Preeti. S. Mittal

Java Basics & OOPs Assignment Answers

## Q1. What is Java? Explain its features.

Java is a high-level, object-oriented programming language developed by Sun Microsystems. It is platform-independent, secure, robust, and simple to use.  
Features:  
- Platform Independent (Write Once, Run Anywhere)  
- Object-Oriented  
- Secure  
- Robust  
- Multithreaded  
- High Performance (using JIT compiler)  
- Distributed

## Q2. Explain the Java program execution process.

1. Write source code (.java file)  
2. Compile using javac to get bytecode (.class file)  
3. Bytecode executed by JVM (Java Virtual Machine)  
4. JVM interprets and runs the program on the specific platform

## Q3. Write a simple Java program to display 'Hello World'.

## 

## Q4. What are data types in Java? List and explain them.

Data types specify the type of data a variable can hold.  
- Primitive Data Types: byte, short, int, long, float, double, char, boolean  
- Non-Primitive Data Types: String, Arrays, Classes, Interfaces

## Q5. What is the difference between JDK, JRE, and JVM?

- JVM: Executes bytecode  
- JRE: Provides libraries and JVM to run Java programs  
- JDK: Includes JRE + development tools (compiler, debugger)

## Q6. What are variables in Java? Explain with examples.

Variables store data values.  
Example:  
int age = 25;  
String name = "John";  
Types:  
- Local Variables  
- Instance Variables  
- Static Variables

## Q7. What are the different types of operators in Java?

- Arithmetic Operators (+, -, \*, /, %)  
- Relational Operators (==, !=, >, <, >=, <=)  
- Logical Operators (&&, ||, !)  
- Assignment Operators (=, +=, -=, etc.)  
- Unary Operators (+, -, ++, --)  
- Bitwise Operators (&, |, ^, ~)

## Q8. Explain control statements in Java (if, if-else, switch).

- if: Executes code block if condition is true  
- if-else: Chooses between two blocks based on condition  
- switch: Selects one of many code blocks to execute based on expression value

## Q9. Write a Java program to find whether a number is even or odd.

## 

## Q10. What is the difference between while and do-while loop?

- while: Condition checked before execution, may not run if false initially  
- do-while: Executes at least once, condition checked after execution

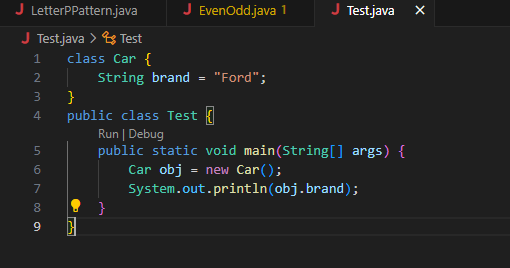
## Q11. What are the main principles of OOPs in Java? Explain each.

- Encapsulation: Wrapping data and methods  
- Inheritance: Acquiring properties from parent class  
- Polymorphism: Same operation in different forms  
- Abstraction: Hiding implementation details

