**Section 1: Java Data Types**

**1.** What are the different primitive data types available in Java?  
**Ans:** byte, short, int, long, float, double, char, boolean.

**2.** Explain the difference between primitive and non-primitive data types in Java.  
**Ans:** Primitive data types store simple values directly in memory and are predefined (e.g., int, double). Non-primitive data types store references to objects, can be user-defined (e.g., String, arrays, classes), and have more functionalities.

**3.** Write a Java program that demonstrates the use of all primitive data types.  
**Ans:** A program would declare variables of each type (byte, short, int, long, float, double, char, boolean) and print their values.

**4.** What is type casting? Provide an example of implicit and explicit casting in Java.  
**Ans:** Type casting is converting one data type into another.

* **Implicit casting (widening):** Automatically converts smaller type to larger type.
* **Explicit casting (narrowing):** Manually converts larger type to smaller type using a cast operator.

**5.** What is the default value of each primitive data type in Java?  
**Ans:**

* byte: 0
* short: 0
* int: 0
* long: 0L
* float: 0.0f
* double: 0.0d
* char: '\u0000'
* boolean: false

**Section 2: Java Control Statements**

**1.** What are control statements in Java? List the types with examples.  
**Ans:** Statements that control the flow of execution. Types:

* Selection: if, if-else, switch
* Looping: for, while, do-while
* Jump: break, continue, return

**2.** Write a Java program to demonstrate the use of if-else and switch-case statements.  
**Ans:** The program would use if-else for conditional checks and switch-case for multiple choice execution.

**3.** What is the difference between break and continue statements?  
**Ans:**

* break: Exits the loop entirely.
* continue: Skips the current iteration and moves to the next iteration.

**4.** Write a Java program to print even numbers between 1 to 50 using a for loop.  
**Ans:** Loop from 1 to 50, check if a number is divisible by 2, and print it.

**V.** Explain the differences between while and do-while loops with examples.  
**Ans:**

* **while:** Condition is checked before execution; may not execute at all.
* **do-while:** Executes at least once before checking condition.

**Section 3: Java Keywords and Operators**

**1.** What are keywords in Java? List 10 commonly used keywords.  
**Ans:** Reserved words with special meaning in Java. Examples: int, class, public, static, final, void, return, if, else, new.

**2.** Explain the purpose of the following keywords: static, final, this, super.  
**Ans:**

* static: Belongs to the class, shared among all objects.
* final: Makes variables constants, prevents method overriding, and inheritance.
* this: Refers to the current object.
* super: Refers to the parent class, used to access parent methods and constructors.

**3.** What are the types of operators in Java?  
**Ans:** Arithmetic, Relational, Logical, Assignment, Unary, Bitwise, Ternary, Instanceof.

**4.** Write a Java program demonstrating the use of arithmetic, relational, and logical operators.  
**Ans:** A program would perform addition, comparison, and logical operations using &&, ||, !.

**5.** What is operator precedence? How does it affect the outcome of expressions?  
**Ans:** It is the priority order in which operators are evaluated. Higher precedence operators are evaluated before lower precedence ones.

**Additional Questions — Java Data Types**

**6.** What is the size and range of each primitive data type in Java?  
**Ans:**

* byte: 1 byte (-128 to 127)
* short: 2 bytes (-32,768 to 32,767)
* int: 4 bytes (-2,147,483,648 to 2,147,483,647)
* long: 8 bytes (-9 quintillion to 9 quintillion approx.)
* float: 4 bytes (~±3.4×10^38)
* double: 8 bytes (~±1.8×10^308)
* char: 2 bytes (0 to 65,535)
* boolean: 1 bit (true/false)

**7.** How does Java handle overflow and underflow with numeric types?  
**Ans:** For integers, it wraps around silently (modular arithmetic). For floating-point, it results in Infinity, -Infinity, or NaN.

**8.** Write a program to convert a double value to an int without data loss.  
**Ans:** Not possible if there’s a fractional part; integer truncation occurs.

**9.** What is the difference between char and String in Java?  
**Ans:**

* char: Holds a single Unicode character.
* String: Holds a sequence of characters and is an object.

**10.** Explain wrapper classes and their use in Java.  
**Ans:** Wrapper classes (Integer, Double, etc.) convert primitives into objects, useful in collections, generics, and for utility methods.

**Additional Questions — Java Control Statements**

**1.** Write a Java program using nested if statements.  
**Ans:** The program would place one if statement inside another for multiple-level checks.

**2.** Write a Java program to display the multiplication table of a number using a loop.  
**Ans:** Loop from 1 to 10, multiply by the given number, and print results.

**3.** How do you exit from nested loops in Java?  
**Ans:** Use a labeled break statement to terminate all enclosing loops.

**4.** Compare and contrast for, while, and do-while loops.  
**Ans:**

* for: Best for known number of iterations.
* while: Best for condition-based looping, condition checked first.
* do-while: Executes at least once, condition checked after execution.

**5.** Write a program that uses a switch-case to simulate a basic calculator.  
**Ans:** The program would take two numbers and an operator, then perform the corresponding arithmetic operation using switch-case.

**Additional Questions — Java Keywords and Operators**

**1.** What is the use of the instanceof keyword in Java?  
**Ans:** Checks if an object is an instance of a given class or subclass.

**2.** Explain the difference between == and .equals() in Java.  
**Ans:**

* ==: Compares references for objects and actual values for primitives.
* .equals(): Compares contents of objects if overridden.

**3.** Write a program using the ternary operator.  
**Ans:** The program would use condition ? value1 : value2 to assign or print values based on a condition.

**4.** What is the use of this and super in method overriding?  
**Ans:**

* this: Refers to current class’s object, used to call current class members.
* super: Refers to parent class, used to call overridden methods or constructors.

**5.** Explain bitwise operators with examples.  
**Ans:**

* &: Bitwise AND
* |: Bitwise OR
* ^: Bitwise XOR
* ~: Bitwise complement
* <<: Left shift
* >>: Right shift