



Victoria University
of Bangladesh

Assessment Topic:

Final Assessment

Course Title: Object-Oriented Programming

Course Code: CSI-211

Submitted To:

Umme Khadiza Tithi

Lecturer, Department of Computer Science & Engineering

Victoria University of Bangladesh

Submitted By:

Ruhul Amin

ID: 2120180051

Department: CSE

Semester: Fall-2022

Batch: 18th

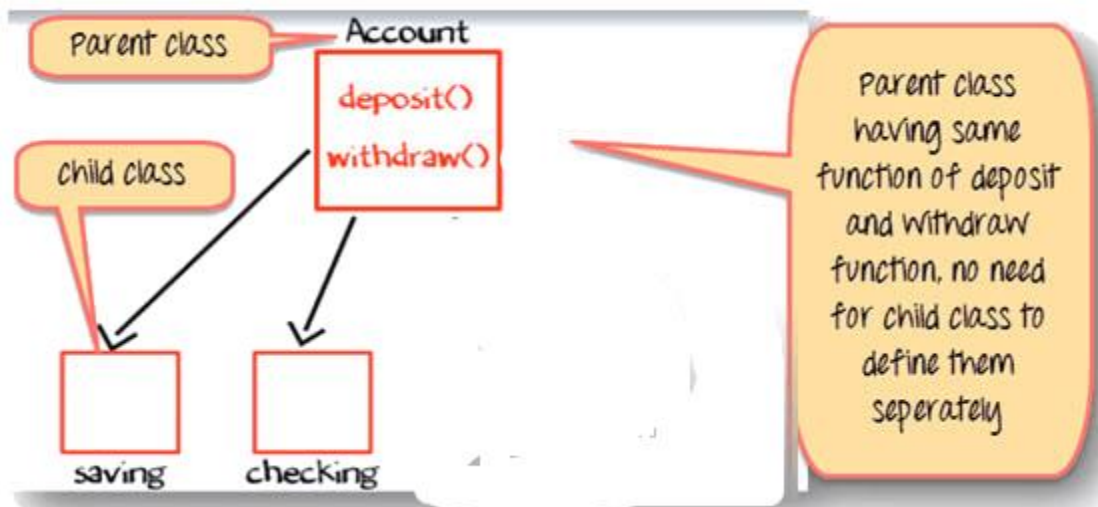
Submission Date: 05th February 2023

Answer to question no 1(a)

Define polymorphism?

Answer: Polymorphism in Java occurs when there are one or more classes or objects related to each other by inheritance. It is the ability of an object to take many forms. Inheritance lets users inherit attributes and methods, and polymorphism uses these methods to perform different tasks. So, the goal is communication, but the approach is different.

For example, you have a smartphone for communication. The communication mode you choose could be anything. It can be a call, a text message, a picture message, mail, etc. So, the goal is common that is communication, but their approach is different. This is called **Polymorphism**. Now, we will learn Polymorphism in Java



Java Polymorphism Example

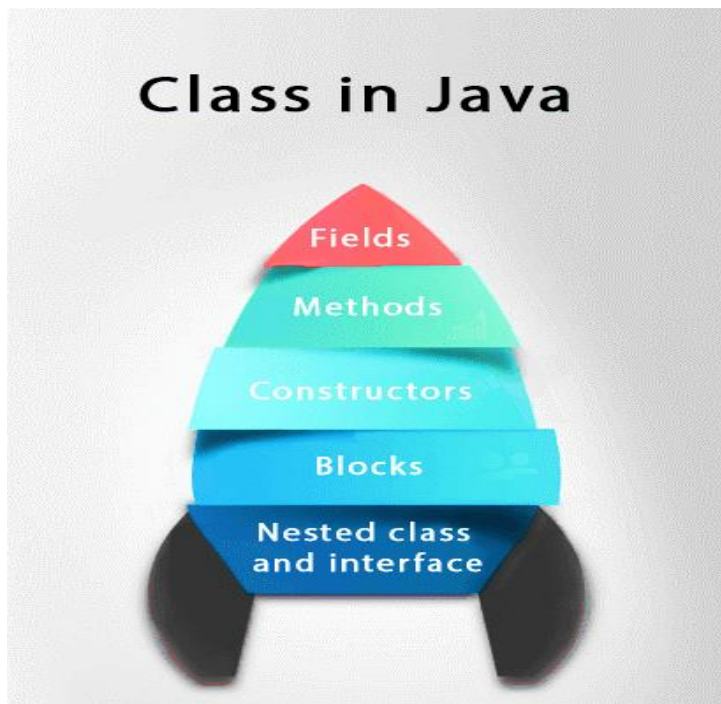
Answer to question no 1(b)

