

## Java Basics Coding Challenges

### 1. Primitive Data Types

#### 1. Challenge: Declare and initialize variables of all primitive types and print their default values.

```
public class PrimitiveDefaults {
    public static void main(String[] args) {
        byte b = 10;
        short s = 100;
        int i = 1000;
        long l = 10000L;
        float f = 10.5f;
        double d = 20.123;
        char c = 'A';
        boolean bool = true;

        System.out.println("byte: " + b);
        System.out.println("short: " + s);
        System.out.println("int: " + i);
        System.out.println("long: " + l);
        System.out.println("float: " + f);
        System.out.println("double: " + d);
        System.out.println("char: " + c);
        System.out.println("boolean: " + bool);
    }
}
```

#### 2. Challenge: Write a program to detect overflow when adding two byte variables.

```
public class ByteOverflow {
    public static void main(String[] args) {
        byte a = 120;
        byte b = 10;
        int sum = a + b;

        byte result = (byte) sum;

        System.out.println("Actual sum (int): " + sum);
        System.out.println("Overflowed result (byte): " + result);
    }
}
```

**3. Challenge: Use type casting to convert double to int and float to byte.**

```
public class TypeCastingExample {
    public static void main(String[] args) {
        double d = 9.99;
        float f = 128.75f;
        int i = (int) d;
        byte b = (byte) f;
        System.out.println("Original double: " + d + ", After casting to int: " + i);
        System.out.println("Original float: " + f + ", After casting to byte: " + b);
    }
}
```

**4. Challenge: Perform bitwise operations between int and byte.**

```
public class BitwiseOperations {
    public static void main(String[] args) {
        int a = 5;
        byte b = 3;

        System.out.println("a & b = " + (a & b));
        System.out.println("a | b = " + (a | b));
        System.out.println("a ^ b = " + (a ^ b));
        System.out.println("~a = " + (~a));
    }
}
```

**5. Challenge: Accept input for all primitive types and display them formatted.**

```
import java.util.Scanner;

public class PrimitiveInput {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter byte: ");
        b = sc.nextByte();
        System.out.print("Enter short: ");
        short s = sc.nextShort();
        System.out.print("Enter int: ");
        int i = sc.nextInt();
        System.out.print("Enter long: ");
        long l = sc.nextLong();
        System.out.print("Enter float: ");
        float f = sc.nextFloat();
        System.out.print("Enter double: ");
        double d = sc.nextDouble();
    }
}
```

```

System.out.print("Enter char: ");
char c = sc.next().charAt(0);

System.out.print("Enter boolean: ");
boolean bool = sc.nextBoolean();

System.out.printf("\n%-10s : %d\n", "byte", b);
System.out.printf("%-10s : %d\n", "short", s);
System.out.printf("%-10s : %d\n", "int", i);
System.out.printf("%-10s : %d\n", "long", l);
System.out.printf("%-10s : %.2f\n", "float", f);
System.out.printf("%-10s : %.2f\n", "double", d);
System.out.printf("%-10s : %c\n", "char", c);
System.out.printf("%-10s : %b\n", "boolean", bool);
    }
}

```

### 1. Challenge: Swap two numbers using a temporary variable.

```

import java.util.Scanner;

public class SwapNumbers {

    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);
        System.out.println("Enter a:");
        int a = s.nextInt();
        System.out.println("Enter b:");
        int b = s.nextInt();

        System.out.println("Before swap: a = " + a + ", b = " + b)
    }
}

```

### 2. Challenge: Swap two numbers without a temporary variable.

```

a = a + b;
b = a - b;
a = a - b;
System.out.println("After swap: a = "+a + " ,b= "+b);

```

### 3. Challenge: Demonstrate variable shadowing within a class and method.

```

public class ShadowExample {
    int number = 10;
}

```

```

void display(int number) {
    System.out.println("Local variable (parameter): " + number);
    System.out.println("Instance variable using 'this': " + this.number);
}

public static void main(String[] args) {
    ShadowExample obj = new ShadowExample();
    obj.display(20);
}
}

