**2014\_JavaTheCompleteReference\_HerbertSchildt – Chapter 7**

**Understanding Static**

* There will be times when you will want to define a class member that will be used independently of any object o that class.
* Normally, a class member must be accessed only in conjunction with an object of its class.
* However, it is possible to create a member that can be used by itself, without reference to a specific instance.
* To create such a member, precede its declaration with the keyword **static**.
* When a member is declared **static**, it can be accessed before any object of its class are created and without reference to any object.
* You can declare both methods and variables do be **static**.
* The most common example of a static member is main(), which is declared as static because it must be called before any object exist.
* Instance variables declared as **static** are, essentially, global variables.
* When objects of it class are declared, no copy of a **static** variable is made.
* Instead, all instances of the class share the same **static** variable.
* Methods declared as static have several restrictions:
  + They can only directly call other static methods.
  + They can only directly access static data.
  + They cannot refer to **this** or **super** in any way.
* If you need to do computation in order to initialize your **static** variables, you can declare a **static block** that executed exactly once, when the class is first loaded.
* The following example shows a class that has a static method, some static variables and a static initialization block.