**Error handling**

**What is Error Handling?**

An Exception is an error that Happens during execution of a program, when that error occurs, python generate an Exception that can be handled, which avoids your program to crash.

**Why use Exceptions?**

Exception are convenient in many ways for handling Error and special conditions in a program, when you think That you have a code which can produce an error Then you can use exception handling

**Compile-Time Errors**

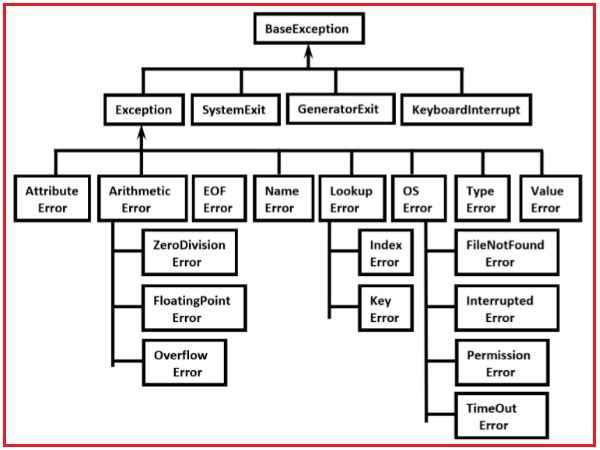
Errors that occurs when you violate the ruler of writing Syntax are knows as compile time errors. This compiler Error indicates something that must be fixed before the code can be compiled. All these error are detected by the compiler and thus are know as compile time error.

Missing symbols like [], [], ()

Printing the value of variable without declaring Missing colon (terminator)

**Run-Time Errors**

Errors which occur during program execution (run-time) after successful are called run-time errors. one of the most common runtime error is division by zero Also known as Division error. these types of error are hard to find as compiler doesn't point to the line at which the error occurs.



Error\_handling.py

try:

    print(a)

except:

    print("error vachindi")

else:

    print("errot raledu")

finally:

    print("always")

try:

    print('a'+10)

    print(a)

except TypeError:

    print("Type Error occured")

except NameError:

    print("Name Error occured")

try:

print('a'+10)

except :

print("Error occured")

try:

print('a'+10)

except Exception as uma:

  print("Error occured")

try:

   print('a'+10)

except type Error:

try:

       print()

    except:

        print("inner")

try:

    print('a'+10)

except:

    print("outer")

    try:

        print(a)

    except:

        print("inner")

Different types of errors

Attributerror - Raised by syntax obj.foo, if object has no member named foo

EoFerRor - Raised of "end of file reached for console or file

Io error-raised upon failure of i/o operation (e.g:opening file )

Index Error - Raised index of sequence is out of bonds

Key Error -Raised if non existent key require for set or dictionary

Key board interrupt- raised if user type

Ctrl- c while program is executing

Name error – raised if non existent identifier used

Stop Iteration-Raised by next (iterator) if no element See Section 18

Type Error -Raised when wrong type of parameter sent to a function.

Value Error - Raised when parameter has Invalid ex: sqrt(-5)

zero division Enar- Raised when thy division operator used with O as divisor.

if True:

    print("This is ")

try:

    print('a'+10)

    print(a)

except Exception as uma:

    print("Error occured")

try:

    print('a'+10)

except:

    print("outer")

    try:

        print(a)

    except:

        print("inner")

* if True:

    print("this is")

* s=str(1)+"d"

print(s)

* d={1:"a"}

print(d[2])

* c=[1,2,3,535,5]

print(c[100])

* while True:

  print("loooop")

* d=[1,424,3,

print(d)

* f=open("kiran.txt",mode="r")
* import gamalkadijarumitayaa
* import os
* c=int(input())

print(c)

print(1/0)

* i=1000000

i.append(100)

* def func(a):

    print(a)

func(1,2)