

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 grade
4. attempts to apply display the grade at the specific index

Program

initialize the list of grades

grades = [85, 90, 78, 42, 88]

Print("grades list:", grades)

index = int(input("enter the index of grade you want to view:"))

Print(f"the grade at index {index} is: {grades[index]}")

~~except index error~~

Print("Invalid index please enter a valid index")

~~except value error~~

Print("invalid input please enter a numerical index")

output:

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

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

Invalid index Please enter a valid index

VRL TECH	
EX NO.	2
PERFORMANCE (%)	2
RESULT AND ANALYSIS (%)	2
MARKS (%)	2
REMARKS (%)	
DATE	21

Q.2 You are developing a python calculator Program that performs Basic arithmetic operations

Algorithm:

1. Start the Program
2. Prompts the user to enter two no
3. attempts to divide the numerator denominator
4. If the denominator is zero, catches the zero division 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 ZeroDivisionError:  
        print("Error: Division by zero is not allowed")  
    except ValueError:  
        print("Error: Please enter valid number")
```

Task 4: Implement Exceptions and Exception Handling in Python

Output:

Enter the numerator: 10

Enter the denominator: 0

Error!

Error: Division by zero is not allowed

Q3: 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. To enforce this rule you decided to create a custom exception called Invalid Age exception.

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

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 Age exception")
```

```
except InvalidAgeException:
```

```
    print("exception occurred: Invalid Age")
```

VEL TECH	
EX NO.	8
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VA VOCE (5)	5
RECORD (5)	
L (20)	
DATE	15

Result: Thus the program for implement exceptions and Exceptional handling is executed and verified successfully.