Define class and objects in java?

Answer: #Class: A class is a group of objects which have common properties. It is a template or blueprint from which objects are created. It is a logical entity. It can't be physical.

A class in Java can contain:

- **Fields**
- **Methods**
- **Constructors**
- **Blocks**
- **Nested class and interface**



Syntax to declare a class:

```
class <class_name>{  
    field;  
    method; }  

```

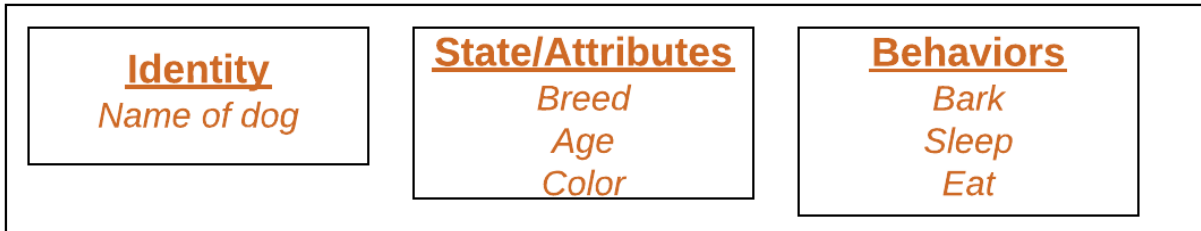
A class is a user-defined blueprint or prototype from which objects are created. It represents the set of properties or methods that are common to all objects of one type. In general, class declarations can include these components, in order:

1. **Modifiers:** A class can be public or has default access.
2. **class keyword:** class keyword is used to create a class.
3. **Class name:** The name should begin with an initial letter (capitalized by convention).
4. **Superclass (if any):** The name of the class's parent (superclass), if any, preceded by the keyword extends. A class can only extend (subclass) one parent.
5. **Interfaces(if any):** A comma-separated list of interfaces implemented by the class, if any, preceded by the keyword implements. A class can implement more than one interface.
6. **Body:** The class body surrounded by braces, { }.

#Object: It is a basic unit of Object-Oriented Programming and represents real-life entities. A typical Java program creates many objects, which as you know, interact by invoking methods. An object consists of:

1. **State:** It is represented by attributes of an object. It also reflects the properties of an object.
2. **Behavior:** It is represented by the methods of an object. It also reflects the response of an object to other objects.
3. **Identity:** It gives a unique name to an object and enables one object to interact with other objects.

