NETWORKZ SYSTEMS
AN ISO 9001:2015 CERTIFIED COMPANY
CHINNAKADA , KOLLAM

EXCEPTON HANDLING IN PYTHON

Python has many built-in exceptions which forces your program to output an error when something in it goes wrong.

When these exceptions occur, it causes the current process to stop and passes it to the calling process until it is handled. If not handled, our program will crash.

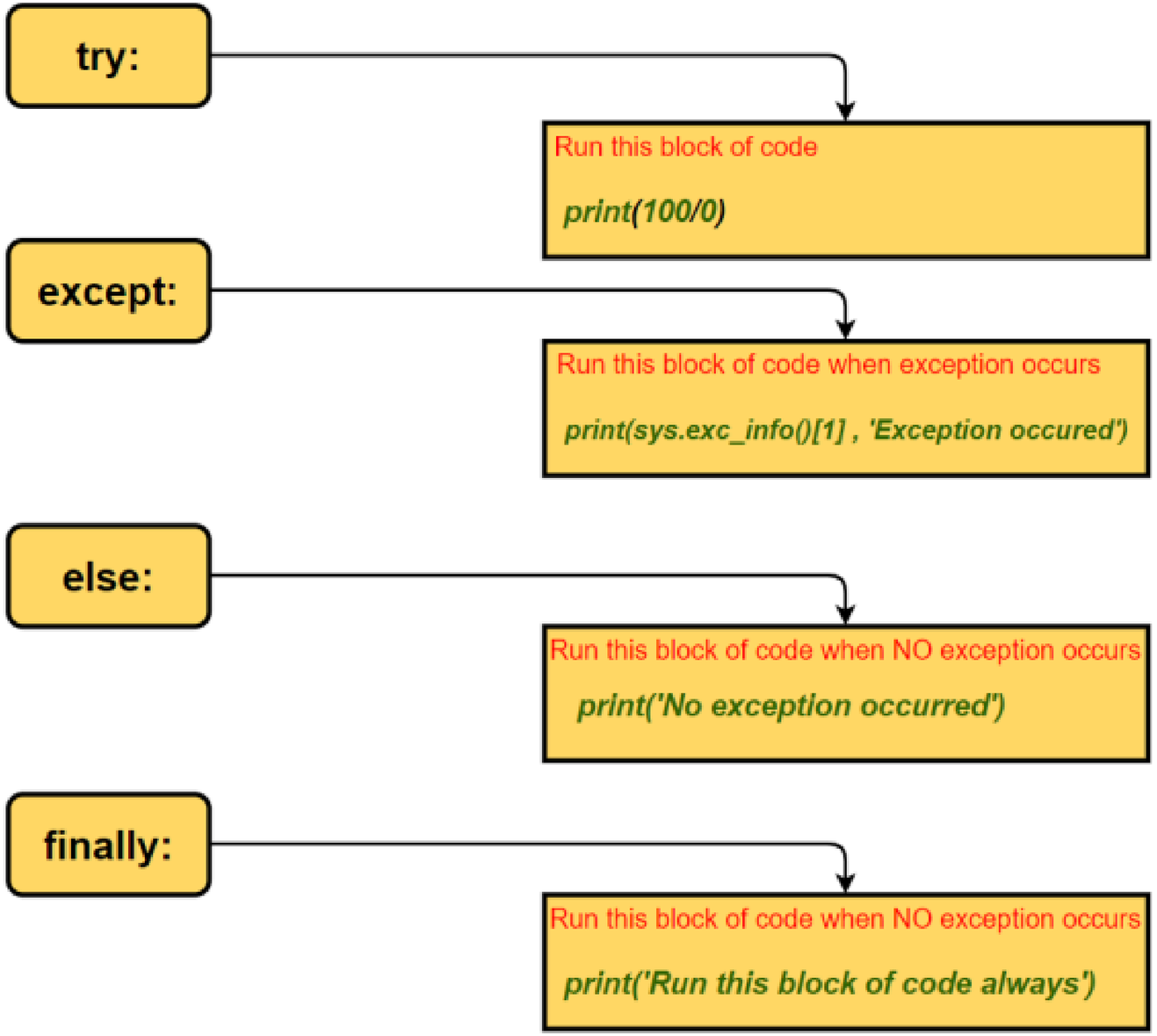
Catching Exceptions in Python :

Exceptions in python can be handled using a try statement. A critical operation which can raise exception is placed inside the try clause and the code that handles exception is written in except clause.

