**A nested class violates the recommendation “low coupling” in class design. Why are nested classes used?**

* It is a way of logically grouping classes that are only used in one place.
* It increases encapsulation.
* Nested classes can lead to more readable and maintainable code.

**INHERITANCE**

A class can be directly derived from only one class ( **Java is a single-inherited OOP language**).

If a class does not have any superclass, then it is implicitly derived from Object class.

constructor cannot be inherited ( constructor of super class can not initialize sub-class objects)

**NUMBERS CLASSES**

We use a Number object rather than a primitive when:

* + As an argument of a method that expects an object.
  + To use constants defined by the class, such as MIN\_VALUE and MAX\_VALUE.
  + To use class methods for converting values to and from other primitive types.

Java’s **StringBuffer** and **StringBuilder** classes represent strings that can be dynamically modified.

StringBuffer is threadsafe.

StringBuilder (introduced in 5.0) is not threadsafe.

* Major difference: string builders are not threadsafe.
* If you want multiple threads to have concurrent access to a mutable string, use a string buffer.
* If your mutable string will be accessed only by a single thread, there is an advantage to using a string builder, which will generally execute faster than a string buffer.