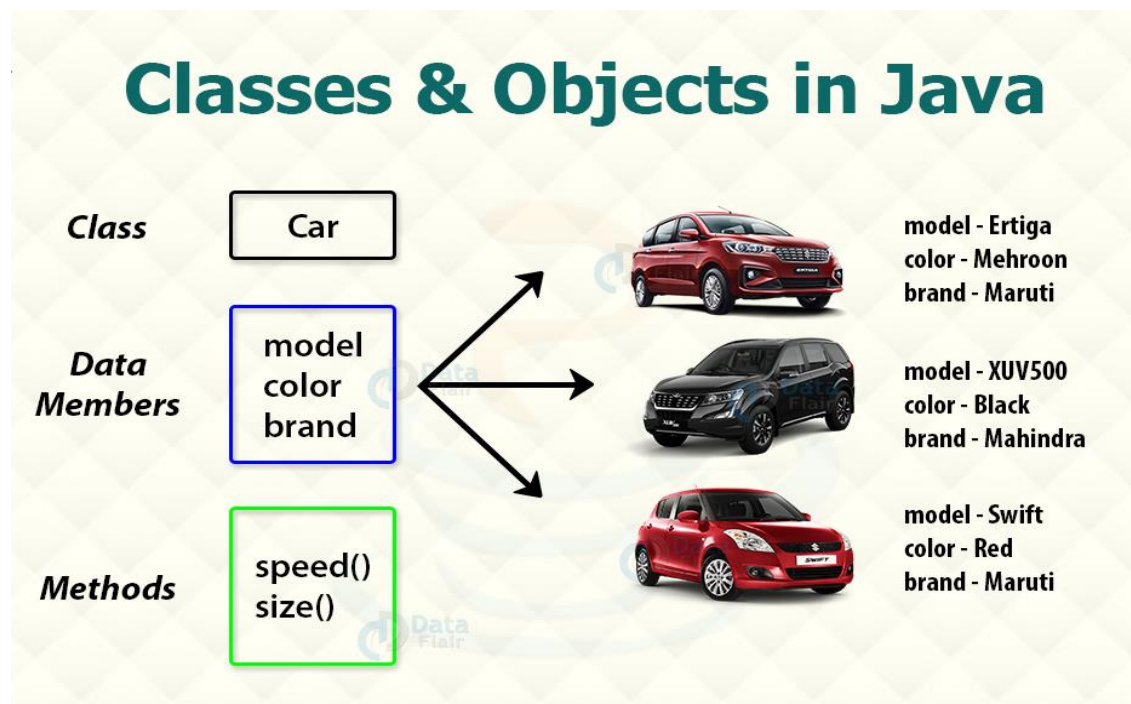
Example of an object: dog



Objects correspond to things found in the real world. For example, a graphics program may have objects such as "circle", "square", "menu". An online shopping system might have objects such as "shopping cart", "customer", and "product".

Object Definitions:

- An object is a real-world entity.
- An object is a runtime entity.
- The object is an entity which has state and behavior.
- The object is an instance of a class.



Answer to question no 1(c)

What are the Benefits of Object-Oriented Programming?

Answer: Object Oriented Programming (OOP) is a programming paradigm that relies on the concept of **classes** and **objects**. It is used to structure a software program into simple, reusable pieces of code blueprints (usually called classes), which are used to create individual instances of objects. There are many object-oriented programming languages including JavaScript, C++, Java

OOP stands for **Object-Oriented Programming**. As you can guess from its name it breaks the program on the basis of the objects in it. It mainly works on Class, Object, Polymorphism, Abstraction, Encapsulation and Inheritance. Its aim is to bind together the data and functions to operate on them.

Some of the well-known object-oriented languages are Objective C, Perl, JavaScript, Python, Modula, Ada, Simula, C++, Smalltalk and some Common Lisp Object Standard. Here we are discussing its benefits on C++.

Benefits of OOP

- We can build the programs from standard working modules that communicate with one another, rather than having to start writing the code from scratch which leads to saving of development time and higher productivity,
- OOP language allows to break the program into the bit-sized problems that can be solved easily (one object at a time).
- The new technology promises greater programmer productivity, better quality of software and lesser maintenance cost.
- OOP systems can be easily upgraded from small to large systems.
- It is possible that multiple instances of objects co-exist without any interference,

- It is very easy to partition the work in a project based on objects.
- It is possible to map the objects in problem domain to those in the program.
- The principle of data hiding helps the programmer to build secure programs which cannot be invaded by the code in other parts of the program.
- By using inheritance, we can eliminate redundant code and extend the use of existing classes.
- Message passing techniques is used for communication between objects which makes the interface descriptions with external systems much simpler.
- The data-centered design approach enables us to capture more details of model in an implementable form.
- OOP models complex things as reproducible, simple structures
- Reusable, OOP objects can be used across programs
- Allows for class-specific behavior through polymorphism
- Easier to debug, classes often contain all applicable information to them
- Secure, protects information through encapsulation

While it is possible to incorporate all these features in an OOP, their importance depends upon the type of project and preference of the programmer. This technology is still developing and current products may be superseded quickly.

