

'''Write a code to Read a file and append lines to a list.'''

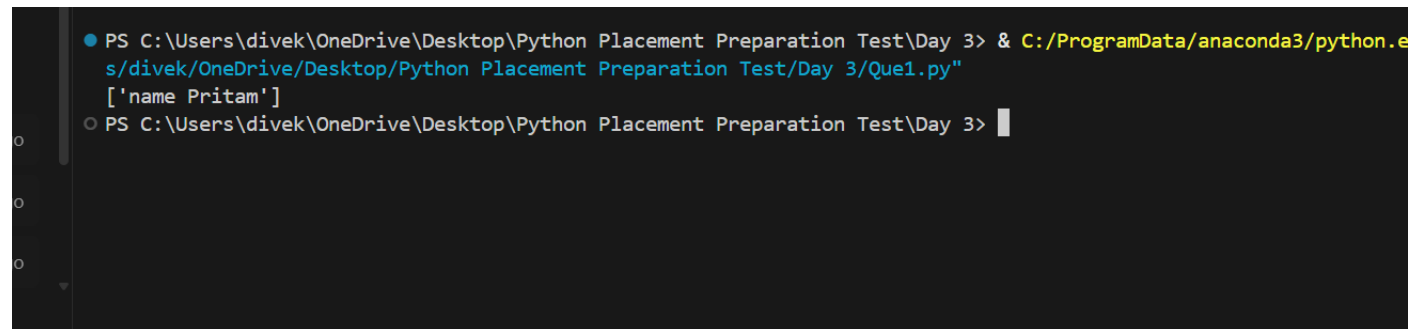
```
lines=[]
```

```
with open('text.txt', 'r') as f:
```

```
    for line in f:
```

```
        lines.append(line)
```

```
print(lines)
```

A screenshot of a Windows command prompt window. The title bar is partially visible at the top. The command prompt shows a blue prompt character followed by the command: PS C:\Users\divek\OneDrive\Desktop\Python Placement Preparation Test\Day 3> & C:/ProgramData/anaconda3/python.exe s\divek/OneDrive/Desktop/Python Placement Preparation Test/Day 3/Que1.py". The output is displayed on the next line: ['name Pritam']. Below the output, there is another blue prompt character followed by: PS C:\Users\divek\OneDrive\Desktop\Python Placement Preparation Test\Day 3> . The terminal has a dark background with light-colored text. There are three small circular icons on the left side of the terminal window, likely representing window management controls.

```
● PS C:\Users\divek\OneDrive\Desktop\Python Placement Preparation Test\Day 3> & C:/ProgramData/anaconda3/python.exe s\divek/OneDrive/Desktop/Python Placement Preparation Test/Day 3/Que1.py"
  ['name Pritam']
○ PS C:\Users\divek\OneDrive\Desktop\Python Placement Preparation Test\Day 3> 
```

'''Write a code to catch an Exception in python? '''

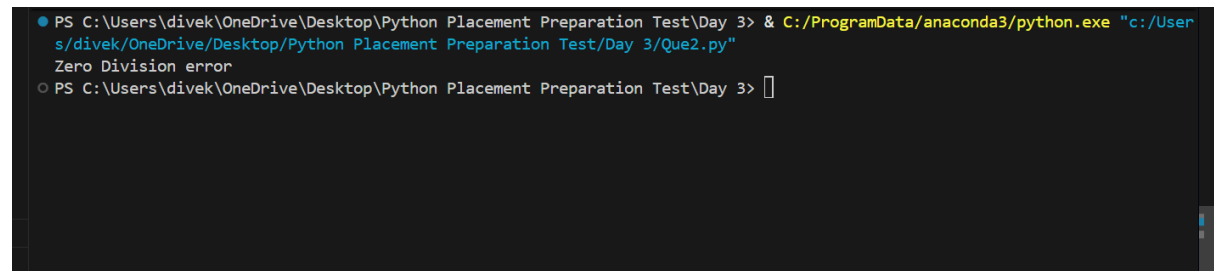
try:

 x=10/0

 print(x)

except ZeroDivisionError:

 print("Zero Division error")



```
PS C:\Users\divek\OneDrive\Desktop\Python Placement Preparation Test\Day 3> & C:/ProgramData/anaconda3/python.exe "c:/Users/divek/OneDrive/Desktop/Python Placement Preparation Test/Day 3/Que2.py"
Zero Division error
PS C:\Users\divek\OneDrive\Desktop\Python Placement Preparation Test\Day 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
```

'''

```
li=['100', 'welcome', 'hi', '200', '300', 'bye', 'welldone', '500']
```

```
total=0
```

```
stri=""
```

```
for i in li:
```

```
    try:
```

```
        total=total+int(i)
```

```
    except:
```

```
        stri=stri+str('#')+str(i)
```

```
print(stri)
```

```
print(total)
```

```
PS C:\Users\divek\OneDrive\Desktop\Python Placement Preparation Test\Day 3> & C:/ProgramData/anaconda3/python.exe "c:/Users/divek/OneDrive/Desktop/Python Placement Preparation Test/Day 3/Que3.py"
● #welcome#hi#bye#welldone
  1100
○ PS C:\Users\divek\OneDrive\Desktop\Python Placement Preparation Test\Day 3>
```

'''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_dict = {"x": 5, "y": 15, "z": 25}
```

```
sorted_items = sorted(input_dict.items(), key=lambda x: x[1])
```

```
sorted_dict = dict(sorted_items)
```

```
print("Sorted Dictionary:", sorted_dict)
```

```
values = list(sorted_dict.values())
```

```
if len(values) % 2 == 1:
```

```
    m1 = values[0]
```

```
    m2 = values[1]
```

```
else:
```

```
    m1 = values[1]
```

```
    m2 = values[2]
```

```
sum=m1+m2
```

```
print(f"Sum of middle two values:{sum}")
```

```
PS C:\Users\divek\OneDrive\Desktop\Python Placement Preparation Test\Day 3> & C:/ProgramData/anaconda3/python.exe "c:/Users/divek/OneDrive/Desktop/Python Placement Preparation Test/Day 3/Que4.py"
● Sorted Dictionary: {'x': 5, 'y': 15, 'z': 25}
  Sum of middle two values:20
○ PS C:\Users\divek\OneDrive\Desktop\Python Placement Preparation Test\Day 3>
```

