**Session 2025-2026**

|  |  |
| --- | --- |
| **Vision:**  *To be a well-known centre for pursuing computer education through innovative pedagogy, value-based education and industry collaboration* | **Mission:** *To establish learning ambience for ushering in computer engineering professionals in core and multidisciplinary area by developing Problem-solving skills through emerging technologies****.*** |

**Program Educational Objectives of the program (PEO):** (broad statements that describe the professional and career accomplishments)

|  |  |  |  |
| --- | --- | --- | --- |
| PEO1 | **Preparation** | **P: Preparation** | **Pep-CL abbreviation**  **pronounce as Pep-si-lL easy to recall** |
| PEO2 | **Core Competence** | **E: Environment (Learning Environment)** |
| PEO3 | **Breadth** | **P: Professionalism** |
| PEO4 | **Professionalism** | **C: Core Competence** |
| PEO5 | **Learning Environment** | **L: Breadth (Learning in diverse areas)** |

**Program Outcomes (PO):** (statements that describe what a student should be able to do and know by the end of a program)

**Keywords of POs:**

Engineering knowledge, Problem analysis, Design/development of solutions, Conduct Investigations of Complex Problems, Engineering Tool Usage, The Engineer and The World, Ethics, Individual and Collaborative Team work, Communication, Project Management and Finance, Life-Long Learning

**PSO Keywords:** Cutting edge technologies, Research

“I am an engineer, and I know how to apply engineering knowledge to investigate, analyse and design solutions to complex problems using tools for entire world following all ethics in a collaborative way with proper management skills throughout my life.” *to contribute to the development of cutting-edge technologies and Research*.

**Integrity:** I will adhere to the Laboratory Code of Conduct and ethics in its entirety.

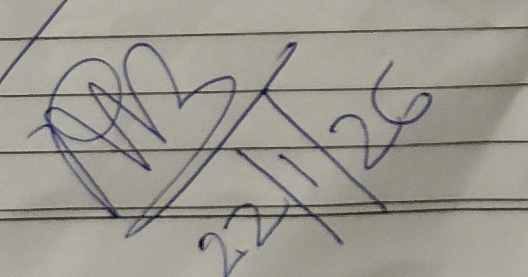
**Name and Signature of Student and Date**

(Signature and Date in Handwritten)

