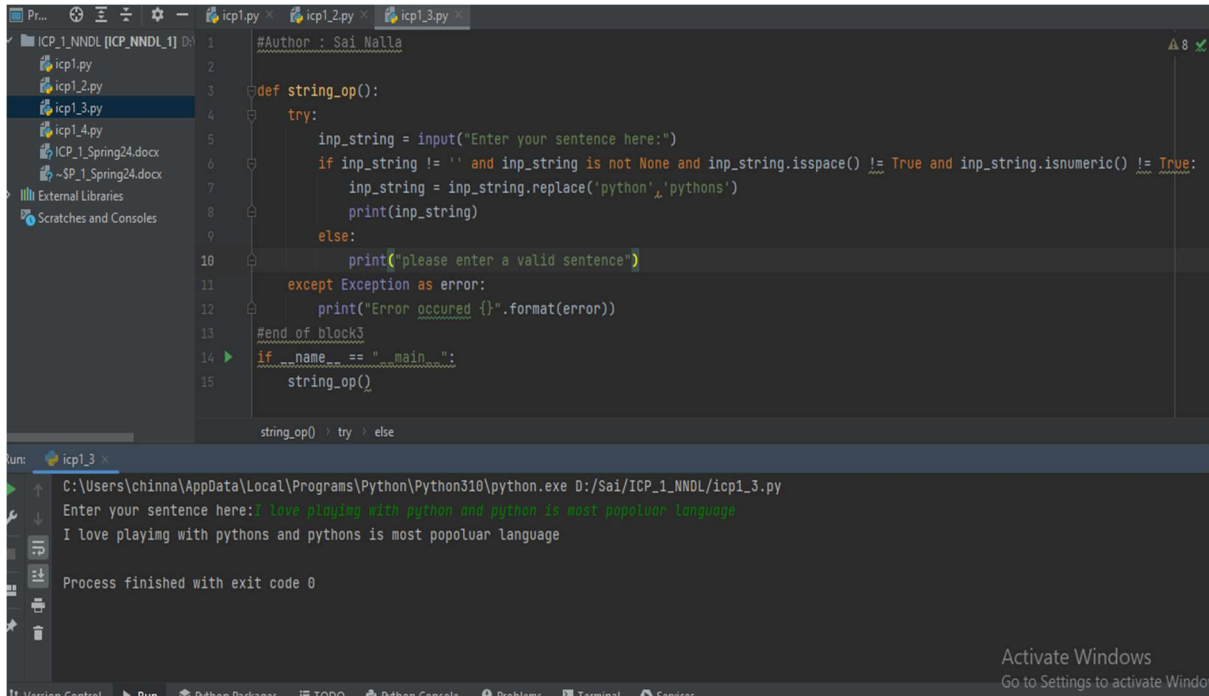
Process finished with exit code 0

```

3. Write a program that accepts a sentence and replace each occurrence of 'python' with 'pythons'.

```
icp1.py × icp1_2.py × icp1_3.py ×
1 #Author : Sai Nalla
2
3 def string_op():
4     try:
5         inp_string = input("Enter your sentence here:")
6         if inp_string != '' and inp_string is not None and inp_string.isspace() != True and inp_string.isnumeric() != True:
7             inp_string = inp_string.replace('python','pythons')
8             print(inp_string)
9         else:
10            print("please enter a valid sentence")
11    except Exception as error:
12        print("Error occured {}".format(error))
13    #end of block3
14    if __name__ == "__main__":
15        string_op()
```

Output



```
Pr... icp1.py × icp1_2.py × icp1_3.py ×
ICP_1_NNDL [ICP_NNDL_1] D:\
icp1.py
icp1_2.py
icp1_3.py
icp1_4.py
ICP_1_Spring24.docx
~SP_1_Spring24.docx
External Libraries
Scratches and Consoles