```

**4. Challenge: Declare a constant and use it in calculations.**

```

public class CircleArea {
    public static final double PI = 3.14159;
    public static void main(String[] args) {
        double radius = 5.0;
        double area = PI * radius * radius;
        System.out.println("Radius: " + radius);
        System.out.println("Area of Circle: " + area);
    }
}

```

**5. Challenge: Create a class with instance, static, and local variables and demonstrate scope.**

```

public class ScopeDemo {
    int instanceVar = 10;
    static int staticVar = 20;
    void display() {
        int localVar = 30;
        System.out.println("Instance Variable: " + instanceVar);
        System.out.println("Static Variable: " + staticVar);
        System.out.println("Local Variable: " + localVar);
    }

    public static void main(String[] args) {
        ScopeDemo obj1 = new ScopeDemo();
        ScopeDemo obj2 = new ScopeDemo();

        obj1.instanceVar = 100;
        ScopeDemo.staticVar = 200;

        obj1.display();
        obj2.display();
    }
}

```

**1. Challenge: Demonstrate all arithmetic operators using two integers.**

```
public class ArithmeticOperators {
    public static void main(String[] args) {
        int a = 10, b = 3;

        System.out.println("Addition: " + (a + b));
        System.out.println("Subtraction: " + (a - b));
        System.out.println("Multiplication: " + (a * b));
        System.out.println("Division: " + (a / b));
        System.out.println("Modulus: " + (a % b));
    }
}
```

**2. Challenge: Use relational operators to compare ages.**

```
public class RelationalOperators {
    public static void main(String[] args) {
        int age1 = 25;
        int age2 = 30;

        System.out.println("age1 > age2: " + (age1 > age2));
        System.out.println("age1 < age2: " + (age1 < age2));
        System.out.println("age1 == age2: " + (age1 == age2));
        System.out.println("age1 != age2: " + (age1 != age2));
    }
}
```

**3. Challenge: Implement a basic calculator using switch and operators.**

```
import java.util.Scanner;

public class BasicCalculator {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.print("Enter first number: ");
        double num1 = sc.nextDouble();
        System.out.print("Enter second number: ");
        double num2 = sc.nextDouble();
        System.out.print("Choose operation (+, -, *, /): ");
        char op = sc.next().charAt(0);

        switch (op) {
            case '+': System.out.println("Result: " + (num1 + num2));
            break;
            case '-': System.out.println("Result: " + (num1 - num2));
            break;
            case '*': System.out.println("Result: " + (num1 * num2));
            break;
            case '/': System.out.println("Result: " + (num1 / num2));
            break;
            default: System.out.println("Invalid operation");
        }
    }
}
```

```

case '/':
    if (num2 != 0)
        System.out.println("Result: " + (num1 / num2));
    else
        System.out.println("Cannot divide by zero!");
break;
default: System.out.println("Invalid operator.");
    }
}
}

```

**4. Challenge: Use bitwise AND, OR, XOR on two binary values.**

```

public class BitwiseOperators {
    public static void main(String[] args) {
        int a = 5;
        int b = 3;

        System.out.println("Bitwise AND (a & b): " + (a & b)); // 0101 & 0011 = 0001 => 1
        System.out.println("Bitwise OR (a | b): " + (a | b)); // 0101 | 0011 = 0111 => 7
        System.out.println("Bitwise XOR (a ^ b): " + (a ^ b)); // 0101 ^ 0011 = 0110 => 6
    }
}

```

**5. Challenge: Demonstrate logical operators with Boolean expressions.**

```

public class LogicalOperators {
    public static void main(String[] args) {
        boolean a = true;
        boolean b = false;

        System.out.println("a && b: " + (a && b));
        System.out.println("a || b: " + (a || b));
        System.out.println("!a: " + (!a));
    }
}

```

**1. Challenge: Reverse a string using `StringBuilder`.**

```

public class ReverseString {
    public static void main(String[] args) {
        String original = "Hello World";
        StringBuilder sb = new StringBuilder(original);

        sb.reverse();

        System.out.println("Original: " + original);
        System.out.println("Reversed: " + sb);
    }
}