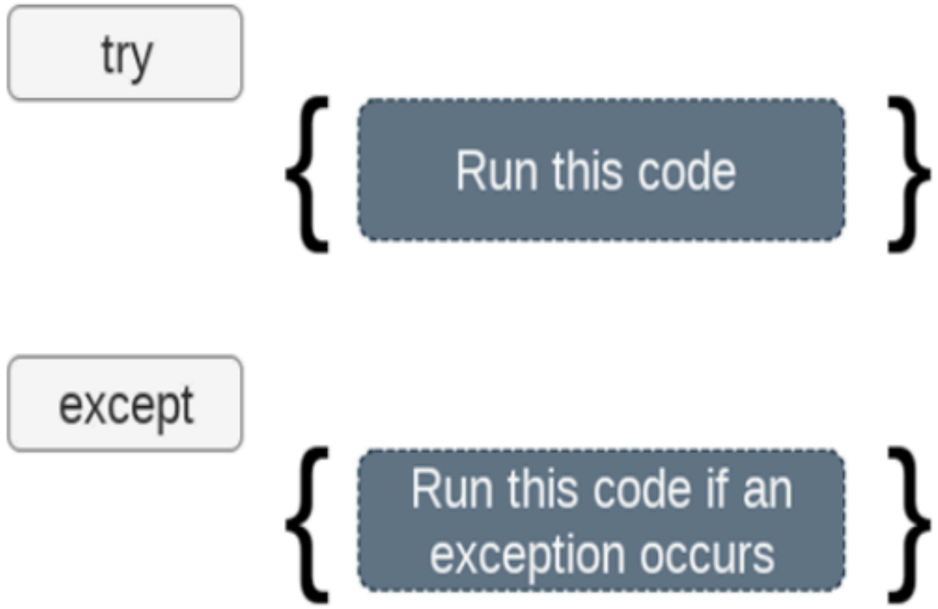
We can also use the else keyword to define a block of code to be executed if no exceptions were raised.

The finally code block will execute regardless of the result of the try and except blocks

Python has many built-in exceptions (ArithmeticError, ZeroDivisionError, EOFError, IndexError, KeyError, SyntaxError, IndentationError, FileNotFoundError etc) that are raised when your program encounters an error



Try-Except



Syntax

```
try:
    #block of code
except Exception1:
    #block of code
except Exception2:
    #block of code
#other code
```

The python allows us to declare the multiple exceptions with the except clause. Declaring multiple exceptions is useful in the cases where a try block throws multiple exceptions

```
try:
#block of code

except (Exception 1,Exception 2,Exception 3,...,Exception n):
#block of code

else:
#block of code
```

```
try:
   a=10/0
except (ArithmeticError,ZeroDivisionError):
    print("Arithmetic Exception")
else:
    print("Successfully Done“)
```

Raising exceptions

An exception can be raised by using the raise clause in python.To raise an exception, raise statement is used. The exception class name follows it.An exception can be provided with a value that can be given in the parenthesis.

```
try:
    age = int(input("Enter the age?"))
    if age<18:
        raise ValueError
    else:
        print("the age is valid")
except ValueError:
        print("The age is not valid")
```

Example 2

```
try:
    a = int(input("Enter a?"))
    b = int(input("Enter b?"))
    if b is 0:
        raise ArithmeticError;
    else:
        print("a/b = ",a/b)
except ArithmeticError:
        print("The value of b can't be 0")
```

THANK YOU

NETWORKZ SYSTEMS

AN ISO 9001:2015 CERTIFIED COMPANY

CHINNAKADA , KOLLAM