

TASK 9:- Implement Exceptions and exceptional handling in python

Aim:- To implement Exceptions and exceptional handling in python

Problem 9.1 You are developing a python program that process a list of Student's grades the program is designed to allow the user to select a grade by specifying an index number however, you need to ensure that the program handles cases where the user input an index that is out of range an index that does not exist the list

Algorithm:-

- 1) Start the program
- 2) Initialize the list of grades (85, 90, 78, 92, 88)
- 3) Prompts the user to enter the index of the grade they wish to view
- 4) Attempt to display the grade at the specified index
- 5) if the index is out of range, catches the index Error and print 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
```

Output :-

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

Enter the index of the grade you want to

View : 10

Invalid index Please enter a valid index

selected element is the 4th item of list

and its value is 92

88 is the 5th item of list

[88, 92]

88

```
print("Grades List:", grades)
# Prompt the user to enter the index of
# the grade they want to view by:
index = int(input("Enter the index of the grade
                  you want to view:"))
# Attempt to display the grade at the specified
# index
print("The grade at index {} is: {}".format(index, 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 integer
print("Invalid input please enter a
      numerical index")
```

Problem 9.2 you are developing a python calculator
program that performs basic arithmetic operations
one of the key functionalities is to divide two
numbers entered by the users however dividing by
zero is not allowed and would cause the program
to crash if not handled properly

- 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

OUTPUT:-

Enter the numerator: 10

Enter the denominator: 0

ERROR!

Error: Division by zero is not allowed.

4) If the denominator is zero catches the zero division error and display 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

print(f"Result: {result}")

except ZeroDivisionError:

Handle division by zero error

print("Error: Division by zero is not allowed")

Call the function to execute the division -

~~operation~~ divide_numbers()

PROBLEM 4.3: 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 or older are allowed to vote To enforce this rules you decide to create a custom exception called invalid age exception, which will be raised whenever an age below 18 is entered

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
- 5) 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 invalidAgeException
```

```
    else:
```

```
        print("Eligible to vote")
```

```
except invalidAgeException:
```

```
    print("Exception occurred: invalidage")
```

Output :-

Enter a number: 15

Exception occurred: invalid Age

OK

VEL TECH	
EX No.	
PERFORMANCE (5)	✓
RESULT AND ANALYSIS (5)	✓
IVA VOCE (5)	✓
RECORD (5)	✓
TOTAL (20)	✓
SIGN WITH DATE	✓

Result:- Thus the program implement exceptions and exceptional handling is executed and verified successfully