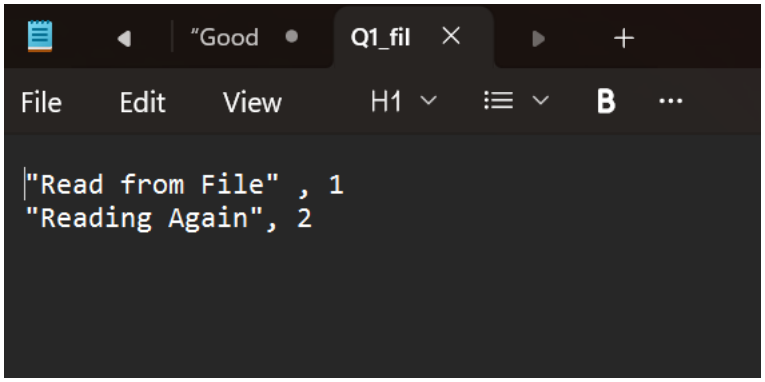
Q.1) Write a code to Read a file and append lines to a list.

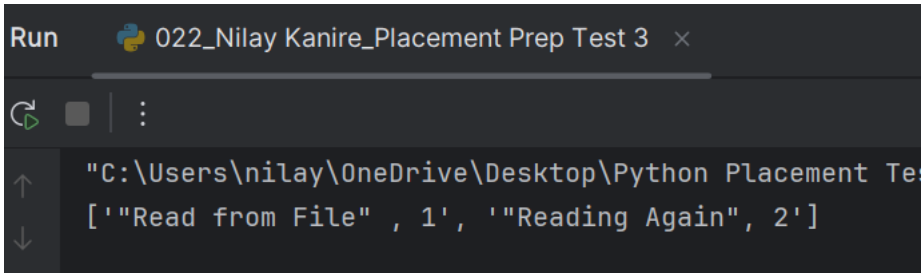
File content ->



Code

```
path = "D:\\downloads\\Q1_file.txt"
lines_list = []
with open(path, 'r') as file:
   for line in file:
      lines_list.append(line.strip())
print(lines_list)
```

#Output ->



Q.2) Write a code to catch an Exception in python?

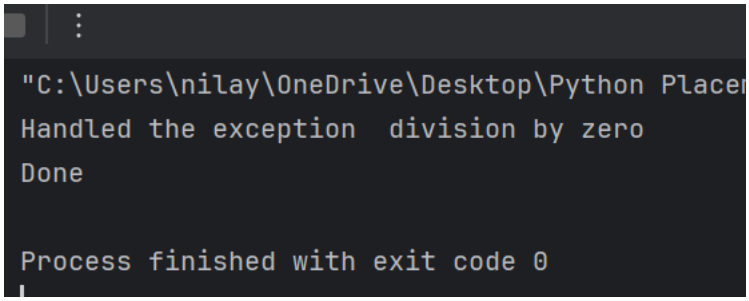
```
num = 10
deno = 0

try:
   div = 10/0
except Exception as e:
```

```
    print("Exception handled: ", e)

print("Done")
```

Output ->



Q.3) Write a Python function that accepts a list containing strings and integers. Merge all string elements using # and add all integer elements.

e.g.
input list is
['100', 'welcome', 'hi', '200', '300', 'bye', 'welldone', '500']

Output should be:
welcome#hi#bye#welldone#
1100

Q.4) Write a script to sort a dictionary based on its values and find the sum of middle two values

input_dict = {"x": 5, "y": 15, "z": 25}
Output:
Sorted Dictionary: {'x': 5, 'y': 15, 'z': 25}
Sum of middle two values: 15 + 5 = 20

or

input_dict = {"x": 5, "y": 15, "z": 25,"p":12}
Output:
Sorted Dictionary: {'x': 5, 'p': 12,'y': 15, 'z': 25}

Sum of middle two values: 12 + 15 = 27

```
input_dict1 = {"x": 5, "y": 15, "z": 25}
sorted_dict1 = dict(sorted(input_dict1.items(), key=lambda item: item[1]))
input_dict2 = {"x": 5, "y": 15, "z": 25, "p": 12}
sorted_dict2 = dict(sorted(input_dict2.items(), key=lambda item: item[1]))
```

```
print("Sorted Dictionary 1 :", sorted_dict1)
values1 = list(sorted_dict1.values())
middle_values = values1[len(values1)//2 - 1 : len(values1)//2 + 1]
print("Sum of middle two values of Dictionary 1:", sum(middle_values))

print("\n Sorted Dictionary 2 :", sorted_dict2)
values2 = list(sorted_dict2.values())
middle_values = values2[len(values2)//2 - 1 : len(values2)//2 + 1]
print("Sum of middle two values of Dictionary 1:", sum(middle_values))
```

Output ->