Understanding of Basic Java

KEY to write a code:

* **data type & variable**
* **2 type of data type:** 
  + - * **Primitive:** main data types and 3 important (String, int, Boolean)
      * **Reference:** used / created for Methods, class, Constructor … etc
* **Data type:** used to store different types of data that helps us solve problems
* **Integer (INT):** when entering numbers
* **String (String):** when entering text
* **Boolean (Boolean):** only true / false
* **Double (double):** entering number with decimal numbers
* **Conditional statement**
* **Conditional statements:** Used when we want to execute some lines. True / false statement; if true – java will print; false – it will go to the second line. Used when we want to execute some lines. True / false statement; if true – java will print; false – it will go to the second line
* **If, Else :** Example of conditional statement
* **Data structure & flow / Functions**
* **Array:** can store multiple number / String.
* **Loop:** something that repeats constantly
* **For loop:** 3 conditions; 1st – where to start with; 2nd how long will it run; 3rd increment / decrements **– {**int I= 0; i < 5; i+1**}**
* **while loop:**
* **Array / list:**
* **Object oriented concept (OOPs)**
* **Method:** block of code / storage of data / collection of statements that are grouped together ready to perform. AKA – function / work
* **Return type:** method may return a value
* **Parameter:** list of data that can be stored inside method parenthesis
* **Argument:** in Main static, we can call the parameters that are
* stored. This is called passing parameters.
* **Constructor:** used to set up some initial value for some properties of an object. Whenever an objects created constructor is executed.
* **Class:** blueprint or template of an object
* **Object**: it is a copy of a specific class. Has 3 characters. (*Identity, State, Behavior*)
* **properties:**
* **Method (Function/ work ):** displays the objects behaviors
* **Encapsulation:** Goes hand to hand with Access Modifiers. Used of to protect sensitive data *{ Private, Public, Default, Protected* }
* **Inheritance**: Creating Main class (*parent*) and extending to other classes (*child*). Mainly it is used to reduce the line of coding.
  + - **Override:** Authorizing to reject or cancel; child class can override a method and change
    - **Overloading:** You can add parameters in child class that are being inherited by parent class *( passing parameters)*
* **Polymorphism:** The task that performs a single action in different ways;
* **Abstraction:** A process of hiding the implementation of the class
* **Inside of Class:** 
  + **Public class**: a temple or blueprint. Everything will be executed in curly brackets
  + **Public static Main void Method:** your java program will compile and run successfully.
  + **Static void:** A block of code that will be executed before the main method / printout page. You do not need to create a method to print in main methods. Since they both static.
  + **Access Modifier:** ( Control of class members)
    - **PRIVATE**: the code is accessible within the same class
    - **PUBLIC**: the code is accessible within all the class throughout the projects
    - **DEFUALT**: the code is accessible within the same package
    - **PROTECTED**: only child class have accessible to the code by extending to parent class (**Inheritance**)

**Automation:**

* **Automation:** testing the app automatically
* **Selenium:** testing tool used to test web applications across different browsers.
* **WebDriver**: Driver that navigate
* **DOM (document object model):** All the codes in the front-end application stored in DOM.
* **HTML :** hyper Text Markup Language
  + **DOM (**