Practical No.5 → Exception Handling.

SDS | Page No. | Date / /

Q1] Write a python program to implement try-except in simple program.

Code:

```
x = 10
y = 20
try :
    print("x :",x)
    print(z)
Except NameError :
    print("Z is not defined")
    print("Y :", y)
```

Output:

```
x : 10
Z is not defined.
Y : 20.
```

Q2] Write a python program to use multiple except.

Code:

```
first = 2
second = 9
third = second/first
print("Ans = "; third)
try :
    fourth = int(input("Enter a number :"))
    third = second / fourth
    print("Ans = ", third)
except ValueError :
    print("Please enter valid input!")
```

```
except ZeroDivisionError:
    print("zero not allowed!")

except ArithmeticError:
    print("please enter value greater than zero")
```

Output:

```
Ans = 2.0
Enter a number: 5
please enter valid input.
```

33] Write a program to use else clause with an except block

Code:

```
first = 2
second = 9
third = second / first
print("Ans = ", third)
try:
    fourth = int(input("Enter a number:"))
    third = second / fourth
    print("Ans =", third)
except Exception:
    print("Please enter value greater than zero");
else:
    print ("Success!!")
```

Output:

```
Ans = 2.0
Enter a Number: 0
please enter a value greater than zero
```

```
Ans = 2.0
Enter a Number: 0
Ans = 4.5
Success!!
```

SDS | Page No. | Date

Q4] Write a python program to use an assertion statement.

(a) with an error message.

Code:

```
def Student(sub_marks):
    assert len(sub_marks) != 0, "No value found"
    return sum(sub_marks)/len(sub_marks)
Student1 = [10,20,30]
print('Average marks = ', Student(Student1))

Student2 = []
print('Average marks = ', Student(Student2))
```

Output:

Error occurred on line 2 & 9.

AssertionError : No value found.

(b) without an error message.

Code:

```
def student(sub_marks):
    assert len(sub_marks) != 0
    return sum(sub_marks)/len(sub_marks)
Student1 = []
print("Average marks =', student(student1))
```

Output:

AssertionError.

SDS | Page No. | Date

Q5] Write a python program to show user defined exception.

Code:

```
class UserException(Exception):
    def __init__(self):
        self.demo = "you created a User defined exception"
try:
    print("Demo for user defined exception \n")
    raise UserException
except UserException:
    print(UserException().demo).
```

Output:

Demo for user defined Exception
you created a user defined exception.

Q6] Write a python program to handle multiple errors with one except statement.

Code:

```
try:
    variable = 55
    if variable < 100:
        variable2 = variable2 / (variable2 - 55)
        print("Value of variable2 =", b)
        variable2 = [1, 2, 3]
        print(variable3[0])
except (IndexError, ZeroDivisionError, NameError):
    print("\n This is program throws multiple errors like
        IndexError, ZeroDivisionError, NameError")
```

Output:

This program throws multiple errors like IndexError, ZeroDivision Error, NameError.)

SDS Page No.
Date / /

Q.7] Write a python program to create a game a character using user defined exceptions.

Code:

```
class OwnException(Exception):
    pass
Class EarlierLetter(OwnException):
    pass
character = 's'
while True:
try:
    demo = input("Please enter a character:-")
    if demo < character:
        raise EarlyLetter
    elif demo> character:
        raise LaterLetter
except EarlyLetter:
    print("OOps!! The entered alphabet is preceding one, try again!")
    print(" ")
except LaterLetter:
    print("OOps!! The entered alphabet is succeeding one, try again!")
    print(" ")
else:
    print("Hurray!! You gussed correctly!")
    break
```

Output:

```
Please enter a character:- f
Oops!! The entered alphabet is precceding one, try again!
Please enter a character:S
Hurray!! You guessed correctly!.
```

24/2/23