

Output:-

-Grade List : [85, 90, 78, 92, 86]

Enter the index of the grade you want to view : 10

Invalid index .please enter a valid index.

201825

Task 9: Implement Exceptions and Exceptional handling in python.

Aim:- To implement Exceptions and Exceptional handling in python

Problem 9.1

Algorithm:-

1. Start the program
2. Initialize a list of grades
3. Prompts the user to enter the index of the grade they wish to view.
4. Attempts to display the grade at the specified index.
5. If the index is out of range, Catch the IndexError and print an error message

Program :-

```
# Initialize the list of grades
```

```
grades = [85, 90, 78, 92, 88]
```

```
# Display the grades list
```

```
Print("Grade List", grades)
```

```
# Prompt the user to enter the index of the grade they want to view.
```

```
try:
```

```
    index = int(input("Enter the index of the grade you want to view:"))
```

```
# Attempt to display the grade at the specified index
```

```
    Print f("The grade at index {index} is {grades[index]}")
```

```
except IndexError:
```

Output:-

Enter the numerator : 10

Enter the denominator : 0

ERROR!

Error:- Division by zero is not allowed.

Problem 9.2

Algorithm:-

1. Start the program
2. Prompts the user to enter two numbers: a numerator and a denominator.
3. Attempts to divide the numerator by denominator.
4. If the denominator is zero, catches the Zero Division Error and displays an error message.

Program:-

```
# Function to perform division
def divide_number():
    try:
        # Prompt the user to enter the numerator
        numerator = float(input("Enter the numerator:"))

        # Prompt the user to enter the denominator
        denominator = float(input("Enter the denominator:"))

        # Attempt to perform division
        result = numerator / denominator

        print(f"Result: {result}")
    except ZeroDivisionError:
        # Handle division by zero error
        print("Error: Division by zero is not allowed!")
    except ValueError:
        # Handle invalid input that is not a number
        print("Error: Please enter valid numbers!")

# Call the function to execute the division operation divide_numbers()
```

Output :-

Enter a number: 18

Exception occurred: Invalid Age

Problem 9.3:-

Algorithm:-

1. Define the custom exception.
2. Prompt the user for input.
3. Check if the age is below 18.
4. Raise an exception if the condition is met.
5. Handle the exception with a custom error message.

Program:-

```
# define Python user-defined exceptions
class InvalidAgeException:
    "Raised when the input value is less than 18"
    pass
# You need to guess this number
number = 18
try:
    input_num = int(input("Enter a number:"))
    if input_num < number:
        raise InvalidAgeException
    else:
        print("Eligible to vote")
except InvalidAgeException:
    print("Exception occurred: Invalid Age")
```

Result:-

Thus the program for Implement Exceptions and Exceptional handling is executed and verified successfully

Age (in years)	VEL TECH
PERFORMANCE (5)	9
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	3
RECORD (5)	3
TOTAL (20)	21.0
SIGN WITH DATE	16/10/2023