Answer to question no 1(d)

Write the "print Java program" Java program in a file called love java.java using java graphics.

Answer:

```
import java.applet.*;
import java.awt.*;

public class app_demo extends Applet
{
    public void paint(Graphics g)
    {
        g.drawString("print Java program",200,200);
    }
}
```

Answer to question no 2(a)

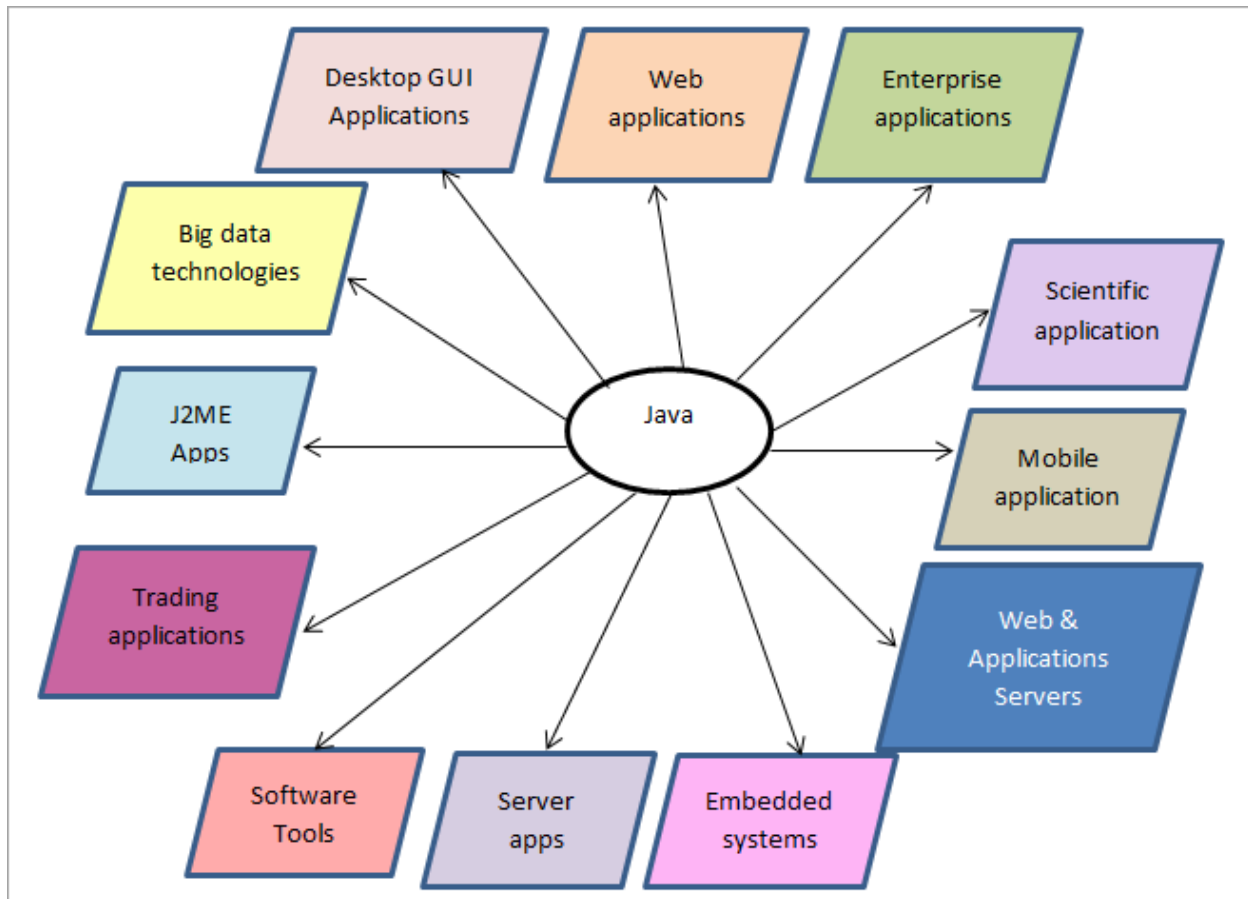
Define Java Applets?

