Output:-

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

Enter the valid index of the grade you want to view : 1, 3

Invalid index. Please enter a valid index.

Task 9. Implement Exceptions and Exceptional handling in Python.

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

Algorithm:-

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

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 grade they want
                      to view:"))
    #Attempt to display the grade at the specified
                                        index
    printf("The grade at index {index} is: {grades[index]}")
except IndexError:
    #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
                                integers
    print("Invalid input. Please enter a numerical index.")
```

Result:- Thus the python program to implement the Grades list was successfully executed.

Output:-

```
Enter the numerator : 10
Enter the denominator : 0
ERROR!
Error: Division by zero is not allowed.
```

Q.2

Aim: To write a python program 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 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
except ZeroDivisionError:
    #Handle division by zero error
    Print("Error: Division by zero is not allowed.")
except ValueError:
    #Handle Division invalid input that is not a number
    print("Error: please enter valid numbers.")
#Call the function to execute the division operation
divide_numbers()
```

Result:

Thus the program to calculator program that performs basic arithmetic operations was successfully executed.

Output :-

Enter a number : 17

Exception occurred : Invalid Age

Task 9.3 Determine eligibility to vote.

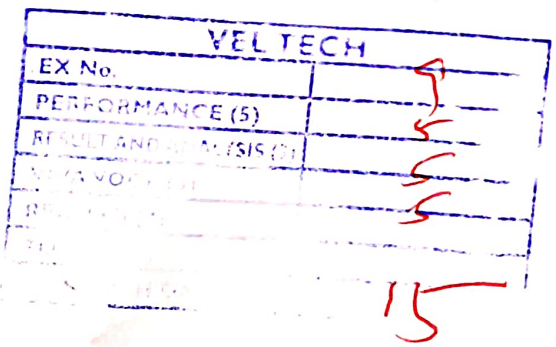
Aim: To write a python application to determine if a person is elegible to vote based on their age.

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 InvalidAgeException(Exception):
    "Raised when the input values is less than 18"
    pass
#You need to guess this number
ifumber = 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 python application to determine if a person is eligible to voted based on their age.