Kartik Gajanan Patil

****

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Session** | **2025-26 (EVEN)** | | **Course Name** | **JAVA FSD Lab** | |
| **Semester** | **4TH** | | **Course Code** | **23ADS1407** | |
| **Roll No** | **144** | | **Name of Student** | **KARTIK GAJANAN PATIL** | |
|  |  | |  |  |  |
| Practical Number | | **01** | | | |
| Course Outcome | | Install and configure the Java Development Kit (JDK) and set up the Java programming environment on a computer system.  **->**Understand the structure of a Java program and execute basic Java applications using command line and IDEs.  **->**Identify and use different data types available in Java for storing and manipulating data.  **->** Write simple Java programs using primitive and non-primitive data types.  **->**Demonstrate correct usage of variables, literals, and type casting in Java programs.  **->** Analyze program output based on the selection of appropriate data types. | | | |
| Aim | | Installation of JAVA (JDK) and introduction to Java Programming ( data types available in java) | | | |
| Problem Definition | | **Write a study assignment on Java Programming.** | | | |
| Theory  (100 words) | | **Java** is a widely used, high-level programming language that is designed to be platform-independent, secure, and object-oriented. To start developing Java applications, you must install the **Java Development Kit (JDK)**. The JDK includes tools like the Java compiler (javac), Java Runtime Environment (JRE), and other utilities needed for writing, compiling, and running Java programs. Java source code is written in .java files, which are then compiled into bytecode (.class files). This bytecode is executed by the **Java Virtual Machine (JVM)**, which allows the same Java program to run on different platforms without modification.  To install the JDK, first download it from the official Oracle website or any trusted OpenJDK provider. Run the installer and follow the setup instructions. After installation, you must configure the system’s **environment variables** by adding the JDK bin directory to the PATH. This allows you to run Java commands from any location in the command prompt or terminal. To verify a successful installation, open the command prompt and type java -version. If installed correctly, the system will display the installed Java version. Once the JDK is installed, Java programs can be developed using simple text editors or advanced IDEs such as Eclipse, IntelliJ IDEA, or NetBeans.  Java programming is based on the concept of objects and classes, making it an **object-oriented language**. It supports features such as inheritance, encapsulation, polymorphism, and abstraction. Java is also known for its security features, robustness, and ability to support multithreading, which allows multiple tasks to run simultaneously. Because Java programs run on the JVM, they are highly portable and can run on various operating systems without requiring changes to the code.  A key part of Java programming is understanding **data types**, which define the kind of data a variable can store. Java has two main categories of data types: primitive and non-primitive. **Primitive data types** are built-in types that store simple values. They include byte, short, int, and long for integer numbers; float and double for decimal numbers; char for single characters; and boolean for true or false values. Each primitive type has a fixed size and range. For example, int is commonly used for whole numbers, while double is used for decimal values with higher precision.  **Non-primitive data types**, also called reference types, store references to objects rather than actual values. Examples include String, arrays, classes, and interfaces. These types are created by programmers and can hold multiple values or complex data. Unlike primitive types, non-primitive data types do not have a fixed size and provide various methods for performing operations. Understanding both primitive and non-primitive data types is essential for writing efficient and error-free Java programs.  If you want, I can also provide **installation steps with screenshots** or **sample Java programs** for practice. | | | |
| Procedure and Execution  (100 Words) | | Algorithm:  1 **Start**  2 Open **IntelliJ IDEA**.  3 Create a **New Java Project** and select the installed **JDK**.  4 Create a **Java class** in the src folder.  5 Write the Java program using **data types**.  6 **Check and remove errors** (fix syntax errors shown in red).  7 Compile and **run the program**.  8 View the output in the **console window**.  9 **Stop**  Step 1: Open IntelliJ IDEA  Start IntelliJ IDEA  Click New Project    STEP 2: CREATE JAVA PROJECT  SELECT JAVA  CHOOSE THE INSTALLED **JDK**  CLICK NEXT  **CLICK FINISH**  STEP 3: CREATE JAVA CLASS  OPEN SRC FOLDER  RIGHT-CLICK->JAVA->NEWCLASS  TYPE CLASS NAME  PRESS ENTER    **STEP 4: WRITE JAVA CODE**    **STEP 5: TO DETECT ERROR**    **STEP 6 : RUN THE PROGRAM**  CLICK ON THE GREEN RUN BUTTON  OR  RIGHT CLICK IN EDITOR AND THEN ON RUN MAIN    **STEP 7 : VIEW OUTPUT**    **STEP 8: VIEWING OUTPUT IN CONSOLE WINDOW**    **STEP 9: STOP** | | | |
| Code: class Main {  public static void main(String[] args) {  int age = 20;  long population = 7800000000L;  float percentage = 85.5f;  double salary = 55000.75;  char grade = 'A';  boolean isJavaFun = true;  String course = "Java Programming";  System.out.println("Integer value (int): " + age);  System.out.println("Integer value (long): " + population);  System.out.println("Floating value (float): " + percentage);  System.out.println("Floating value (double): " + salary);  System.out.println("Character value: " + grade);  System.out.priSystem.out.println("String value: " + course); }} | | | |
| Output: | | | |
| Output Analysis | | The output demonstrates the use of different Java data types. The int data type stores a small whole number (20), while long stores a large integer value (7800000000). The float data type represents a decimal value (85.5) with single precision, and double represents a decimal value (55000.75) with higher precision. The char data type stores a single character (‘A’). The boolean data type displays a logical value (true). The String data type stores and displays text, shown as “Java Programming”. | | | |
| Link of student Github profile where lab assignment has been uploaded | | [**https://github.com/Kartikpatil1905**](https://github.com/Kartikpatil1905)  **https://github.com/Kartikpatil1905/JAVA-prac1-** | | | |
| Conclusion | | The given output confirms that Java supports multiple data types to store and display different kinds of data efficiently. Each data type is used according to the nature and size of the value, such as integers, large numbers, decimal values, characters, logical values, and strings. This demonstrates how Java ensures proper memory usage and accurate data handling. Understanding data types helps programmers choose the correct type for variables, avoid errors, and write efficient and reliable Java programs. | | | |
| Plag Report (Similarity index < 12%) | |  | | | |
| Date | | **22\1\2026** | | | |

****