|  |  |
| --- | --- |
| **SESSION** | **AUGUST2023** |
| **PROGRAM** | **BACHELOR of COMPUTER APPLICATION (BCA)** |
| **SEMESTER** | **IV** |
| **course CODE & NAME** | **DCA2202 – PROGRAMNING IN JAVA** |

**Assignment Set- 1**

**1. a. Explain any five features of Java.**

**Ans 1a.**

**Five features of Java**

**1. Simple**

Java is easy for the professional programmer to learn and use effectively. For an experienced C++ programmer, learning Java will require very little effort because Java inherits the C/C++ syntax and many of the object- oriented features of C++. Expressiveness is more in Java. Most of the complex or confusing features in C++ are removed in Java like pointers etc. or implemented in a cleaner, more approachable manner.

**2. Secure**

Java forces you to handle unexpected errors. This ensures that Java programs are robust

Its Half solved only

Buy Complete assignment from us

**Price – 190/ assignment**

**MUJ Manipal University Complete SolvedAssignments session JULY 2023**

buy cheap assignment help online from us easily

we are here to help you with the best and cheap help

**Contact No – 8791514139 (WhatsApp)**

**OR**

**Mail us-** **bestassignment247@gmail.com**

**Our website -** [**www.assignmentsupport.in**](http://www.assignmentsupport.in)

**b. What is Byte code? Explain in brief.**

**Ans 1b.**

Bytecode is a low-level representation of a computer program that is used as an intermediate step between source code and machine code in various programming languages and execution environments. It plays a crucial role in the execution of programs in languages like Java, C#, and Python, among others. Here's a brief explanation of bytecode:

* **Intermediate Representation:** Bytecode serves as an intermediate representation of a program. When you write code in a high-level programming language like Java, the

**2. How do you implement inheritance in Java?**

**Ans 2.**

The ***extends*** keyword is used to derive a class from a superclass, or in other words, extend the functionality of a superclass.

**Syntax**

Public class <subclass\_name> extends <superclass\_name>

**Example**

Public class Confirmed extends Ticket

**3. What are the different methods under Data Input Stream and Data Output Stream?**

**Ans 3.**

DataInputStream and DataOutputStream are Java input/output classes used for reading and writing binary data in a platform-independent manner. These classes are part of the java.io package, and they wrap around other input and output stream sources, allowing primitive data types like int, float, short, and so on to be read from or written to the stream in a standardized format.

**DataInputStream:**

The DataInputStream allows an application to read primitive Java data types from an underlying input stream in a machine-independent way. It's usually used in combination with

**Assignment Set – 2**

**1. a. How to use assertion in Java?**

**Ans 1a.**

In Java, assertions are used as a debugging aid that provides a mechanism for testing assumptions during development. The assert keyword allows you to set up situations where, if a given condition turns out to be false, an error is thrown.

**b. What is Autoboxing and Unboxing in Java?**

**Ans 1b.**

Autoboxing and Unboxing are two of Java's mechanisms to bridge the gap between primitive data types (like **int**, **char**, **double**) and their corresponding wrapper classes (like **Integer**, **Character**, **Double**).

**2. Draw and explain the JDBC Application Architecture?**

**Ans 2.**

JDBC, or Java Database Connectivity, is a Java API that allows Java applications to interact with databases. It provides a standard interface for connecting to relational databases. Here, I will describe the JDBC Application Architecture and its key components.

**JDBC Application Architecture:**

1. **Application Layer:** This is the layer where your Java application resides. The application uses JDBC APIs to communicate with databases. Java applications utilize SQL statements embedded in Java code to perform CRUD operations on the database.

**3. What are the important steps required to create a JavaFX FXML Application?**

**Ans 3.**

JavaFX, enhanced with the FXML layout description language, offers a robust solution for building user interfaces in Java applications. Utilizing FXML for JavaFX applications involves several key steps. Here’s a rundown of those steps to create a JavaFX FXML application:

1. **Setting up the Environment:**

JavaFX SDK: Ensure you have the JavaFX SDK installed and added to your IDE. Some IDEs might require additional configuration or plugins.