

Task-9.1

Implement Exceptions and exceptional handling in Python.

Aim:- To implement exception and exceptional handling in Python.

Algorithms

1. Start the Program.
2. Initialize a list of grades.
3. Prompts the user to enter the index of the grade.
4. Attempts to display the grade at the index.
5. If the index is out of range catches the index error.

Program:-

```
# initialize the list of grades
grades = (88, 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.
Index = int(input("Enter the index to view:"))

# Attempt to display the grade
Print ("The grade at index")

except IndexError:
# Handle the case where the index.
Print ("Invalid error")

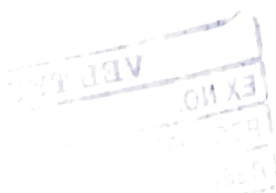
# Handle the case where the index is out of range.
Print ("Invalid/error")
```

output:-

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

Enter the index of the grade you want to view:

Invalid error: Please enter a valid index.



[Faint handwritten text at the bottom of the page, partially obscured by a red line.]

Output:-

Enter the numerator: 10

Enter the denominator: 2

Result: 5

Error: Division by zero is not allowed

Task-9.2

Aim:- To develop a Python calculator Program that perform basic arithmetic operations.

Algorithm:-

1. start the Program.
2. Prompts the user to enter two numbers.
3. attempts to divide the numerator by denominator.
4. If the denominator is zero.

Program:-

```
# Function to perform division.
def divide-numbers():
    try:
        # Prompt the user to enter
        numerator = float(input("Enter numerator: "))
        # Prompt the user to enter
        denominator = float(input("Enter denominator: "))
        # Attempt to perform division.
        result = numerator / denominator
        print("result: {result}")
    except ZeroDivisionError:
        # Handle division by zero error.
        print("Error")
    except ValueError:
        # Handle invalid input.
        print("Error")
    # call the function to execute.
    divide-numbers()
```

Output:-

Enter a number: 15

Exception occurred: Invalid Age

Task-9.3.

Aim:-

To building 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.
5. Handle the exception with a custom.

Program:- # define Python user-defined exceptions.

class InvalidAgeException:

Raised when the input value.

You need to guess this number
number = 18.

try:

input_num = int(input())

If input_num < number.

raise InvalidAgeException.

else:

print("eligible to vote")

except InvalidAgeException:

print("Exception occurred")

VEL TECH	
PERFORMANCE (5)	9
RESULTS ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	5
TOTAL (20)	
: WITH DATE	

Result:-

Thus the Program for implement exceptions and handling is executed and verified successfully.