Java Variables

Description

- Two types of access Specifications are available in java to decide the scope of a variable, method, constructor or class. They are:

  - **access modifiers** and
  - **non-access modifiers**.

Below is the list of access modifiers and non-access modifiers in Java.

Access Modifiers	Non-Access Modifiers
private	static
default or No Modifier	final
protected	abstract
public	synchronized
	transient
	volatile
	strictfp

Fig1. Access / non-access modifiers

Access Modifiers:

- Access Modifiers are of four types:

1. private
2. default
3. protected
4. public

Access Modifiers for Class

- public: Available to the whole Java world.
- default: Available only within the specific file only.

Access Specifications for Instance & Static Variables:

- default: Available only to the classes that defines it. Can be accessed via Inheritance or Direct access by other class or method in the same file.
- public: Available to the whole Java world. Can be accessed via Inheritance or Direct access.
- protected: Available only for its sub classes that inherit / extend the specific class.
- private: Available only inside its defining class.

Access Modifiers for Methods

- default: Available to the classes that are created in the same package. Can be accessed through Inheritance or Direct access.
- public: Available to the whole Java world. Can be accessed through Inheritance or Direct access.
- protected: Available only for its sub classes that inherit / extend the specific class.
- private: Available only inside its defining class.

Access Modifier for Local Variable

- There are no specific access specifications for local variables other than the keyword "final".

Non-Access Modifiers:

- Non-access modifiers never change the availability levels of variables or methods.
- They supply certain special properties.
- Discussion follows on following types of non-access specifications:

  - **Final**
  - **Static**
  - **Transient**
  - **Synchronized**
  - **Volatile**

Final:

- Proclaim a field as final and hence avoids from being changed.
- Requires initialization at the time of declaration.

Static:

- Constructs variables and methods for a java class that are available with no class instance.

Transient:

- Value of transient instance variable indicates java virtual machine that it will not be persevered when an object is serialized.

Synchronized:

- A synchronized method can be contacted by only one thread at a time.

Volatile:

- Informs the java compiler that it can be modified without prior notice to other parts the java coding. Utilized as part of multithreading concepts in java programs.