1 #Author : Sai Nalla
2
3 def string_op():
4     try:
5         inp_string = input("Enter your sentence here:")
6         if inp_string != '' and inp_string is not None and inp_string.isspace() != True and inp_string.isnumeric() != True:
7             inp_string = inp_string.replace('python','pythons')
8             print(inp_string)
9         else:
10            print("please enter a valid sentence")
11    except Exception as error:
12        print("Error occured {}".format(error))
13    #end of block3
14    if __name__ == "__main__":
15        string_op()

string_op() > try > else

Run: icp1_3 ×
C:\Users\chinna\AppData\Local\Programs\Python\Python310\python.exe D:/Sai/ICP_1_NNDL/icp1_3.py
Enter your sentence here:I love playing with python and python is most popolvar language
I love playing with pythons and pythons is most popolvar language

Process finished with exit code 0

Activate Windows
Go to Settings to activate Windows
```

4) . Use the if statement conditions to write a program to print the letter grade based on an input class score. Use the grading scheme we are using in this class.

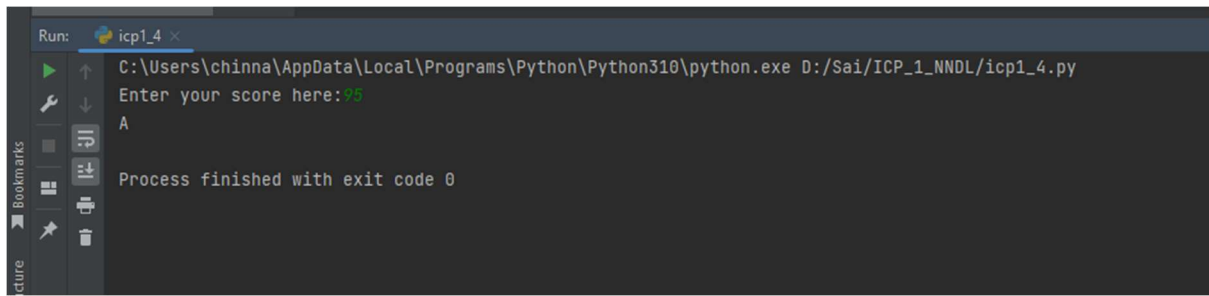
```
icp1.py x icp1_2.py x icp1_3.py x icp1_4.py x
1  #Author : Sai Nalla
2  #start
3  def grading_classScore():
4      try:
5          try:
6              class_score = int(input("Enter your score here:"))
7          except ValueError:
8              print("Please enter number in numeric format not strings")
9              return None
10         if class_score != '' and class_score is not None:
11             if class_score > 100 or class_score < 0:
12                 print("Score not in range please enter a valid score")
13             else:
14                 if class_score >= 90 and class_score <= 100: #Range of Grade A score
15                     print("A")
16                 elif class_score >= 80 and class_score <= 89: #Range of Grade B score
17                     print("B")
18                 elif class_score >= 70 and class_score <= 79: #Range of Grade C score
19                     print("C")
20                 elif class_score >= 60 and class_score <= 69: #Range of Grade D score
21                     print("D")
22                 else:
23                     print("F") #Grade F score
24
```

```

23         else:
24             print("please enter a valid number")
25     except Exception as error:
26         print("Error occurred {}".format(error))
27 #end
28 if __name__ == "__main__":
29     grading_classScore()
```

## Output

```
Run: icp1_4 x
C:\Users\chinna\AppData\Local\Programs\Python\Python310\python.exe D:/Sai/ICP_1_NNDL/icp1_4.py
Enter your score here:65
D
Process finished with exit code 0
```



Run: icp1\_4 x

```
C:\Users\chinna\AppData\Local\Programs\Python\Python310\python.exe D:/Sai/ICP_1_NNDL/icp1_4.py
Enter your score here:95
A
Process finished with exit code 0
```

This screenshot shows a Python IDE console window. The title bar indicates the file 'icp1\_4.py'. The command prompt shows the execution of the script. The user entered '95' as a score, and the program outputted 'A'. The process finished with exit code 0.



Run: icp1\_4 x

```
C:\Users\chinna\AppData\Local\Programs\Python\Python310\python.exe D:/Sai/ICP_1_NNDL/icp1_4.py
Enter your score here:25
F
Process finished with exit code 0
```

This screenshot shows the same Python IDE console window. The user entered '25' as a score, and the program outputted 'F'. The process finished with exit code 0.