Abhinav Raj Srivastava.
Java 2.0 hindenglish.

Assignment-8. (constructor)

Q-1 what is capsulation?

(a) In Java, constructor is basically a special way of calling of a method while creating an object of a class.

(b) The constructor method have it's name of it's class's names and it has no return type not even void.

Example -

```
public Student {
    private name;
    private int age;
    public Student (name, age) {
        this.name = name;
        this.age  = age;
    }
}
public static void main (String[] args) {
    Student S1 = new Student (Ram, 21);
    Student S2 = new Student (Raju, 20);
}
```

Q-2 What is constructor chaining?

Constructor chaining refers to the process of one constructor calling another constructor within the same class or in its Superclass.

Q-3 Can we call a Subclass constructor from a Superclass constructor?

No, it is not possible to call a Subclass constructor from a Superclass constructor in Java.

Q-4 What happens if you keep a return type from a constructor?

In Java constructor do not have a return type so if we keep some return type with it, we will get a compilation error.

Ex- constructor Student in class construct. Student cannot be applied to the given type.

Q-5 What is No-argument constructor?

(a) A no argument constructor is basically a constructor without any argument from the calling source.

(b) It do not get any information from the calling point of the class.

```
Ex- class Student {

    // name
      private String name;

      public Student ( ) {

    System.out.println (" No argument
                        constructor");
    }

Static public void main ( String [ ] args ) {

      Student S1 = new Student ( );

    }.
```

Q-6 How no-argument constructor is different from default constructor?

(a) A default constructor is automatically generated by the Java compiler if no explicit constructor are defined in a class and we can not set or initialize value using default constructor in the class.

(b) A no-argument constructor is explicitly defined in the code and can be used to initialize an object's state when no argument are provided during object creation.

Q-7 When do we need constructor overloading?

(i) Constructor overloading is a method or way of creating multiple Constructor of different parameter of a single same name.

(ii) We need constructor overloading when we are given more than one object of same class having different parameter.

Q-8 What is a default constructor?

* In Java whenever we create an object of a class and do not create any constructor explicitly then behind the scene, Java compiler create a default constructor of that object inside the class of object having zero receiving parameter as well as a empty body.

Example-

```
class Student {
    private String name;

    public Student() {
        // empty body
    }

    public static void main(String[] args) {
        Student S1 = new Student();
    }
}
```

* Default constructor

Assignment - 9 (Encapsulation)

Q-1 What is Encapsulation in Java? why it is called Data hiding?

* Encapsulation - In Java Encapsulation means to making Data hiding and Data Abstraction of a class from the outside class.

So that noonce can access or manipulate the directly.

Also, Building of Data and corresponding Method into a single unit is called Encapsulation

* Data Hiding - Using Encapsulation method we are able to make our data inside a class private, So that outside the class, no method or way can access the Data or manipulate directly.

Q-2 Important features of Encapsulation -

(a) Data is private - We make the Data private in Encapsulation so that no one outside the class or unit can access the Information Directly.

(b) public method - We can make public as well as private both the during making a method in the class.

(c) Security increase - Encapsulation make our software more secure and trustable as the Data access directly from outside the class is not possible.

Q-3 What are the getter and Setter method in Java Explain?

i. Getter() - Getter method is used to get some Data or Information in order to initialize the Data with the varia

ii - Setter() - Setter methods are use to set value of the instance varia through the parameter it receive from method call.

Q-4 use of this Keyword give some Example.

this Keyword- In Java Keyword 'this' is used at the time of Constructor creation and Initialization of some data to the instance variables of any particular object.

Example-
```
class Student {
    private String name;
    private int age;
    public Student (name, age) {
        this.name = name;
        this.age = age;
    }
}
public static void main (String [] args) {
    Student S1 = new Student (Ram, 20);
    Student S2 = new Student (Raju, 21);
}
```

Q-5 advantage of Encapsulation ?

* Data privacy.
* Data hiding
* Data abstraction
* Access control
* Code flexibility.
* Code Reusability.
* Security.
* Evolution of compatibility

Q-6. How to achive Encapsulation in Java ? Give an Example ?

Ex-
```
public class fruits {
    Private String name;
    Private int price;
    public fruits (name, price) {
        this.name = name;
        this.price = price;
    }
} public Example {
public static void main (String[] args) {
    fruits f1 = new fruits (Apple, 100);
}
```

Assignment-9 : (practice questions).

