

Ejercicio 1: Cálculo de sombreros necesarios

```
public class Main {  
    public static int calcularSombreros(double costoFijo, double costoVariable, double  
costoMaximo) {  
        return (int) Math.ceil(costoFijo / (costoMaximo - costoVariable));  
    }  
  
    public static void main(String[] args) {  
        System.out.println(calcularSombreros(800, 20, 25));  
    }  
}
```

Ejercicio 2: Determinar ficha ganadora

```
public class Main {  
    public static boolean esGanadora(int[] numeros) {  
        int puntos = 0;  
        for (int num : numeros) {  
            if (esPrimo(num)) puntos += 4;  
            if (num % 2 == 0) puntos += 1;  
            if (num % 7 == 0) puntos += 2;  
        }  
        return puntos > 70;  
    }  
  
    private static boolean esPrimo(int num) {  
        if (num < 2) return false;
```

```

        for (int i = 2; i <= Math.sqrt(num); i++) {
            if (num % i == 0) return false;
        }
        return true;
    }

    public static void main(String[] args) {
        int[] numeros = {2, 3, 7, 14, 29};
        System.out.println(esGanadora(numeros));
    }
}

```

Ejercicio 3: Número espejo

```

public class Main {
    public static boolean esNumeroEspejo(int numero) {
        String numStr = Integer.toString(numero);
        String reverseStr = new StringBuilder(numStr).reverse().toString();
        return numStr.equals(reverseStr);
    }

    public static void main(String[] args) {
        System.out.println(esNumeroEspejo(121));
    }
}

```

Ejercicio 4: Clasificación de estudiantes

```
import java.util.ArrayList;
import java.util.List;

public class Main {

    public static List<Integer>[] clasificarEdades(int[] edades) {

        List<Integer>[] grupos = new ArrayList[4];

        for (int i = 0; i < 4; i++) grupos[i] = new ArrayList<>();

        for (int edad : edades) {

            if (edad % 5 == 0 && edad % 2 != 0) grupos[0].add(edad);

            else if (edad % 4 == 0) grupos[1].add(edad);

            else if (edad % 3 == 0 && edad % 4 != 0 && edad % 5 != 0 && edad % 2 != 0)
grupos[2].add(edad);

            else grupos[3].add(edad);

        }

        return grupos;

    }

    public static void main(String[] args) {

        int[] edades = {10, 12, 15, 16, 17}; // Ejemplo

        List<Integer>[] grupos = clasificarEdades(edades);

        for (int i = 0; i < grupos.length; i++) {

            System.out.println("Grupo " + (i + 1) + ": " + grupos[i]);

        }

    }

}
```

Ejercicio 5: Operaciones con notas

```
import java.util.Arrays;

public class Main {

    public static double calcularPromedio(int[] notas) {
        return Arrays.stream(notas).average().orElse(0);
    }

    public static int notaMasAlta(int[] notas) {
        return Arrays.stream(notas).max().orElse(0);
    }

    public static int notaMasBaja(int[] notas) {
        return Arrays.stream(notas).min().orElse(0);
    }

    public static void main(String[] args) {
        int[] notas = {70, 85, 90, 65, 88, 72, 95, 60, 80, 75};
        System.out.println("Promedio: " + calcularPromedio(notas));
        System.out.println("Nota más alta: " + notaMasAlta(notas));
        System.out.println("Nota más baja: " + notaMasBaja(notas));
    }
}
```

Ejercicio 6: Admisión a un evento

```
public class Main {  
    public static String verificarAdmisión(String nombre, int edad, int espacioDisponible) {  
        if (edad < 15 || edad > 30) return "Lo siento, " + nombre + ", tu edad no está en el rango  
definido.";   
        if (espacioDisponible <= 0) return "Lo siento, " + nombre + ", el evento está lleno.";   
        return "¡Felicidades, " + nombre + ", eres parte del evento!";  
    }  
  
    public static void main(String[] args) {  
        System.out.println(verificarAdmisión("Juan", 25, 10));  
    }  
}
```

Ejercicio 7: Construcción de estantes

```
public class Main {  
    public static String calcularEstantes(int pedido, int[] materiales, int[] consumo) {  
        int[] sobrantes = new int[materiales.length];  
        for (int i = 0; i < materiales.length; i++) {  
            sobrantes[i] = materiales[i] - (consumo[i] * pedido);  
        }  
        if (Arrays.stream(sobrantes).allMatch(v -> v >= 0)) {  
            int adicionales = Arrays.stream(sobrantes).min().getAsInt() /  
Arrays.stream(consumo).max().getAsInt();  
            return "Es posible realizar el pedido. Con el material sobrante puedes construir " +  
adicionales + " estantes.";   
        } else {  

```

```

        return "No es posible cubrir el pedido.";
    }
}

public static void main(String[] args) {
    int pedido = 15;
    int[] materiales = {26, 33, 200, 20, 510};
    int[] consumo = {4, 6, 12, 2, 14};
    System.out.println(calcularEstantes(pedido, materiales, consumo));
}
}

```

Ejercicio 9: Envíos de paquetes

```

public class Main {
    public static String calcularEnvio(int[] pesos) {
        int totalPeso = Arrays.stream(pesos).sum();
        if (totalPeso > 3000) return "El envío es especial por exceso de peso.";
        return "Peso total: " + totalPeso + " gramos.";
    }

    public static void main(String[] args) {
        int[] pesos = {200, 300, 150, 800};
        System.out.println(calcularEnvio(pesos));
    }
}

```