**Assignment – 3**

1. An exception is an event that disrupts the normal flow of the program. It is an object which is thrown at runtime.
2. The keywords are **try, catch, throw, throws** and **finally.**
3. (Image included in GitHub repository)
4. If it is necessary to perform different tasks at the occurrence of different Exceptions, use java multi catch block. At a time only one Exception occurs – and at a time only one catch block is executed. All catch blocks must be ordered from most specific to most general i.e. catch for **ArithmeticException** must come before catch for **Exception**.
5. The classes that extend **Throwable** class except **RuntimeException** and **Error** are known as Checked Exceptions. They are checked at Compile Time. The classes that extend **RuntimeException** are known as Unchecked Exceptions. They are checked at Run Time.
6. The **throw** keyword is used to explicitly throw an exception, is used within a method and is followed by an instance. The **throws** keyword is used to declare an exception, is used with a method signature and is followed by a class.
7. To create a custom exception, use the **Exception** keyword to extend a custom class and define the custom exception inside it.
8. The **OutOfMemoryError** is thrown when the Java Virtual Machine cannot allocate an object because it is out of memory, and no more memory could be made available by the garbage collector.
9. Some of the common main thread exception scenarios are:
   1. Exception in thread main **java.lang.UnsupportedClassVersionError**: This exception comes when your java class is compiled from another JDK version and you are trying to run it from another java version.
   2. Exception in thread main **java.lang.NoClassDefFoundError**: There are two variants of this exception. The first one is where you provide the class full name with .class extension. The second scenario is when Class is not found.
   3. Exception in thread main **java.lang.NoSuchMethodError**: main: This exception comes when you are trying to run a class that doesn’t have main method.
   4. Exception in thread “main” **java.lang.ArithmeticException**: Whenever any exception is thrown from main method, it prints the exception is console. The first part explains that exception is thrown from main method, second part prints the exception class name and then after a colon, it prints the exception message.
10. The differences are –
    1. Final is used to apply restrictions on class, method and variable. Final class can't be inherited, final method can't be overridden and final variable value can't be changed.
    2. Finally is used to place important code, it will be executed whether exception is handled or not.
    3. Finalize is used to perform clean up processing just before object is garbage collected.
11. The main method should simply terminate if any exception occurs.
12. Yes, but it’s best avoided.
13. Some of the good practices are –
    1. Clean up Resources in a Finally Block or Use a Try-With-Resource Statement
    2. Use a Finally Block
    3. Prefer Specific Exceptions
    4. Throw Exceptions with Descriptive Messages
    5. Catch the most specific exception first