

NAME: FARDOUSE LOMAT JAHAN RUMPA

ID: 2219170041

Dept: CSE (E)

Batch: 17th

Course code: CSI 211

Course Title: Object oriented programming

(Q)

Ans to the Qus NO: 01(a)

Ans: Polymorphism: Polymorphism is one of the OOPs feature that allows us to perform a single action in different ways. It is the capability of a method to do different things based on the object that it is acting upon. In other words, polymorphism allows you define one interface and have multiple implementations.

For example, in English, the verb run has a different meaning if you use it with a laptop, a ~~face~~ a foot race, and business. Here, we understand the meaning of run based on the other words used along with it. The same also applied to polymorphism.

Ans to the Qus NO: 01 (b)

Answer: Class in Java:

Classes and objects are basic concepts of object oriented programming that revolve around real life entities.

Class is a set of object which shares common characteristics behavior and common properties/attributes. -lies

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 (Refer this for details)

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 (super class), if any, preceded by the keyword `extends`. A class can only extend (sub-class) 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 is surrounded by braces `{ }`.

Constructors are used for initializing new objects.
Object in Java:

An object in Java is the most fundamental unit of the object-oriented programming paradigm. It includes the real-world entities and manipulates them by invoking methods. An object in Java consists of the following:

1. Identity.
2. Behavior.
3. State.

Identity: This is the unique name given by the user that allows it to interact with other objects in the project.

Example: "Name of the student"

Behavior: The behavior of an object is the method that you declare inside it. This method interacts with other objects present in the project.

Example: studying, playing, Writing.

State: The parameters present in an object represent its state based on the properties reflected by the parameters of other objects in the project.

Example: section, Roll number, Percentage.

Ans to the Qus NO: 01 (c)

Answer: Benefits of OOP:

There are miscellaneous benefits of OOPs we have to just focus on learning. It is a vast concept by which we can make good decisions as a developer.

1. Through inheritance, we can eliminate redundant code and extend the use of existing classes.

2. We can develop the programs from standard working modules that communicate with one another rather than having to start writing the code from abrasion which leads to saving of development time and advanced productivity.

3. The benefits of OOPs language allow it to break the program into bit-sized problems that can be answered fluently (one object at a time)

4. The new technology promises higher productivity, a better quality of software, and lower conversation cost.

5. The benefits of OOPs systems can be fluently

upgraded from small to large systems.

5. ~~The benefits of OOPs systems can be fluently.~~

5. It's possible that multiple cases of objects co-occur without any embarrassment.

6. It's actually easy to partition the work into a design grounded on objects.

7. It's possible to frame the objects in the problem area to those in the program.

8. Data hiding's principle is to help the programmer to make secure programs that can not be raided by the code in other corridors of the program.

9. By using legacy, we can exclude extra programmes and extends the use of being classes.

10. Communication passing methods are used for communication between objects which makes the interface descriptions with external systems much simpler.

11. The data-centered design approach enables us to capture further details of the model in an implementable form.

12. It's possible to map objects in the problem domain to those objects in the program.

Ans to the Qus no: 2 (a)

Ans: Applets:

Applet is a special type of program that is embedded in the webpage to generate the dynamic content. It runs inside the browser and works at client side.

An Applet is a Java program that can be embedded into a web page. It runs inside the web browser and works at client side.

Applets are used to make the website more dynamic and entertaining.

Ans to the Qus no: 02 (b)

Ans: We know that Java is the "king of all programming languages". Since its foundation, this language has become a backbone for billions of devices and applications.

This language is continually ranked first in the rankings of software developers as the best choice of programming languages. Java is used in the majority of applications, from mobile phones to enterprise servers and computing platforms.

You would be wondering why Java is so popular and where it is exactly used. In this article by Techkidvan, we will learn about the major application areas of Java.

Below is the Java applications List:

- Desktop GUI applications.
- Mobile Applications.
- Enterprise Applications.
- Scientific Applications.
- Web-based applications.
- Embedded Systems.
- Big-Data Technologies.
- Distributed Applications.
- Cloud-based applications.
- Web servers and application servers.
- Software tools.
- Gaming Applications.

Applications of Java programming Language:

① Desktop GUI Applications of Java:

Desktop applications can be easily developed using Java. We use APIs like AWT, Swing, JavaFX to build these applications. AWT (Abstract Windowing Toolkit) is an interface used to develop window-based applications in Java. It is "not totally Java-based" as it uses window user interface functionalities such as a menu, button, list, etc.

Swing is a GUI widget toolkit, which uses AWT and provides certain advanced components like trees, tables, scroll panes, tabbed panels and lists. Swing is "totally Java-based" and uses swing packages of Java to develop applications. It responds to all the mouse-click-events, key-entries etc.

JavaFX is a Modern way to develop a

desktop applications in Java, it is graph-based and totally different from AWT and Swings.