## Q12. What is a class and an object in Java? Give examples.

- Class: Blueprint of objects  
- Object: Instance of a class

**Example:**

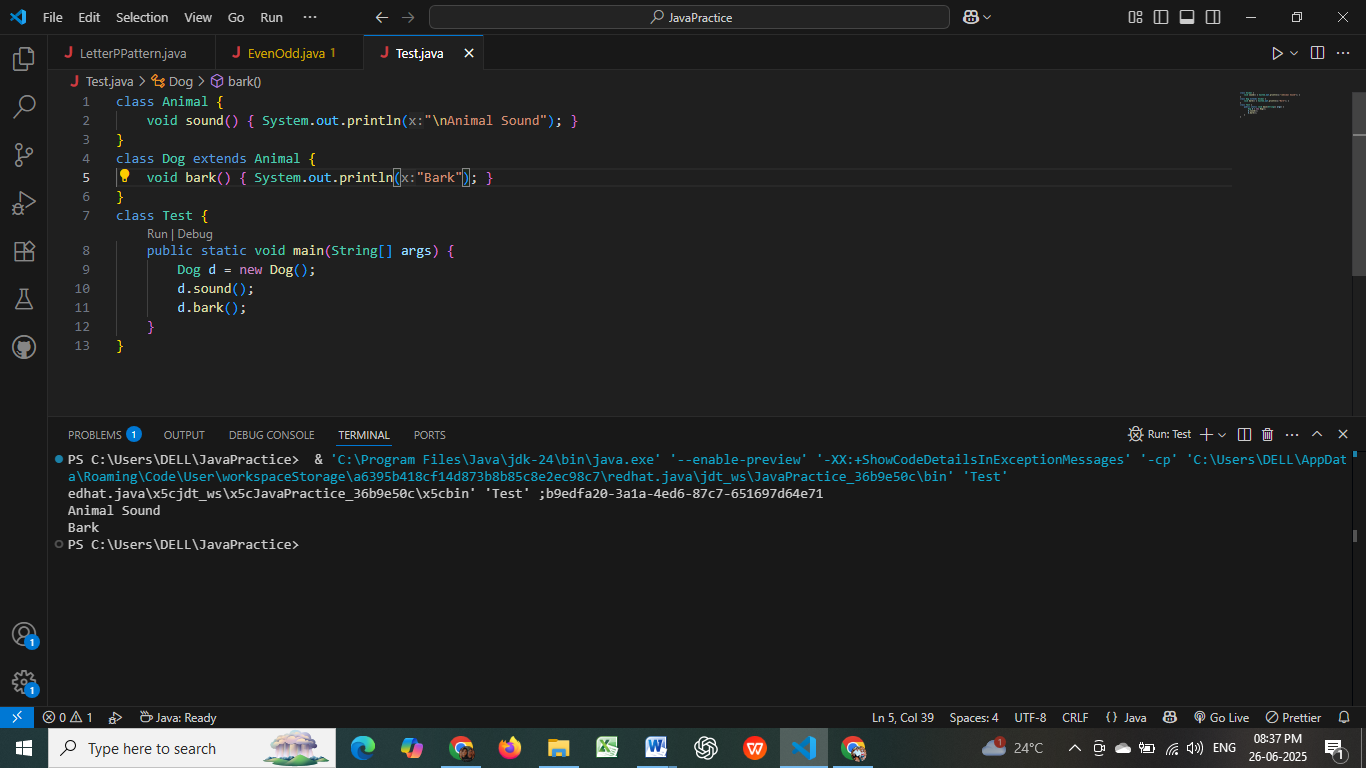


Q13. Write a program using class and object to calculate area of a rectangle.

## 

## Q14. Explain inheritance with real-life example and Java code.

Inheritance allows one class to acquire properties of another.  
Example:



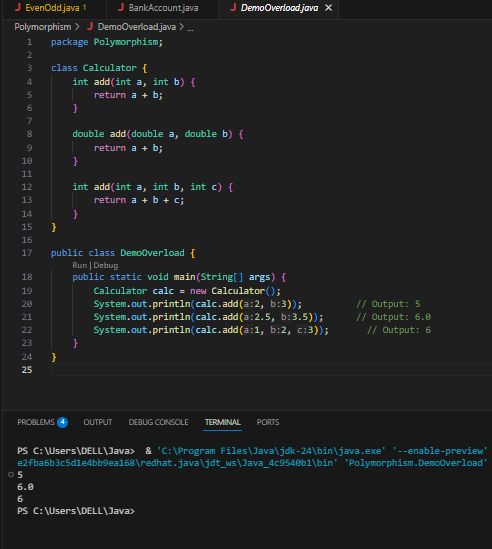
## Q15. What is polymorphism? Explain with compile-time and runtime examples.

Polymorphism means many forms.  
- Compile-time (Method Overloading)  
- Runtime (Method Overriding)  
Example:  
class Demo {  
 void show(int a) { System.out.println(a); }  
 void show(String s) { System.out.println(s); }  
}  
Method Overriding:  
class Parent {  
 void display() { System.out.println("Parent"); }  
}  
class Child extends Parent {  
 void display() { System.out.println("Child"); }  
}

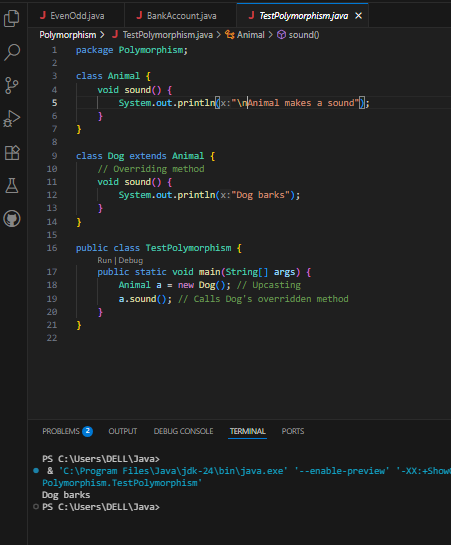
## Q16. What is method overloading and method overriding? Show with examples.