Answer: An applet is a Java program that can be embedded into a web page. It runs inside the web browser and works on the client side. An applet is embedded in an HTML page using the APPLET or OBJECT tag and hosted on a web server.

- Applet is a special type of program that is embedded in the webpage to generate dynamic content. It runs inside the browser and works on the client side.
- Applets are used to make the website more dynamic and entertaining.

Answer to question no 2(b)**# Describe types of applications that run on java?****Answer:** Applications of Java

- #1) Desktop GUI Applications
- #2) Web Applications
- #3) Mobile Applications
- #4) Enterprise Applications
- #5) Scientific Applications
- #6) Web Servers & Applications Servers
- #7) Embedded Systems
- #8) Server Apps in Financial Industry
- #9) Software Tools
- #10) Trading Applications
- #11) J2ME Apps
- #12) Big Data Technologies



#1) Desktop GUI Applications: Java language provides a lot of features that help us to develop GUI applications. Java provides AWT, Swing API or Java Foundation Classes, or the latest JavaFX (from Java 8 onwards). These APIs/features help us develop advanced GUI applications including advanced tree-based or even 3D graphical applications.

Real-world desktop tools developed using Java:

- Acrobat Reader
- ThinkFree

#2) Web Applications

Java provides features for web development as well as Servlets, Struts, Spring, hibernate, JSPs, etc. that allow us to develop highly secured easily to program software.

Real-World Web Tools using Java:

- Amazon

- Broadleaf
- Wayfair

#3) Mobile Applications

Java language provides a feature named J2ME which is a cross-platform framework to build mobile applications that can run across Java-supported smartphones and feature phones.

One of the popular mobile operating systems Android is developed using Java-based Android SDK.

Popular Java-based mobile apps:

- Netflix
- Tinder
- Google Earth
- Uber

#4) Enterprise Applications

Java is the first choice for developing enterprise programs mainly because of its powerful features delivering high performance. Apart from performance, Java also makes applications more powerful, secure, and easily scalable.

Java language has the Java Enterprise Edition (Java EE) platform that comes with API and runtime environment features for scripting and running enterprise software, network tools, and web services.

According to Oracle, almost 97% of enterprise computers are running on Java. The higher performance and faster computing provided by Java have resulted in most enterprise applications being developed in Java.

Real-time Enterprise Applications using Java:

- Enterprise Resource Planning (ERP) systems
- Customer Resource Management (CRM) systems

#5) Scientific Applications

Java has powerful security and robustness features that make it popular for developing scientific applications. Java also provides powerful mathematical calculations that give the same results on different platforms.

Most popular Java-based scientific tool:

- Mat lab

#6) Web Servers & Applications Servers

The entire Java ecosystem has numerous applications and web servers today. Among web servers, we have Apache Tomcat, Project Jigsaw, Rimfaxe Web Server (RWS), Jo! Etc. dominates the space.

Similarly, application servers like WebSphere, JBoss, WebLogic, etc. are dominating the industry commercially.

#7) Embedded Systems

Embedded systems are low-level systems that form a part of the larger electromechanical systems. These are tiny chips, processors, etc., and are also called integrated systems.