```

**2. Challenge: Append multiple strings using `StringBuilder` and print.**

```
public class AppendStrings {    public
static void main(String[] args) {
    StringBuilder sb = new StringBuilder();

    sb.append("Java ");
    sb.append("is ");
    sb.append("powerful!");

    System.out.println("Final String: " + sb);
}
}
```

**3. Challenge: Replace characters in a string using `StringBuilder`.**

```
public class ReplaceCharacters {

public static void main(String[] args) {

    StringBuilder sb = new StringBuilder("Jxvx is fxn");

    sb.setCharAt(1, 'a');
    sb.setCharAt(5, 'a');

    sb.setCharAt(10, 'u');

    System.out.println("Modified String: " + sb);

}

}
```

**4. Challenge: Insert a word into a string at a specific position.**

```
public class InsertWord {
    public static void main(String[] args) {
        StringBuilder sb = new StringBuilder("I coding");

        sb.insert(2, "love ");

        System.out.println("Updated Sentence: " + sb);
    }
}
```

**5. Challenge: Delete part of a string using `StringBuilder`.**

```
public class DeletePart {
    public static void main(String[] args) {
        StringBuilder sb = new StringBuilder("I love Java programming");
```

```

        sb.delete(7, 12);

        System.out.println("After Deletion: " + sb);
    }
}

```

**1. Challenge: Use `charAt()`, `length()`, and `substring()` methods.**

```

public class StringMethodsDemo {
    public static void main(String[] args) {
        String str = "Programming";

        char firstChar = str.charAt(0);

        int len = str.length();

        String sub = str.substring(3, 8);
        System.out.println("Original String: " + str);
        System.out.println("First Character: " + firstChar);
        System.out.println("Length: " + len);
        System.out.println("Substring (3 to 7): " + sub);
    }
}

```

**2. Challenge: Count the number of vowels in a string.**

```

public class CountVowels {
    public static void main(String[] args) {
        String str = "Hello World";
        int count = 0;
        str = str.toLowerCase();
        for (int i = 0; i < str.length(); i++) {
            char c = str.charAt(i);
            if ("aeiou".indexOf(c) != -1) {
                count++;
            }
        }

        System.out.println("Number of vowels: " + count);
    }
}

```

**3. Challenge: Check if a string is a palindrome.**

```

public class PalindromeCheck {
    public static void main(String[] args) {
        String str = "madam";
        String reversed = "";

        for (int i = str.length() - 1; i >= 0; i--) {
            reversed += str.charAt(i);
        }
    }
}

```



```

    }

    if (str.equals(reversed)) {
        System.out.println(str + " is a palindrome.");
    } else {
        System.out.println(str + " is not a palindrome.");
    }
}
}

```

4. Challenge: Convert a string to upper case and lower case.

```

public class CaseConversion {
    public static void main(String[] args) {
        String str = "Java Programming";

        System.out.println("Original: " + str);
        System.out.println("Upper Case: " + str.toUpperCase());
        System.out.println("Lower Case: " + str.toLowerCase());
    }
}

```

5. Challenge: Remove spaces and special characters from a string.

```

public class CleanString {
    public static void main(String[] args) {
        String input = "He@!#l$o Wo!rld!";
        String cleaned = input.replaceAll("[^a-zA-Z0-9]", "");
        System.out.println("Original: " + input);
        System.out.println("Cleaned: " + cleaned);
    }
}

```

1. Challenge: Get current date and time using **LocalDateTime**.

```

import java.time.LocalDateTime;
public class CurrentDateTime {
    public static void main(String[] args) {
        LocalDateTime current = LocalDateTime.now();
        System.out.println("Current Date and Time: " + current);
    }
}

```

2. Challenge: Calculate age given a birth date.

```

import java.time.LocalDate;
import java.time.Period;

public class CalculateAge {

```

```

public static void main(String[] args) {
    LocalDate birthDate = LocalDate.of(2000, 7, 27);
    LocalDate today = LocalDate.now();
    Period age = Period.between(birthDate, today);
    System.out.println("Age: " + age.getYears() + " years");
}
}

