Task9:- To Implement Exceptions and exceptional handling in python.

Algorithm:-

1. start the program
2. Initialize a list of grades(eg.., [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. If the index is out of range, catches the indexError and prints an error message, 'Invalid index. please enter a valid index."

Program:

```
#Initialize the list of grades
grades = [85,90,78,92,88]
#Display the grades list
print("GradesList:", grades)
#prompt the user to enter the index of grade they want to view try:
    index = int(input("Enterthe index of the grade you want to view:"))
    #Attempt to display the grade at the specified index
    Print(f"The grade atindex{index}is:{grades[index]}")
except Index Error:
    #Handle the case where the index is out of range
    Print("Invalid index. please enter a valid index.")
except valueError:
    #Handle the case where the input is not an integer
    print("Invalid input. please enter a numerical index.")
```

output:-

gradeslist: [85, 90, 78, 92, 88]
enter the index of grade you want to view:10
invalid index. Please enter a valid index.

9.2:- You are developing a python calculator ~~python~~ Program that performs basic arithmetic operations.

Algorithm:-

* Start the program
* prompts the user to enter two numbers: a numerator and a denominator.
* Attempts to divide the numerator by the denominator
* If the denominator is zero, catches the ZerodivisionError 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 numberator
        numerator = float(input("Enter the numerator:"))
        #Prompt the user to enter the denominator
        deminator = 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 the numerator: 10

Enter the denominator: 0

Error: Division by zero is not allowed.

9.3:- you are building a python application to determine if a person is eligible to vote based on their age. According to the rules, only individuals who are 18 years or older are allowed to vote.

Algorithm:-

Define the custom exception
Prompt the user for input.
check if the age is below 18.
Raise an exception if the condition is met
Handle the exception with a custom error message

Program:-

```
#define python user-defined exceptions
class InvalidAgeException(Exception):
    "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 Invalid age exception
    else:
        Print("Eligible to vote")
except Invalid ageexception:
    Print("Exception occurred. Invalid age")
```

VEL TECH	
EX No.	9
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	5
TOTAL (20)	20
SIGN WITH DATE	8/10

Result:-

Thus the program for implement exceptions and exceptional handling is executed and verified successfully

output:-

enter a number :12

Exception occurred : invalid Age.