¡Bienvenido a tu tutorial interactivo sobre las Clases Wrapper en Java! En este tutorial, basado en el libro *Programación orientada a objetos en Java*, aprenderás sobre las clases wrapper y cómo utilizarlas en tus programas. El objetivo es que aprendas de manera activa y práctica, así que tendrás preguntas de opción múltiple, tareas y ejercicios prácticos. ¡Vamos a empezar!

**1. Introducción a las Clases Wrapper**

**Conceptos Clave**

En Java, las clases wrapper son clases que proporcionan una forma de envolver los tipos de datos primitivos en objetos. Cada tipo de dato primitivo tiene una clase wrapper correspondiente:

* **Byte** para byte
* **Short** para short
* **Integer** para int
* **Long** para long
* **Float** para float
* **Double** para double
* **Character** para char
* **Boolean** para boolean

Las clases wrapper ofrecen métodos útiles para convertir entre tipos primitivos y cadenas, así como para realizar operaciones relacionadas con los datos primitivos.

**Pregunta de Opción Múltiple 1**

¿Cuál de las siguientes clases wrapper se usa para envolver un valor float?

A) Double  
B) Float  
C) Character  
D) Integer

**2. Creación de Objetos Wrapper**

**Conceptos Clave**

Las clases wrapper proporcionan constructores para crear objetos a partir de valores primitivos. También puedes usar métodos estáticos para convertir cadenas a tipos primitivos y viceversa.

**Ejemplo:**

java

Copiar código

int numero = 42;

Integer numeroWrapper = Integer.valueOf(numero); // Creación de un objeto Integer

String numeroStr = "123";

int numeroConvertido = Integer.parseInt(numeroStr); // Conversión de String a int

**Tarea 1**

Escribe un programa que declare una variable de tipo double, cree un objeto Double a partir de ella, y convierta una cadena a un double usando Double.parseDouble(). Ejemplo:

java

Copiar código

public class WrapperExample {

public static void main(String[] args) {

double valor = 10.5;

Double valorWrapper = Double.valueOf(valor);

String valorStr = "25.75";

double valorConvertido = Double.parseDouble(valorStr);

System.out.println("Valor original (double): " + valor);

System.out.println("Valor convertido desde String (double): " + valorConvertido);

}

}

**Pregunta de Opción Múltiple 2**

¿Qué método se usa para convertir una cadena en un objeto Integer?

A) Integer.parseString()  
B) Integer.parseInt()  
C) Integer.valueOf()  
D) Integer.toString()

**3. Uso de Métodos en Clases Wrapper**

**Conceptos Clave**

Las clases wrapper tienen métodos útiles que te permiten manipular datos. Por ejemplo:

* **intValue()**: Devuelve el valor primitivo correspondiente del objeto wrapper.
* **toString()**: Convierte el valor primitivo a una cadena.
* **compareTo()**: Compara dos objetos wrapper.

**Ejemplo:**

java

Copiar código

Integer num1 = 10;

Integer num2 = 20;

int diferencia = num1.compareTo(num2); // Compara dos Integer

String numStr = num1.toString(); // Convierte Integer a String

int numPrimitivo = num1.intValue(); // Obtiene el valor int del Integer

**Tarea 2**

Crea un programa que compare dos valores Integer usando el método compareTo(), convierta un Integer a String usando toString(), y obtenga el valor primitivo usando intValue(). Ejemplo:

java

Copiar código

public class WrapperMethods {

public static void main(String[] args) {

Integer num1 = 45;

Integer num2 = 30;

int comparacion = num1.compareTo(num2);

String numStr = num1.toString();

int numPrimitivo = num1.intValue();

System.out.println("Comparación: " + comparacion);

System.out.println("Número como String: " + numStr);

System.out.println("Valor primitivo: " + numPrimitivo);

}

}

**Pregunta de Opción Múltiple 3**

¿Qué método se utiliza para convertir un objeto Integer en una cadena (String)?

A) Integer.toString()  
B) Integer.valueOf()  
C) Integer.toValue()  
D) Integer.parseString()

**4. Autoboxing y Unboxing**

**Conceptos Clave**

Java realiza automáticamente la conversión entre tipos primitivos y sus correspondientes clases wrapper, conocida como autoboxing y unboxing.

* **Autoboxing**: Conversión automática de un tipo primitivo a su correspondiente clase wrapper.
* **Unboxing**: Conversión automática de un objeto wrapper a su tipo primitivo.

**Ejemplo:**

java

Copiar código

Integer num = 100; // Autoboxing

int valor = num; // Unboxing

**Tarea 3**

Escribe un programa que demuestre autoboxing y unboxing con un Integer y un Double. Ejemplo:

java

Copiar código

public class AutoboxingUnboxing {

public static void main(String[] args) {

Integer num = 50; // Autoboxing

int valor = num; // Unboxing

Double d = 12.34; // Autoboxing

double valorDouble = d; // Unboxing

System.out.println("Número entero: " + valor);

System.out.println("Número double: " + valorDouble);

}

}

**Pregunta de Opción Múltiple 4**

¿Qué es el autoboxing en Java?

A) Conversión automática de un tipo primitivo a un objeto wrapper  
B) Conversión automática de un objeto wrapper a un tipo primitivo  
C) Conversión manual de un tipo primitivo a un objeto wrapper  
D) Conversión manual de un objeto wrapper a un tipo primitivo

**5. Resumen y Evaluación**

**Resumen**

* **Clases Wrapper**: Proveen una forma de envolver tipos primitivos en objetos y ofrecen métodos útiles para manipulación y conversión.
* **Autoboxing y Unboxing**: Procesos automáticos de conversión entre tipos primitivos y sus clases wrapper.
* **Métodos Comunes**: parseXXX(), valueOf(), intValue(), toString(), etc.

**Ejercicio Final**

Crea un programa que utilice las clases wrapper para realizar las siguientes tareas:

1. Convertir un char a Character y obtener su valor ASCII.
2. Comparar dos valores Double y mostrar el resultado.
3. Convertir un Boolean a String y viceversa.

java

Copiar código

public class WrapperFinal {

public static void main(String[] args) {

char letra = 'A';

Character charWrapper = letra;

int ascii = (int) charWrapper.charValue();

Double d1 = 5.67;

Double d2 = 5.67;

int comparacionDouble = d1.compareTo(d2);

Boolean booleano = true;

String booleanStr = booleano.toString();

Boolean booleanConvertido = Boolean.parseBoolean(booleanStr);

System.out.println("Valor ASCII del carácter: " + ascii);

System.out.println("Comparación Double: " + comparacionDouble);

System.out.println("Boolean como String: " + booleanStr);

System.out.println("Boolean convertido desde String: " + booleanConvertido);

}

}

**Pregunta Final**

¿Qué método se usa para convertir un char a un Character y obtener el valor ASCII?

A) charValue()  
B) Character.toString()  
C) Character.getNumericValue()  
D) Character.getValue()

**Sección de Preguntas y Respuestas Correctas**

1. **Pregunta de Opción Múltiple 1**
   * **Respuesta Correcta:** B) Float
2. **Pregunta de Opción Múltiple 2**
   * **Respuesta Correcta:** B) Integer.parseInt()
3. **Pregunta de Opción Múltiple 3**
   * **Respuesta Correcta:** A) Integer.toString()
4. **Pregunta de Opción Múltiple 4**
   * **Respuesta Correcta:** A) Conversión automática de un tipo primitivo a un objeto wrapper
5. **Pregunta Final**
   * **Respuesta Correcta:** C) Character.getNumericValue()