```

**3. Challenge: Format date in dd-MM-yyyy format.**

```

import java.time.LocalDate; import
java.time.format.DateTimeFormatter;
public class FormatDate {
    public static void main(String[] args) {
        LocalDate today = LocalDate.now();
        DateTimeFormatter formatter = DateTimeFormatter.ofPattern("dd-MM-yyyy");
        String formatted = today.format(formatter);
        System.out.println("Formatted Date: " + formatted);
    }
}

```

**4. Challenge: Add 5 days to current date and print.**

```

import java.time.LocalDate;
public class AddDays {
    public static void main(String[] args) {
        LocalDate today = LocalDate.now();
        LocalDate futureDate = today.plusDays(5);
        System.out.println("Today: " + today);
        System.out.println("Date after 5 days: " + futureDate);
    }
}

```

**5. Challenge: Round a decimal to 2 places using BigDecimal.**

```

import java.math.BigDecimal;
import java.math.RoundingMode;

public class RoundDecimal {
    public static void main(String[] args) {
        BigDecimal number = new BigDecimal("123.45678");
        BigDecimal rounded = number.setScale(2, RoundingMode.HALF_UP);

        System.out.println("Original: " + number);
        System.out.println("Rounded: " + rounded);
    }
}

```

**1. Challenge: Use if-else to determine if a number is positive, negative, or zero.**

```
import java.util.Scanner;

public class NumberSignCheck {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a number: ");
        int num = sc.nextInt();

        if (num > 0) {
            System.out.println("The number is positive.");
        } else if (num < 0) {
            System.out.println("The number is negative.");
        } else {
            System.out.println("The number is zero.");
        }
    }
}
```

**2. Challenge: Implement nested if to find the largest among 3 numbers.**

```
import java.util.Scanner;

public class LargestOfThree {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter three numbers: ");
        int a = sc.nextInt(), b = sc.nextInt(), c = sc.nextInt();

        if (a >= b) {
            if (a >= c) {
                System.out.println("Largest is: " + a);
            } else {
                System.out.println("Largest is: " + c);
            }
        } else {
            if (b >= c) {
                System.out.println("Largest is: " + b);
            } else {
                System.out.println("Largest is: " + c);
            }
        }
    }
}
```

### 3. Challenge: Validate login with username and password.

```
import java.util.Scanner;

public class LoginValidation {
    public static void main(String[] args) {
        String correctUsername = "admin";
        String correctPassword = "1234";

        Scanner sc = new Scanner(System.in);
        System.out.print("Enter username: ");
        String username = sc.next();
        System.out.print("Enter password: ");
        String password = sc.next();

        if (username.equals(correctUsername) && password.equals(correctPassword)) {
            System.out.println("Login successful!");
        } else {
            System.out.println("Invalid username or password.");
        }
    }
}
```

### 4. Challenge: Categorize age groups using if-else ladder.

```
import java.util.Scanner;

public class AgeCategory {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter age: ");
        int age = sc.nextInt();

        if (age < 0) {
            System.out.println("Invalid age.");
        } else if (age < 13) {
            System.out.println("Child");
        } else if (age < 20) {
            System.out.println("Teenager");
        } else if (age < 60) {
            System.out.println("Adult");
        } else {
            System.out.println("Senior Citizen");
        }
    }
}
```

### 5. Challenge: Determine student grade using percentage.

```
import java.util.Scanner;
```

```

public class StudentGrade {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter percentage: ");
        double percentage = sc.nextDouble();

        if (percentage >= 90) {
            System.out.println("Grade: A+");
        } else if (percentage >= 80) {
            System.out.println("Grade: A");
        } else if (percentage >= 70) {
            System.out.println("Grade: B");
        } else if (percentage >= 60) {
            System.out.println("Grade: C");
        } else if (percentage >= 50) {
            System.out.println("Grade: D");
        } else {
            System.out.println("Grade: F (Fail)");
        }
    }
}