Examples of desktop GUI applications are Acrobat Reader, Thinkfree, Media Player, Antivirus, etc.

② Mobile Applications of Java :

A mobile application is an application created for mobile phones and tablets. In today's era, the majority of phones and smart devices have Android OS and Android development is not possible without Java.

Java platform, Micro Edition (Java ME or J2ME) is a cross-platform framework to build applications that run across all Java supported devices including feature phones and smart phones. Further, applications for Android, one of the most popular mobile operating systems, are usually scripted in Java using the Android Software Development Kit (SDK) or other environments.

③ Enterprise applications of Java :

An enterprise application is a large software system which operates in a corporate environment, to satisfy the needs of an organization, rather than of individual users. Java becomes the first choice for the development of enterprise applications because of its robust features that match the requirements for the same. In today's era, most of the enterprise organizations are based on the applications of Java only because it is the most secure, powerful, scalable language.

Examples of enterprise are Business corporations, schools, banks, ERP (Enterprise Resource planning) and

CRM (Customer Resource Management) systems, clubs, charities, governments, interest-based user groups etc.

④ Scientific Applications of Java:

A scientific application is an application that affects real-world activities using mathematics. Java supports the development of scientific applications, because of its powerful features.

→ Java becomes the best choice for writing scientific applications involving scientific calculations and mathematical operations. It provides a fast, secure and highly portable environment to these applications, which is the basic requirements for these applications.

→ MATLAB (Mathematical Laboratory) which is one of the most popular scientific applications, uses Java for developing both front-end (interactive user-interface) and back-end (a core part of the system).

Examples of scientific applications are applications related to research, science, medical science, space, aeronautics etc.

⑤ Web-based applications of Java:

A web-application is a client-server program that is delivered on the Internet through a browser interface.

→ E-commerce web applications also use Java with the help of open-source e-commerce platforms, such as Broadleaf.

→ Java provides easy coding and high security which enables the development of a large number of applications.

for health, social security, education and insurance.

Examples of web-based applications are icode.co.in online forms, shopping carts, Gmail, Google sheets, Google Slides and many more.

⑥ Embedded Systems:

Embedded Systems are present everywhere. It is a combination of many small computing units that assemble together to perform dedicated functions for the larger systems.

Example, a motor system, entertainment and multimedia in a car, mobile computing and Networking use an embedded System.

⑦ Big-data technologies:

→ An open-source framework, called Hadoop, associated with big data, is written in Java. Moreover, the Automatic Garbage Collection and strong memory management give it higher priority over the other programming languages.

→ Many prominent big data technologies like Apache Hadoop, Apache Spark, Apache Mahout, etc are the sub-projects of Java.

→ Hadoop and other big data technologies are also using Java in one way or the other. For example, Apache's Java-based HBase and Accumulo (Open-Source), and Elasticsearch as well.

⑧ Distributed Applications of Java:

→ Distributed applications is an application or software that executes or runs on multiple computers within a network.

→ RMI (Remote Procedure Invocation) and CORBA (Common

Object Request Broker Architecture) are the APIs to develop distributed applications.

⑨ Cloud-Based Applications of Java:

→ Java has long been the programming language that provides a structure for web-applications, and now it has reached cloud applications, because of its distributed nature.

→ Java provides us with features that can help us build applications used in SaaS (Software-as-a-Service), IaaS (Infrastructure-as-a-Service) and PaaS (Platform-as-a-Service) development.

⑩ Web Servers and Application Server:

→ Java ecosystem contains multiple Java web-servers and application servers.

→ Java provides web-servers including Apache Tomcat, Simple ; J0!, Rinfaxe Web-server (RWS) Apache HTTP Server, Resin, Adobe JRun and Project Jigsaw.

→ WebLogic, Glass-fish, Wildfly, Web-Sphere and JBoss EAP occupy commercial application server space.

⑪ Software Tools:

A software tool is a set of computer programs that developers use to develop, analyze, maintain, debug, or support other applications and programs. Many developers use Java to write and develop useful software tools. Examples of software tools are Eclipse, IntelliJ Idea, and NetBeans IDE.

⑫ Gaming Applications:

→ Java supports JMonkey Engine which is the most powerful open-source 3D-Engine and has the capacity to design 3-Dimensional games.

Ans to the Qus NO 02 (c)Answer: Multi-threading:

Multi-threading is a Java feature that allows concurrent execution of two or more parts of a program for maximum utilization of CPU. Each part of such program is called a thread. So, threads are light-weight processes within a process.

Threads can be created by using two mechanisms

- Extending the thread class
- Implementing the Runnable interface.

Common advantages and disadvantages of Multi-threading in Java:

Following are some of the common advantages of Multi-threading:

- Enhanced performance by decreased development time.
- Simplified and streamlined program coding
- Improved 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 science.

Multi-threading does not only provide you with benefits, it has its disadvantages too. Let us go through some common disadvantages:

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