# ASSIGNMENT -6

**Hemavarshini D**

**1. What is private access specifier?**

Private members of a class cannot be accessed from outside the class. They are accessible only within the class.

A class cannot be private except for inner classes. Inner classes are members of the outer class. So, members of the class can be private.

**2. What are getter and setter methods? why do we need them?**

Mutator methods are the java methods, which are used to set/modify data in Objects.

Accessor methods are the java methods, which are used to get/access data from Objects.

Getter and Setter are methods used to protect the data and make our code more secure.

**3. Why this keyword in the setter method?**

Inside setter methods we use "this" keyword to access the class property, **because** usually the given parameter name is the same as the class variable so 'this' is used to identify which of them belong to the class and which one was passed to the method.

**4. Difference between localvariable and member variable/instance variable.**

An **instance variable** is a variable that is declared in a class but outside a method while the **local variable** is a variable declared within a method or a constructor.

An **instance variable** is a variable that is bound to the object itself while the **local variable** is a variable that is typically used in a method or a constructor.

It is possible to use access modifiers for **instance variables**, whereas it is not possible to use access modifiers for the **local variables**.

An **instance variables** can have default values, **local variables** do not have default values.

An **instance variables** are created when creating an object, whereas **local variables** are created when entering the method or a constructor.

**5. What is reference variable?**

A variable that holds reference of an object is called a reference variable. Variable is a name that is used to hold a value of any type during program execution.

**Naming convention ==>** same as class name but starts with lower camel case**.**

**6. Syntax of creating an object?**

**Syntax :** ClassName referenceVariable = new ClassName();

**7. Explain in detail what happens when we create an object?**

When the statement College myCollege = new College(); will be executed by JVM, an object will be created in the heap memory and stores the given data “Hello, hema” in it.

The address of the object is stored in the reference variable myCollege in the stack memory.

An object reference is a unique hexadecimal number that represents a memory address of the object. It is useful to access members of objects.

When a new object is created, a new reference number is allocated to it. It means that every object in Java will have a unique reference.

**8. What is class?**

A class is a model to create objects. It means that we write properties and actions of objects in the class.

Properties are represented by variables, and actions are represented by methods. So, a class consists of variables and methods.

**9. What is object?**

An entity that has state and behavior is known as object in Java.

Objects in Java consists of states or attributes (called data members) and behavior (called methods).

**10. What are the default values of all the datatypes?**

| Data Type | Default Value | Default size |
|---|---|---|
| byte | 0 | 1 byte |
| short | 0 | 2 bytes |
| int | 0 | 4 bytes |
| long | 0L | 8 bytes |
| float | 0.0f | 4 bytes |
| double | 0.0d | 8 bytes |
| boolean | false | 1 bit |

char ‘\u0000’ 2 bytes

**11. Difference between the static methods and instance method?**

**Static methods** can be called without the object of the class, **Instance methods** require an object of the class.

**Static methods** are associated with the class, **Instance methods** are associated with the objects.

**Static methods** can only access static attributes of the class, **Instance methods** can access all the attributes of the class.

A **static method** is declared with the static keyword, **Instance methods** do not require any keyword.

The **static method** will exist as a single copy for a class, The **instance method** exists as multiple copies depending on the number of instances of the class.

**12. Syntax of accessing the member variable in the main?**

ReferenceVariable followed by (.) dot operator and VariableName

**13.Syntax of instance method defination?**

<returntype> <method_name> ([list of parameters]) {

// Body of the method

}

**14. Syntax of static method defination?**

<accessModifier> **static <**returnType> <method_name>(Parameters){

// Body of the method

}

**15. Difference between actual parameter and formal parameter?**

The **Actual parameters** are the values that are passed to the function when it is invoked. The **Formal Parameters** are the variables defined by the function that receives values when the function is called.

The **actual parameters** are passed by the calling function. The **formal parameters** are in the called function.

In **actual parameters**, there is no mentioning of data types. Only the value is mentioned. In **formal parameters**, the data types of the receiving values should be included.

**16.Why we need the parameter or arguments to the methods?**

The parameters are used in the method body and at runtime will take on the values of the arguments that are passed in.

Parameters refers to the list of variables in a method declaration. Arguments are the actual values that are passed in when the method is invoked.

**17.Why we need the return statement and return type to the method.**

Return statement used to return the value from a method and the flow of program execution comes out of it goes back to the caller method.

Return type returns a value of expected data type from the method

**18. Method can be private.( true or false)**

True

**19. What is the error message we get if we access private variable or method outside the class?**

The filed Class.variable is not visible;

Example:The field Employee mobileNumber is not visible;