```

**1. Challenge: Check if a number is even and divisible by 5**

```

import java.util.Scanner;
public class EvenAndDivisibleBy5 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a number: ");
        int num = sc.nextInt();

        if (num % 2 == 0 && num % 5 == 0) {
            System.out.println(num + " is even and divisible by 5.");
        } else {
            System.out.println(num + " is not even and divisible by 5.");
        }
    }
}

```

**2. Challenge: Validate a triangle (sum of angles = 180).**

```

import java.util.Scanner;

public class TriangleValidation {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter three angles: ");
        int a1 = sc.nextInt(), a2 = sc.nextInt(), a3 = sc.nextInt();
    }
}

```

```

        if (a1 + a2 + a3 == 180 && a1 > 0 && a2 > 0 && a3 > 0) {
            System.out.println("Valid Triangle");
        } else {
            System.out.println("Invalid Triangle");
        }
    }
}

```

### 3. Challenge: Check if year is a leap year.

```

import java.util.Scanner;
public class LeapYearCheck {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a year: ");
        int year = sc.nextInt();

        if ((year % 4 == 0 && year % 100 != 0) || year % 400 == 0) {
            System.out.println(year + " is a leap year.");
        } else {
            System.out.println(year + " is not a leap year.");
        }
    }
}

```

### 4. Challenge: Check character type (vowel/consonant/digit/special).

```

import java.util.Scanner;

public class CharacterTypeCheck {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a character: ");
        char ch = sc.next().charAt(0);

        if (Character.isDigit(ch)) {
            System.out.println("It is a digit.");
        } else if (Character.isLetter(ch)) {
            ch = Character.toLowerCase(ch);
            if ("aeiou".indexOf(ch) != -1) {
                System.out.println("It is a vowel.");
            } else {
                System.out.println("It is a consonant.");
            }
        } else {
            System.out.println("It is a special character.");
        }
    }
}

```

## 5. Challenge: Check eligibility for vote, driving, and job using conditions.

```
import java.util.Scanner;

public class EligibilityCheck {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter your age: ");
        int age = sc.nextInt();

        if (age >= 18) {
            System.out.println("Eligible to vote.");
        } else {
            System.out.println("Not eligible to vote.");
        }

        if (age >= 16) {
            System.out.println("Eligible for driving learner's license.");
        } else {
            System.out.println("Not eligible for driving.");
        }

        if (age >= 21) {
            System.out.println("Eligible for most jobs.");
        } else {
            System.out.println("May not be eligible for all jobs.");
        }
    }
}
```

### 1. Challenge: Create a calculator using switch.

```
import java.util.Scanner;

public class SwitchCalculator {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter first number: ");
        double a = sc.nextDouble();
        System.out.print("Enter second number: ");
        double b = sc.nextDouble();
        System.out.print("Enter operation (+, -, *, /): ");
        char op = sc.next().charAt(0);

        switch (op) {
            case '+':
```

```

        System.out.println("Result: " + (a + b));
break;    case '-':
        System.out.println("Result: " + (a - b));
break;    case '*':
        System.out.println("Result: " + (a * b));
break;    case '/':
        if (b != 0)
            System.out.println("Result: " + (a / b));
        else
            System.out.println("Cannot divide by zero!");
break;    default:
        System.out.println("Invalid operator.");
    }
}
}

```

## 2. Challenge: Map number to month name using switch.

```

import java.util.Scanner;

public class MonthMapper {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter month number (1-12): ");
        int month = sc.nextInt();

        switch (month) {
            case 1: System.out.println("January"); break;
            case 2: System.out.println("February"); break;
            case 3: System.out.println("March"); break;
            case 4: System.out.println("April"); break;
            case 5: System.out.println("May"); break;
            case 6: System.out.println("June"); break;
            case 7: System.out.println("July"); break;
            case 8: System.out.println("August"); break;
            case 9: System.out.println("September"); break;
            case 10: System.out.println("October"); break;
            case 11: System.out.println("November"); break;
            case 12: System.out.println("December"); break;
            default: System.out.println("Invalid month number.");
        }
    }
}