Java can produce robust tools that can handle application exceptions efficiently and is fast too as it is better for developing low-level programs.

Embedded systems applications using Java:

- SIM cards use Java technology
- Blue-ray disc player

#8) Server Apps In Financial Industry

Financial institutions like banks, and investors need various software programs to run their day-to-day business like front and back office electronic trading systems, writing settlement and confirmation systems, data processing, etc.

Java is mostly used for these tools to write server-side applications that receive data from one server, process it, and sends the processed data to the other servers or processes.

Most of the leading financial institutions like Barclays, Citi group, Goldman Sach, etc. use Java-based software tools for their business.

#9) Software Tools

Many software tools used for development are written in Java. **For example**, IDEs like Eclipse, IntelliJ IDEA, and Net beans are all written and developed in Java.

These are also the most popular desktop GUI-based tools used today. Earlier swing and AWT were the features that are mostly used to develop Software but nowadays JavaFx has become more popular.

#10) Web Applications: Java provides support for web applications through Servlets, Struts, or JSPs. The easy programming and higher security offered by the programming language have allowed a large number of government applications for health, social security, education, and insurance to be based on Java. Java also finds applications in the development of eCommerce web applications using open-source eCommerce platforms, such as Broadleaf.

#11) J2ME Apps

Apart from iOS and android based mobile handsets, there are handsets from Nokia and Samsung that use J2ME. J2ME is also popular with products like Blu-ray, Cards, and Set-

Top Boxes, etc. The popular application WhatsApp available on Nokia is available in J2ME.

#12) Big Data Technologies

Big data is the most popular and trending topic in the software industry today. Big data deals with analyzing and systematically extracting information from complex data sets.

An open framework that is associated with big data is called Hadoop and is entirely written in Java. Features like automatic garbage collection, memory distribution, and stack provision system, Java gets an edge over other technologies. We can safely say that Java is the future of big data.

Real-time Java-based big data Technologies:

- Hadoop
- Apache HBase
- ElasticSearch
- Accumulo

Answer to question no 2(c)

What is Multithreading? Define the advantages and disadvantages of Multithreading.

Answer:

Multithreading in Java is a process of executing multiple threads simultaneously.

A thread is a lightweight sub-process, the smallest unit of processing. Multiprocessing and multithreading, both are used to achieve multitasking.

However, we use multithreading than multiprocessing because threads use a shared memory area. They don't allocate separate memory areas so saves memory, and context-switching between the threads takes less time than the process.

Java Multithreading is mostly used in games, animation, etc.

Following are some of the common advantages of multithreading:

- Enhanced performance by decreased development time
- Simplified and streamlined program coding
- Improvised GUI responsiveness
- Simultaneous and parallelized occurrence of tasks
- Better use of cache storage by utilization of resources
- Decreased cost of maintenance
- Better use of CPU resource

1) It **doesn't block the user** because threads are independent and you can perform multiple operations at the same time.

2) You **can perform many operations together, so it saves time.**

3) Threads are **independent**, so it doesn't affect other threads if an exception occurs in a single thread.

#Following are some of the common disadvantages of multithreading:

- Complex debugging and testing processes
- Overhead switching of context
- Increased potential for deadlock occurrence
- Increased difficulty level in writing a program
- Unpredictable results

Answer to question no 2(d)

