Task 9: implement Expections and exceptional handling in python

Aim:- To implement expections and exceptional handling in Python

Algorithm:-

1. Start the program
2. initialize a list of grades (e.g [85, 90, 78, 92, 88]).
3. prompts the user to enter a 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. catches the indexerror and print error "invalid index. please index a valid index".

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(intput("Enter the index of grade you want to view:"))
#Attempt to display the grade at the specified index
print (f" the grades at index{index} is: {grades[index]}")
Except Indexerror:
    # handle the case where the index is out of range
    print(" invalid index. please enter a validindex.")
Except Value error:
    # handle the Case where the input is not an integer
    print(" invalid input .please enter a numerical index.")
```

output:-

gradelist: [85,90,78,92,88]

Enter the index of the grade you want to view: 85

invalid index. please enter a valid index

Task:- 9.2

Aim:- To write a python calculator program that perform basic arithemetic 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 zero division Error and displays an error message: "Error: Division by zero is not allowed"

Program:-

```
# Function to perform division
def divide_numbers():
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
  expect zero divisionerror:
  # Handle division by zero error
  print("Error: Division by zero is not allowed.")
  Expect value error:
  # 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 the numerator :10

Enter the denominator :12

result : 1.2

output:-

Enter a number: 19

Eligible to vote

Task 9.3 Determine eligibility to vote

Aim:- To write 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 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 Invalid age expection (expection):
  "Raise when the input values 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 Invalid age expection
  else:
    print("Eligible to vote")
except Invalid age expection:
  print("Expection occurred: Invalid age")
```

VELTECH	
EX No.	9
PERFORMANCE (5)	5
RESULT AND ANALYSIS (3)	5
VIVA VOCE (3)	5
RECORD (4)	
TOTAL (15)	
SIGN WITH DATE	15

Thus the python application to determine if a person is eligible to vote based on their age:-