```



### 3. Challenge: Implement a simple menu using switch.

```
import java.util.Scanner;

public class SimpleMenu {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.println("Menu:");
        System.out.println("1. Print Hello");
        System.out.println("2. Add two numbers");
        System.out.println("3. Exit");

        System.out.print("Enter your choice: ");
        int choice = sc.nextInt();

        switch (choice) {
        case 1:
            System.out.println("Hello!");
            break;
        case 2:
            System.out.print("Enter two numbers: ");
            int a = sc.nextInt();
            int b = sc.nextInt();
            System.out.println("Sum: " + (a + b));
            break;
        case 3:
            System.out.println("Exiting program...");
            break;
        default:
            System.out.println("Invalid choice.");
        }
    }
}
```

### 4. Challenge: Use enhanced switch (Java 14+) for better syntax. import java.util.Scanner;

```
public class EnhancedSwitch {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter day number (1-3): ");
        int day = sc.nextInt();

        String result = switch (day) {
        case 1 -> "One";
        case 2 -> "Two";
        case 3 -> "Three";
        default -> "Invalid day";
        };
    }
}
```

```

        System.out.println("Output: " + result);
    }
}

```

## 5. Challenge: Implement day of the week based on integer input.

```

import java.util.Scanner;

public class DayOfWeek {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter number (1-7): ");
        int day = sc.nextInt();

        switch (day) {
            case 1: System.out.println("Sunday"); break;
            case 2: System.out.println("Monday"); break;
            case 3: System.out.println("Tuesday"); break;
            case 4: System.out.println("Wednesday"); break;
            case 5: System.out.println("Thursday"); break;
            case 6: System.out.println("Friday"); break;
            case 7: System.out.println("Saturday"); break;
            default: System.out.println("Invalid input.");
        }
    }
}

```

### 1. Challenge: Print multiplication table for a number.

```

import java.util.Scanner;
public class MultiplicationTable {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a number: ");
        int num = sc.nextInt();
        System.out.println("Multiplication Table for " + num + ":");
        for (int i = 1; i <= 10; i++) {
            System.out.println(num + " x " + i + " = " + (num * i));
        }
    }
}

```

### 2. Challenge: Use break and continue in loops.

```

public class BreakContinueDemo {
    public static void main(String[] args) {
        System.out.println("Using continue (skip 5):");
    }
}

```

```

        for (int i = 1; i <= 10; i++) {
            if (i == 5)
                continue;
            System.out.print(i + " ");
        }

        System.out.println("\nUsing break (stop at 5):");
        for (int i = 1; i <= 10; i++) {
            if (i == 5)
                break;
            System.out.print(i + " ");
        }
    }
}

```

### 3. Challenge: Find factorial of a number.

```

import java.util.Scanner;

public class FactorialCalculator {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a number: ");
        int n = sc.nextInt();
        long fact = 1;

        for (int i = 1; i <= n; i++) {
            fact *= i;
        }

        System.out.println("Factorial of " + n + " is: " + fact);
    }
}

```

### 4. Challenge: Print Fibonacci series.

```

import java.util.Scanner;

public class FibonacciSeries {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter number of terms: ");
        int n = sc.nextInt();

        int a = 0, b = 1;
        System.out.print("Fibonacci Series: " + a + " " + b + " ");

        for (int i = 3; i <= n; i++) {
            int c = a + b;

```

```

        System.out.print(c + " ");
        a = b;
    b = c;
    }
}
}

```

**5. Challenge: Find sum of even numbers from 1 to 100.**

```

public class SumEvenNumbers {
    public static void main(String[] args) {
        int sum = 0;

        for (int i = 2; i <= 100; i += 2) {
            sum += i;
        }

        System.out.println("Sum of even numbers from 1 to 100: " + sum);
    }
}

```

**1. Challenge: Find the largest and smallest element in an array.**

```

public class MinMaxArray {
    public static void main(String[] args) {
        int[] numbers = {23, 56, 12, 89, 4, 67};
        int min = numbers[0];
        int max = numbers[0];

        for (int num : numbers) {
            if (num < min) min = num;
            if (num > max) max = num;
        }

        System.out.println("Smallest: " + min);
        System.out.println("Largest: " + max);
    }
}