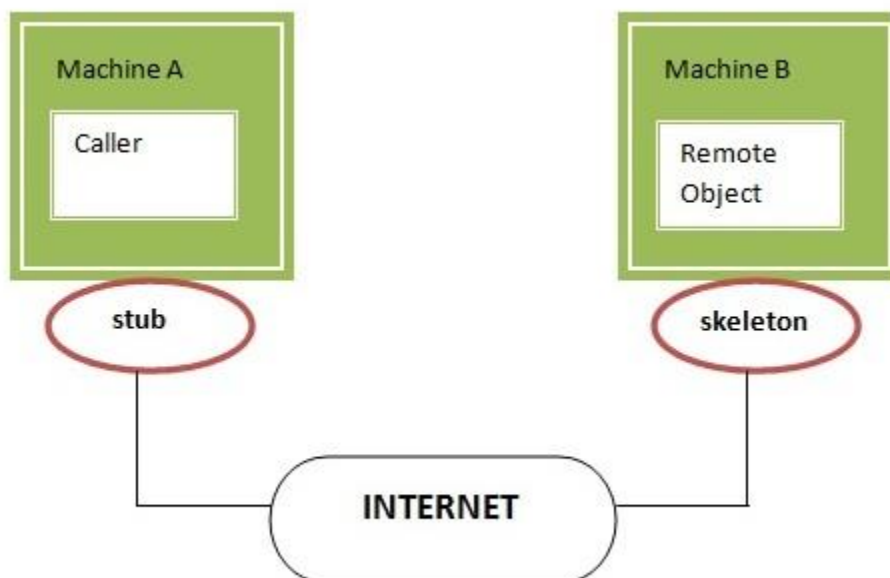
What is java RMI? Differences between stub and skeleton.

Answer:

RMI (Remote Method Invocation): The RMI (Remote Method Invocation) is an API that provides a mechanism to create a distributed application in java. The RMI allows an object to invoke methods on an object running in another JVM.

The RMI provides remote communication between the applications using two objects stub and skeleton.

In the distributed computing environment, stub stands for a client-side object participating in the distributed object communication and a skeleton stands for a server-side object participating in distributed object communication.



Differences between Stub and Skeleton:

Stub	Skeleton
<p>The stub acts as a gateway for client-side objects and all outgoing requests to server-side objects that are routed through it. The stub wraps client object functionality and by adding the network logic ensures a reliable communication channel between the client and server. The stub can be written up manually or generated automatically depending on the chosen communication protocol.</p>	<p>A skeleton act as the gateway for server-side objects and all incoming client requests are routed through it. The skeleton wraps server object functionality and exposes it to the clients, moreover adding the network logic ensures a reliable communication channel between clients and server. Skeletons are written up manually or generated automatically depending on the chosen communication protocol.</p>
<p>The stub is responsible for:</p> <ol style="list-style-type: none"> 1. initiating the communication toward the server skeleton 2. translating calls from the caller object 3. marshaling of the parameters 4. informing the skeleton that the call should be invoked 5. passing arguments to the skeleton over the network 6. unmarshalling of the response from the skeleton 	<p>The Skeleton is responsible for:</p> <ol style="list-style-type: none"> 1. translating incoming data from the stub to the correct up-calls to server objects 2. unmarshalling of the arguments from received data 3. passing arguments to server objects 4. marshaling of the returned values from server objects 5. passing values back to the client stub over the network

7. informing the skeleton that the call is complete	
---	--

Answer to question no 3(a)

What are java servlets? Advantages and disadvantages of having servlets.

Answer:

Servlet can be described in many ways, depending on the context.

- Servlet is a technology that is used to create a web application.
- Servlet is an API that provides many interfaces and classes including documentation.
- Servlet is an interface that must be implemented for creating any Servlet.
- Servlet is a class that extends the capabilities of the servers and responds to incoming requests. It can respond to any requests.
- Servlet is a web component that is deployed on the server to create a dynamic web page.

Advantages of Servlet:

There are many advantages of Servlet over CGI. The web container creates threads for handling the multiple requests to the Servlet. Threads have many benefits over the Processes such as they share a common memory area, lightweight, cost of communication between the threads are low. The advantages of Servlet are as follows:

