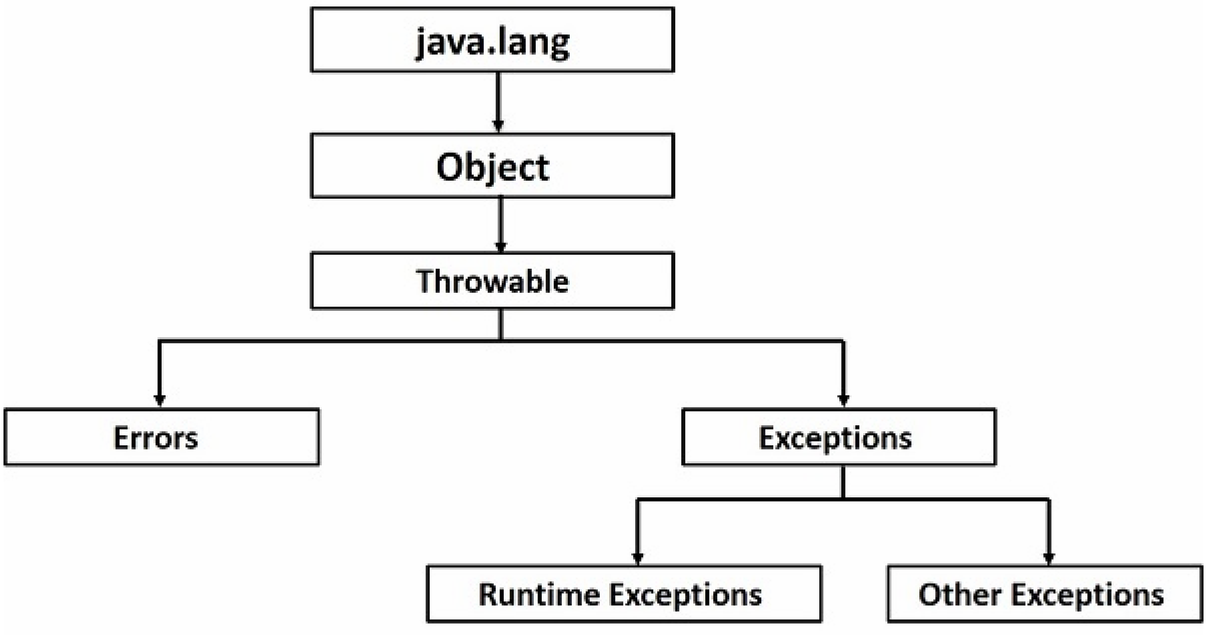
1. Exception Handling
    a. Unwanted interruption that stops the normal execution of program



2. Types of Exception
    a. Checked (compile time)
        i. Occurs at compile time
        ii. Sub class of Exception other than RuntimeException
        iii. Programmer must handle this exception
        iv. Ex: - ClassNotFoundException, InteruptedException, IOException
    b. Un-Checked (run-time)
        i. Occurs at runtime
        ii. RuntimeException and its subclasses
        iii. May or not handle this exception It will stop the execution when exception occurs
        iv. Ex: - ArithmeticException, IndexOutOfBoundException, NullPointerException, ClassCastException
3. Exception Handlers
    a. try & catch

i. try block used to declare exception and catch those exception in catch block

ii. try with multiple catch block

- used to catch sequential exception
- multiple catch can have declaration according to exception hierarchy (child class first then the super class next and so on ….)

iii. nested try catch

b. finally

i. Always gets executed

ii. Used to run block of code irrespective of exception ex:- file close, connection close.

iii. Assignment:- Can try with finally (without catch) can be used. Or Catch with finally (without try) be used.

c. throws

i. Used to handle only checked exception (throw the exception)

ii. It does not handle the exception, it just passes the exception to calling method.

iii. While overriding:-

- May or may not throw same exception
- Child class method cannot throw new Checked exception.
- Cannot throw broader or newer checked exception

d. throw

i. Used to re-throw the exception.

ii. Used generally for un-checked exceptions

iii. Used to work with runtime or user defined exception.

iv. Must be last statement inside that block of code.

4. User defined Exception

a. Used to throw the exception message.

b. Must be subclass of Exception. Preferred one is subclass of RunTimeException.