Java Programming Assignment -Answers

# Section 1: Java Data Types

## 1. What are the different primitive data types available in Java?

byte, short, int, long, float, double, char boolean.

## 2. Explain the difference between primitive and non-primitive data types in Java.

Primitive data types are predefined by Java.

Non-primitive data types are created by the programmer and can store multiple values.

## 3. Write a Java program that demonstrates the use of all primitive data types.

public class Main {

public static void main(String[] args) {

byte a = 10;

short b = 100;

int c = 1000;

long d = 10000;

float e = 10.5f;

double f = 99.99;

char g = 'A';

boolean h = true;

System.out.println(a + ", " + b + ", " + c + ", " + d);

System.out.println(e + ", " + f);

System.out.println(g + ", " + h);

}

}

Output:

A black screen with white numbers

AI-generated content may be incorrect.

## 4. What is type casting? Provide an example of implicit and explicit casting in Java.

Converting one data type to another data type.  
- Implicit: int a = 10; long b = a;  
- Explicit: double x = 9.78; int y = (int) x;

## 5. What is the default value of each primitive data type in Java?

byte: 0  
short: 0  
int: 0  
long: 0  
float: 0.0  
double: 0.0  
char: ' '  
boolean: false

# Section 2: Java Control Statements

## 1. What are control statements in Java? List the types with examples.

Control statements manage the flow of execution.

Types: if-else, switch, loops (for, while, do-while), break, continue.

## 2. Write a Java program to demonstrate the use of if-else and switch-case statements.

If-else:

public class Eampleg1 {

public static void main(String[] args) {

int a = 2;

if (a == 1) {

System.out.println("One");

} else {

System.out.println("Not One");

}

}

}

Switch:

public class Example2 {

public static void main(String[] args) {

int a = 2;

switch (a) {

case 1:

System.out.println("One");

break;

case 2:

System.out.println("Two");

break;

default:

System.out.println("Other number");

}

}

}

## 3. What is the difference between break and continue statements?

break statement exits the loop.

continue statement skips the current iteration and continues with the next.

## 4. Write a Java program to print even numbers between 1 to 50 using a for loop.

## public class Example3 {

## public static void main(String[] args) {

## for(int i=1; i<=50; i++) {

## if(i % 2 == 0) {

## System.out.print(i+",");

## }

## }

## }

## }

## 5. Explain the differences between while and do-while loops with examples.

while loop checks condition before execution.

Eg: public class Example {

public static void main(String[] args) {

int i = 1;

while (i <= 5) {

System.out.println("While Loop: " + i);

i++;

}

}

}

do-while loop checks condition after execution.  
 Eg: public class Example2 {

public static void main(String[] args) {

int i = 1;

do {

System.out.println("Do-While Loop: " + i);

i++;

} while (i <= 5); // condition checked after execution

}

}

# Section 3: Java Keywords and Operators

## 1. What are keywords in Java? List 10 commonly used keywords.

Keywords are reserved words and also that are pre-defined in Java.

Examples: class, public, static, void, int, new, return, if, else, switch

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

static: shared among all instances  
final: constant  
this: refers to current object  
super: refers to superclass

## 3. What are the types of operators in Java?

Arithmetic, Relational, Logical, Assignment, Bitwise, Unary, Ternary.

## 4. Write a Java program demonstrating the use of arithmetic, relational, and logical operators.

public class Example {

public static void main(String[] args) {

int a = 10, b = 20;

System.out.println("Arithmetic: " + (a + b));

System.out.println("Relational: " + (a < b));

System.out.println("Logical: " + (a < b && b > 15));

}

}

## 5. What is operator precedence? How does it affect the outcome of expressions?

It determines the order of evaluation in expressions.

\*, / have higher precedence than +, -

# Additional Questions - Java Data Types

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

byte: 1 byte (-128 to 127)  
short: 2 bytes  
int: 4 bytes  
long: 8 bytes  
float: 4 bytes  
double: 8 bytes  
char: 2 bytes  
boolean: 1 bit (true/false)

## 7. How does Java handle overflow and underflow with numeric types?

Java silently wraps around using modulo arithmetic.

Example: byte b = 127 + 1 results in -128.

## 8. Write a program to convert a double value to an int without data loss.

public class Example1 {

public static void main(String[] args) {

double d = 9.99;

int i = (int) d;

System.out.println(i);

}

}

## 9. What is the difference between char and String in Java?

char means a single alphabet.

String means combination of characters.

## 10. Explain wrapper classes and their use in Java.

Wrapper classes provide object representation of primitives.

eg: Integer for int. used in collections

# Additional Questions - Java Control Statements

## 6. Write a Java program using nested if statements.

## public class Main1 {

## public static void main(String[] args) {

## int num = 10;

## if (num > 0) {

## if (num % 2 == 0) {

## System.out.println("Even");

## }

## }

## }

## }

## 7. Write a Java program to display the multiplication table of a number using a loop.

## public class Main {

## public static void main(String[] args) {

## int n = 5;

## for(int i = 1; i <= 10; i++) {

## System.out.println(n + " x " + i + " = " + (n\*i));

## }

## }

## }

## 8. How do you exit from nested loops in Java?

By using labeled break:  
public class Main {

public static void main(String[] args) {

outer: // label for outer loop

for (int i = 1; i <= 3; i++) {

for (int j = 1; j <= 3; j++) {

System.out.println("i = " + i + ", j = " + j);

if (i == 2 && j == 2) {

break outer;

}

}

}

System.out.println("Exited from nested loops");

}

}

## 9. Compare and contrast for, while, and do-while loops.

Similarities: All are loops and execute block of code

Differences:

for loop checks condition before each iteration.

while loop checks condition before each iteration.

do-while loop checks condition after each iteration.

## 10. Write a program that uses a switch-case to simulate a basic calculator.

# public class Main {

# public static void main(String[] args) {

# int a = 9, b = 2;

# char op = '+';

# switch (op) {

# case '+':

# System.out.println(a + b);

# break;

# case '-':

# System.out.println(a - b);

# break;

# case '\*':

# System.out.println(a \* b);

# break;

# case '/':

# if (b != 0) {

# System.out.println(a / b);

# } else {

# System.out.println("Error: Division by zero");

# }

# break;

# default:

# System.out.println("Invalid operator");

# }

# }

# }

# Additional Questions - Java Keywords and Operators

## 6. What is the use of the `instanceof` keyword in Java?

It is used to check if an object is an instance of a specific class or subclass.

## 7. Explain the difference between `==` and `.equals()` in Java.

== compares reference (memory address),

.equals() compares object content.

## 8. Write a program using the ternary operator.

## public class Main {

## public static void main(String[] args) {

## int a = 10, b = 20;

## int max = (a > b) ? a : b;

## System.out.println("Max: " + max);

## }

## }

## 9. What is the use of `this` and `super` in method overriding?

`this` refers to current class object and

`super` refers to parent class method/constructor.

## 10. Explain bitwise operators with examples.

public class Example2 {

public static void main(String[] args) {

int a = 5; // 0101

int b = 3; // 0011

System.out.println("a & b = " + (a & b)); // 1

System.out.println("a | b = " + (a | b)); // 7

System.out.println("a ^ b = " + (a ^ b)); // 6

System.out.println("~a = " + (~a)); // -6

System.out.println("a << 1 = " + (a << 1));// 10

System.out.println("a >> 1 = " + (a >> 1));// 2

}

}