

Soluciones Actividades

En este apartado, te compartimos algunas soluciones a los ejercicios que te hemos propuesto. Pero recuerda, no encontrarás un manual rígido ni una única manera de resolver los problemas. En su lugar, **te ofrecemos una perspectiva que abre puertas a diferentes maneras de enfrentar cada desafío.**

Cada actividad es una oportunidad para profundizar y comprender las posibles soluciones. Te animamos a ir más allá de buscar respuestas directas, y a utilizar tu curiosidad para explorar y personalizar los conocimientos adquiridos. Aquí, **el objetivo no es replicar respuestas, sino entender el proceso de pensamiento detrás de cada solución y cómo aplicarlo en distintas situaciones.**

Te alentamos a que, al utilizar estas soluciones, te tomes el tiempo necesario para comprender cada línea de código, para analizar cómo funciona y para adaptarlo a tus propias necesidades y proyectos.

La programación es un arte que requiere comprensión profunda y creatividad personal, y este espacio está diseñado para que desarrolles esas habilidades de manera óptima.



Actividad: Imprimiendo un mensaje por consola

1. Crea un programa que imprima "Bienvenido a tu curso de Java" en la consola.
2. Crea un programa que imprima "Mi nombre es [colocaAquiTuNombre]" en la consola.
3. Crea un programa que imprima "Mi edad es [colocaAquiTuEdad]" en la consola.

RESOLUCIÓN:

1.

```
public class App {  
    public static void main(String[] args) {  
        System.out.println("Bienvenido a tu curso de Java");  
    }  
}
```

2. .

```
public class App {
    public static void main(String[] args){
        String nombre = "TuNombre"; // Reemplaza "TuNombre" con tu nombre real
        System.out.println("Mi nombre es " + nombre);
    }
}
```

3.

```
public class App {
    public static void main(String[] args) {
        int edad = 25; // Reemplaza 25 con tu edad real
        System.out.println("Mi edad es " + edad);
    }
}
```



Actividad: Utilizando variables

1. Declara una variable por cada uno de los siguientes tipos de dato, asignando un valor (nombre y valor de tu preferencia):

- a. boolean
- b. char
- c. String
- d. Int
- e. long
- f. float

2. Imprime en consola el contenido de al menos tres de las variables que declaraste en el paso anterior.

3. Declara una nueva variable para almacenar tu edad y otra para almacenar tu nombre e imprimir en pantalla un texto que diga **“Mi nombre es [nombre], y tengo [edad] años”**.

RESOLUCIÓN:

```
public class App {
    public static void main(String[] args) {
        // Paso 1: Declarar variables y asignar valores
        boolean esEstudiante = true;
        char inicialApellido = 'G';
        String nombreCompleto = "Juan Pérez";
        int edad = 30;
        long numeroGrande = 1234567890L;
        float altura = 1.75f;

        // Paso 2: Imprimir contenido de las variables
    }
}
```

```

System.out.println("Valor de la variable boolean: " + esEstudiante);
System.out.println("Valor de la variable char: " + inicialApellido);
System.out.println("Valor de la variable String: " + nombreCompleto);

// Paso 3: Imprimir texto con nombre y edad
String nombre = "Juan";
int miEdad = 30;
System.out.println("Mi nombre es " + nombre + ", y tengo " + miEdad +
" años.");
}
}

```



Actividad: Seleccionando el Tipo de Dato Adecuado en Java

Datos:

1. La temperatura media de una ciudad durante el verano (en grados Celsius).
2. El precio de un producto en una tienda (en dólares y centavos).
3. El nombre completo de una persona.
4. La fecha de nacimiento de una persona.
5. El número de habitantes de una ciudad.
6. La duración de una película (en minutos).
7. Si un artículo está disponible o no en una tienda (true/false).
8. La distancia entre dos ciudades (en kilómetros).
9. El número de teléfono de una persona.
10. La cantidad de existencias de un producto en una tienda.

Instrucciones:

- A. Para cada dato proporcionado, selecciona el tipo de dato más adecuado en Java.
- B. Justifica tu elección explicando por qué crees que ese tipo de dato es el más apropiado para representar la información dada.

RESOLUCIÓN:

1. Temperatura media de una ciudad durante el verano: **double** (se permite decimal y cubre un amplio rango de valores posibles para la temperatura).
2. Precio de un producto en una tienda: **double** (se permite decimal y puede representar valores de dinero precisos).
3. Nombre completo de una persona: **String** (los nombres pueden contener caracteres alfanuméricos y no necesitan realizar cálculos aritméticos).

4. Fecha de nacimiento de una persona: **LocalDate** (los objetos LocalDate son adecuados para representar fechas y proporcionan métodos útiles para manipularlas).
5. Número de habitantes de una ciudad: **long** (un entero de 64 bits es adecuado para representar números grandes, como la población de una ciudad).
6. Duración de una película: **int** (los números enteros son adecuados para representar duraciones en minutos, ya que no se pueden tener fracciones de minutos).
7. Disponibilidad de un artículo en una tienda: **boolean** (los valores booleanos son adecuados para representar estados de verdadero o falso, en este caso, la disponibilidad del artículo).
8. Distancia entre dos ciudades: **double** (se permite decimal y puede representar distancias precisas en kilómetros).
9. Número de teléfono de una persona: **String** (los números de teléfono pueden contener caracteres especiales y no necesitan realizar cálculos aritméticos).
10. Cantidad de existencias de un producto en una tienda: **int** (los números enteros son adecuados para representar cantidades enteras como la cantidad de existencias de un producto).

💡 Te animamos a explorar las **diferencias entre los tipos de datos double y float** en Java, enfocándote en su precisión, rango de valores y las convenciones de uso vigentes en la industria de la programación. Mientras que float podría ser adecuado para representar datos con una precisión de hasta dos dígitos decimales, el tipo double ofrece una mayor precisión y un rango de valores más amplio. Esta característica puede resultar beneficiosa en aplicaciones comerciales donde la exactitud de los datos desempeña un papel crucial.