```

**2. Challenge: Sort an array in ascending order.**

```

import java.util.Arrays;

public class SortArray {
    public static void main(String[] args) {
        int[] numbers = {23, 5, 12, 89, 4};
        Arrays.sort(numbers);

        System.out.println("Sorted array: " + Arrays.toString(numbers));
    }
}

```

```
}
```

**3. Challenge: Calculate average of numbers in an array.**

```
public class ArrayAverage {
    public static void main(String[] args) {
        int[] numbers = {10, 20, 30, 40, 50};
        int sum = 0;

        for (int num : numbers) {
            sum += num;
        }

        double average = (double) sum / numbers.length;
        System.out.println("Average: " + average);
    }
}
```

**4. Challenge: Count occurrence of an element.**

```
public class CountOccurrence {
    public static void main(String[] args) {
        int[] numbers = {1, 2, 3, 2, 4, 2, 5};
        int target = 2;
        int count = 0;

        for (int num : numbers) {
            if (num == target) count++;
        }

        System.out.println("Element " + target + " occurs " + count + " times.");
    }
}
```

**5. Challenge: Reverse elements of an array.**

```
import java.util.Arrays;

public class ReverseArray {
    public static void main(String[] args) {
        int[] numbers = {10, 20, 30, 40, 50};
        int start = 0, end = numbers.length - 1;

        while (start < end) {
            int temp = numbers[start];
            numbers[start] = numbers[end];
            numbers[end] = temp;
            start++;
            end--;
        }
    }
}
```

```

        System.out.println("Reversed array: " + Arrays.toString(numbers));
    }
}

```

**1. Challenge: Define an enum for days of the week.**

```

public class DaysEnumExample {
    enum Day {
        MONDAY, TUESDAY, WEDNESDAY, THURSDAY,
        FRIDAY, SATURDAY, SUNDAY
    }

    public static void main(String[] args) {
        Day today = Day.MONDAY;
        System.out.println("Today is: " + today);
    }
}

```

**2. Challenge: Use enum in switch case.**

```

public class EnumSwitchExample {
    enum Day {
        MONDAY, TUESDAY, WEDNESDAY, THURSDAY, FRIDAY,
        SATURDAY, SUNDAY
    }

    public static void main(String[] args) {
        Day today = Day.FRIDAY;

        switch (today) {
            case MONDAY -> System.out.println("Start of the
week");
            case FRIDAY -> System.out.println("Almost weekend");
            case SUNDAY -> System.out.println("Relax! It's
Sunday");
            default -> System.out.println("Midweek day");
        }
    }
}

```

### 3. Challenge: Iterate over enum values.

```
public class EnumIterationExample {
    enum Day {
        MONDAY, TUESDAY, WEDNESDAY, THURSDAY,
        FRIDAY, SATURDAY, SUNDAY
    }

    public static void main(String[] args) {
        System.out.println("Days of the week:");
        for (Day day : Day.values()) {
            System.out.println(day);
        }
    }
}
```

### 4. Challenge: Assign properties to enum constants.

```
public class EnumWithProperties {
    enum Planet {
        MERCURY(3.30), VENUS(4.87), EARTH(5.97), MARS(0.64);

        private final double mass;

        Planet(double mass) {
            this.mass = mass;
        }

        public double getMass() {
            return mass;
        }
    }

    public static void main(String[] args) {
        for (Planet planet : Planet.values()) {
            System.out.println(planet + " has mass " + planet.getMass() + "
x10^24 kg");
        }
    }
}
```

### 5. Challenge: Create an enum to represent traffic light states.

```
public class TrafficLightExample {
    enum TrafficLight {
        RED("Stop"),
        GREEN("Go"),
        YELLOW("Slow down");

        private final String action;
```

```
        TrafficLight(String action) {
this.action = action;
    }

    public String getAction() {
return action;
    }
}

    public static void main(String[] args) {
for (TrafficLight light : TrafficLight.values()) {
    System.out.println(light + " means: " + light.getAction());
}
}
}
```