- Overloading: Same method name, different parameters  
- Overriding: Subclass provides specific implementation of parent method  
**Example:**

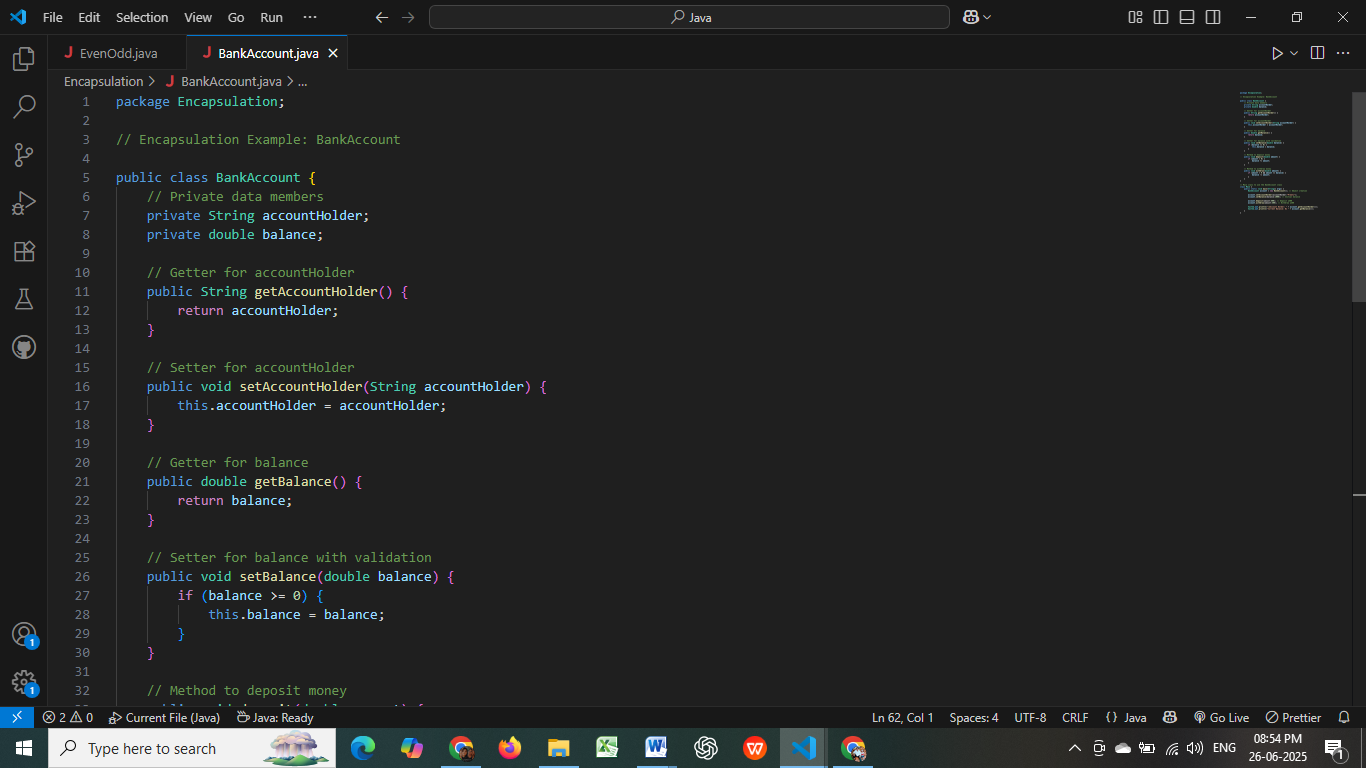
**Overloading:**

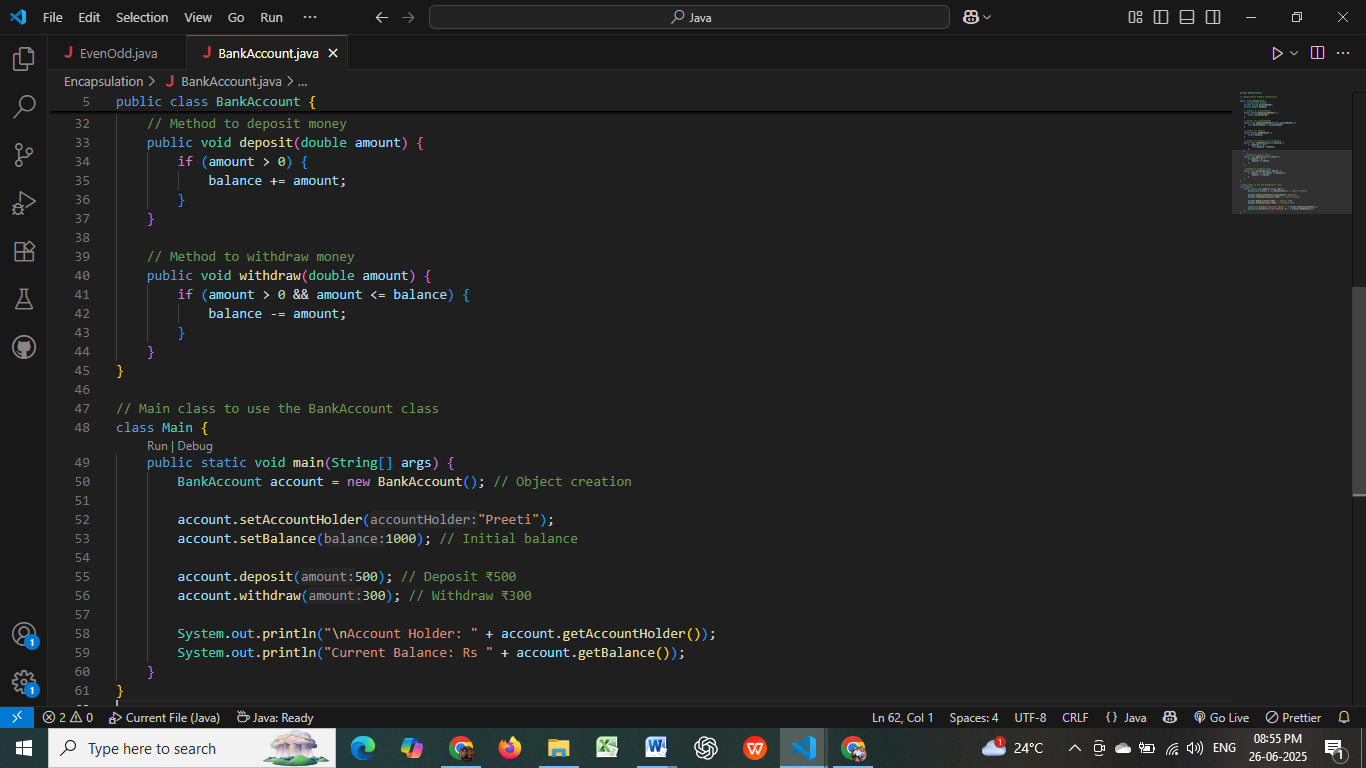


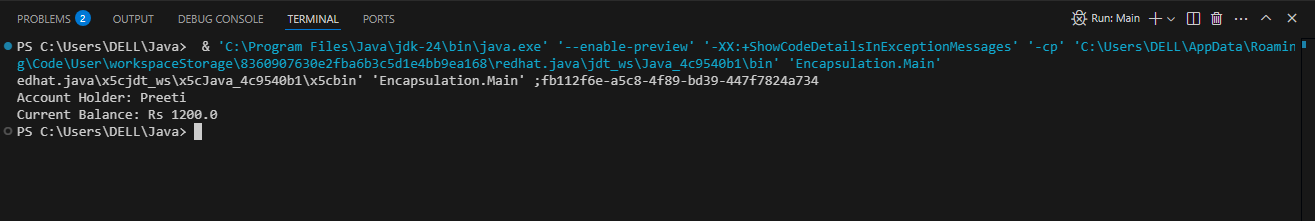
**Overriding:**



## Q17. What is encapsulation? Write a program demonstrating encapsulation.

Encapsulation binds data and code.  
**Example:**  






## Q18. What is abstraction in Java? How is it achieved?

Abstraction hides implementation details, achieved by:  
- Abstract classes (using abstract keyword)  
- Interfaces

## Q19. Explain the difference between abstract class and interface.

- Abstract class: Can have method implementations, constructors, variables  
- Interface: Only abstract methods (Java 8+ allows default/static methods)  
- A class can implement multiple interfaces but inherit only one class

## Q20. Create a Java program to demonstrate the use of interface.

