```
11) Enter First Number: 10
12) Enter Second Number: 0
13) ArithmeticError
```

Single except block that can handle multiple exceptions:

We can write a single except block that can handle multiple different types of exceptions.

except (Exception1,Exception2,exception3,..): or
except (Exception1,Exception2,exception3,..) as msg :

Parenthesis are mandatory and this group of exceptions internally considered as tuple.

Eg:

```
1)  try:
2)      x=int(input("Enter First Number: "))
3)      y=int(input("Enter Second Number: "))
4)      print(x/y)
5)  except (ZeroDivisionError,ValueError) as msg:
6)      print("Plz Provide valid numbers only and problem is: ",msg)
7)
8)  D:\Python_classes>py test.py
9)  Enter First Number: 10
10) Enter Second Number: 0
11) Plz Provide valid numbers only and problem is:  division by zero
12)
13) D:\Python_classes>py test.py
14) Enter First Number: 10
15) Enter Second Number: ten
16) Plz Provide valid numbers only and problem is:  invalid literal for int() with b
17) ase 10: 'ten'
```

Default except block:

We can use default except block to handle any type of exceptions.
In default except block generally we can print normal error messages.

Syntax:

```
except:
    statements
```

Eg:

```
1)  try:
2)      x=int(input("Enter First Number: "))
3)      y=int(input("Enter Second Number: "))
4)      print(x/y)
```