Q-1 create a class that keeps track of the number of instance created. Implement a static variable and method to accomplish this.

```
public ~~static~~ Student {

    static String name;
    static int age;

    public static Result ( ) {

        int math = 90;
        int Bio = 100;
        S.o.p ( math );
        S.o.p ( Bio );
    }
}

public Demo {
public static void main ( String[] args) {

    Student.name = "Hari";
    Student.age = 20;
    Student.Result ( );
}
}
```

Q-2 : write a program and a constructor with parameter and initialise the variable using the constructor.

```
public Student {

    private String name;
    private int age;

public Student ( String name, int age ) {

    this.name = name;
    this.age = age;

System.out.println (" your name is " +name+ " and
                    you are " +age+ " years old");
}
}

public class Demo {
public static void main (String[] args) {

Student S1 = new Student (Ram, 20);
Student S2 = new Student (Sham, 10);
```

// output :

your name is Ram and you are 20 years old.
your name is Sham and you are 10 years old.

Q-3 use a private keyword for a variable and use setter and getter method to initialise and print the values.

```
public class Student {

    private String name;
    private int age;

public    Setter ( String name, int age ) {

        this.name  = name;
        this.age   = age;
    }

public   getter ( ) {

    System.out.println ( name );
    System.out.println ( age );

    }
}

    public  class Demo {

        public static void main ( String [ ] args ) {

Student. Setter ( "Abhinav", 21 );
Student . getter ( );
}
```

Q-4. write a program to call an method without creating an object of a class.

```
public class Student {

Static  public  Student ( String name ) {

        System.out.println ( "Hello " + name +
                        " it is a constructor call");
    }
}

public class Demo {

public static void main ( String [ ] args ) {

    Student. Student ( "Ram" );

}
}
```

// output

Hello Ram it is a constructor call.

Q-5 write a program which has static block and constructor overloading, initialise variable using constructor and print it.

part-i

```
public class Student {
    private String name;
/* i). */  static {
        System.out.println("This is a
                static block");
    }

/* ii). */  void Student1 (String name) {
        this.name = name;
        System.out.println(name);
    }

    void Student1 (String name, int age) {
        System.out.println(name);
        System.out.println(age);
    }
}
```

```
public class Demo {
public static void main (String[] args){
    Student1 S1 = new Student1(Ram);
    Student1 S2 = new Student1(Ram, 20);
}
}
```

// output

Ram
Ram 21.

Assignment - 10 (Static).

Q-1 Why do we need Static Keywords in Java?

(a) In Java, the 'static' keyword is used to define class-level members that are shared among all instances of class.

(b) It can be applied to variable, method, and nested class.

(c) • variable, methods or blocks created using static keyword, are loaded onto the memory when the class of it loaded on the memory

(d) • It is created once and till the end of the program execution it will be inside the memory.

• main Reasons for using 'static' Keyword in Java.

1. Static Variables.
2. Static method
4. Static BLOCK

Q-2 Class loading and how does the Java program actually executes?

Class loading in Java is a process by which the JVM load the classes and interfaces into memory during the execution of a Java program.

The class loading process consists of three steps —

(a) • loading — the JVM's class loader locates and read the byte code of a class or interface.

• The class loader then creates an internal representation of the class called a "class" object.

(b) linking — linking is divided into three sub-categories: verification, preparation, and resolution.

(c). Initialization - this is the final step, the JVM executes the static initialization code of the class.

- Static variables are assigned their explicit initial values, and any static initializer blocks are executed in the order they appear in the source code.

Q-3. Can we mark a local variable as static?

- NO, it is not possible to mark a local variable as 'static' in Java.

- the 'static' keyword is used to define class-level members, such as variables and methods, that are shared among all the instance.

- local variable are variables declared within a method, constructor or block, and they are scoped to that specific block.

- They are created and destroyed each time the block is executed.

Shiva
Date
Page

Q-4: Why Static block executed before the main method in Java?

- When a Java program starts, the JVM first loads the class into the memory and initialize its static members, including static variables and static blocks.
- The 'main' method then executed as the entry point of the program.

Q-5: Why is a static method is also called as a class method?

A Static Method in Java is also called a class Method Because it is associated with the class itself rather than with instance of the class.

Q-6: use of Static block in Java?

i. for class level Initialization.
ii. for utilization of memory.
iii. for saving memory.
iv. Exception handling during class initialization
v. Static configuration or setup.

Q-1

	Static variable	Instance variable
(a)	they are created once and stored among all objects of the class	they are created each time of creation of any object
(b)	they're created inside the heap memory	they are also created inside a heap memory.
(c)	the life of a static variable is from starting of the program till end of the execution of program	the life of a IV are start with its object creation and end with object distruction.

Q-2.

	Static member	non-Static member.
(a)	It belongs to a class itself rather than any certain instance of class.	they are associated with individual instance of the class.
(b)	ClassName.Staticmethod();	object.nonStaticMethod();
(c)	Static member are allocated memory only once, when the class loaded into memory.	they are allocated memory for each time any instance of the class