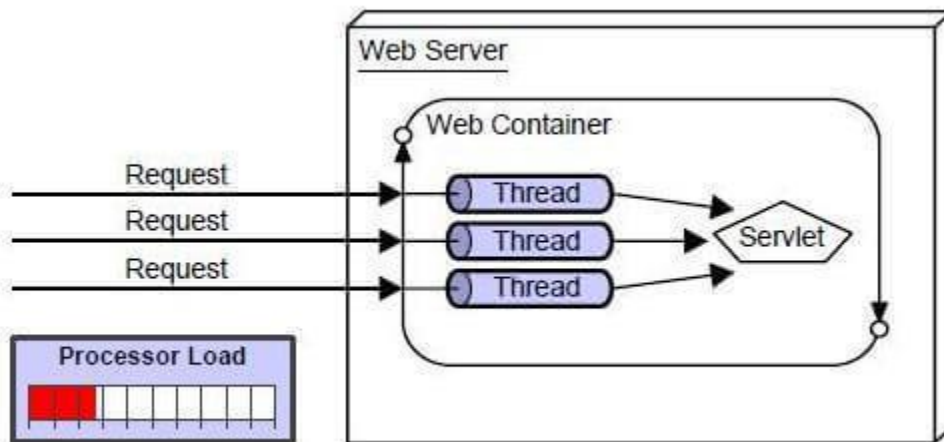
Better performance: because it creates a thread for each request, not process.

Portability: because it uses Java language.

Robust: JVM manages Servlets, so we don't need to worry about the memory leak, garbage collection, etc.

Secure: because it uses java language.

- ✓ Servlet loads only one copy of it into JVM (Java Virtual Machine).
- ✓ It saves time to respond to the first request which increases response time.
- ✓ It uses standard API that is supported by numerous web servers.
- ✓ Programming languages like platform independence and ease of development.
- ✓ You can access the large set of APIs that are available for the Java platform.
- ✓ The web container makes threads for handling more than one request to the servlet.
- ✓ Easy coordination between two or more servlet to make web applications.
- ✓ Servlet containers support many other features like sessions, resource management, persistence, security, etc.
- ✓ Servlet usage doesn't constrain by the web browser.



Disadvantages of Servlet:

Here are the disadvantages for using Servlet:

- ✓ One servlet is loaded into JVM. It does matter numbers of requests.
- ✓ When there is a request, there is a thread, not a process.
- ✓ Servlet is persistent until it destroys.
- ✓ Designing in a servlet is difficult and slows down the application.
- ✓ You need a JRE (Java Runtime Environment) on the server to run servlets.

- ✓ For non-java developers, the servlet is not suitable as they are required to have a broad knowledge of Java servlet.
- ✓ HTML code is mixed up with Java code therefore, changes done in one code can affect another code.
- ✓ Writing HTML code in servlet programming is very difficult. It also makes the servlet looks bulky.
- ✓ In servlet programming, if you want to use implicit objects, you need to write some additional code in order to access them.
- ✓ Developers must take care of exception handling because servlet programming is not thread-safe by default.

Answer to question no 3(b)

Write a Java program to check Even or Odd Numbers?

Answer:

```
import java.util.Scanner;
public class EvenOdd {
    public static void main(String[] args) {
        Scanner reader = new Scanner(System.in);
        System.out.print("Enter a number: ");
        int num = reader.nextInt();

        if(num % 2 == 0)
            System.out.println(num + " is even");
        else
            System.out.println(num + " is odd");
    }
}
```

}

Online Java Compiler IDE

For Multiple Files, Custom Library and File Read/Write, use our new - [Advanced Java IDE](#)

```
1 import java.util.Scanner;
2
3 public class EvenOdd {
4
5     public static void main(String[] args) {
6
7         Scanner reader = new Scanner(System.in);
8
9         System.out.print("Enter a number: ");
10        int num = reader.nextInt();
11
12        if(num % 2 == 0)
13            System.out.println(num + " is even");
14        else
15            System.out.println(num + " is odd");
16    }
17 }
```

Execute Mode, Version, Inputs & Arguments

JDK 17.0.1

Interactive

CommandLine Arguments:

Result

compiled and executed in 8.76 sec(s)

```
Enter a number: 13
13 is odd
|
```