17/9/25 Task-9.1 Implement Exceptions and Exceptional handling python

AIM :- To implement Exceptions and Exceptional handling in python.

ALGORITHM :-

1. Start the program.
2. Initializes a list of grades (e.g., 85, 90, 78, 92, 88).
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. End.

Program :-

```
# initialize the list of grades.
grades = [85, 90, 78, 92, 88]

# Display the grades list
Print ("Grades 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
range.
Print ("Invalid index. Please enter a valid index.")
except value Error ,
    Print ("Invalid Input. Please enter a numerical
            index.").
```

Output :-

Grades list : [85, 90, 78, 92, 88]

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

Invalid index. Please enter a valid index.

Task 9.2 Exceptions and Exceptional handling

AIM :- To develop a python calculator program that performs basic arithmetic operations.

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 the denominator.
4. If the denominator is zero, catches the ZeroDivision Error

PROGRAM :-

```
def divide_numbers():
    try:
        numerator = float(input("Enter the numerator:"))
        denominator = float(input("Enter the denominator:"))
        result = numerator / denominator
        Print(f"Result: {result}")
    except ZeroDivision Error:
        Print("Error: Division by zero is not allowed")
    except value Error:
        Print("Error: Please enter valid numbers")
```

Output:

```
Enter the numerator : 8
Enter the denominator: 0
ERROR!
Error : Division by zero is not allowed.
```

AIM :- To build a python application to determine if a person is eligible to vote based on their Age.

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 :-

```
Class InvalidAgeException(Exception):
    "Raised when the input value is less than 18"
    pass
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 occured: Invalid Age")
```

PERFORMANCE (5)	5
RESULT AND ANALYSIS (3)	3
VIVA VOCE (3)	3
RECORD (4)	4
TOTAL (15)	
SIGN WITH DATE	15

RESULT :- Thus the program for Implement Exceptions and Exceptional handling is executed and verified successfully.

Output :- Enter a number: 15
Exception occured: Invalid Age