**Repaso de términos de BASES DE DATOS**

1. Base de datos: Es una colección estructurada de información organizada en tablas o archivos, que se almacenan en un sistema informático y se utilizan para gestionar, organizar y recuperar datos de manera eficiente.
2. Tabla: Es una estructura de datos que representa una colección de registros organizados en filas y columnas. Cada columna tiene un nombre y un tipo de dato, y cada fila contiene datos relacionados.
3. Registro: Es una instancia o una entrada específica en una tabla de una base de datos. También se le conoce como fila y contiene un conjunto de valores de datos relacionados.
4. Campo: Es la unidad más pequeña de datos en una tabla. También se le conoce como columna y representa un atributo específico de un registro, como un nombre, una fecha o un número.
5. Clave primaria: Es un campo o conjunto de campos en una tabla que identifica de manera única a cada registro. Se utiliza para asegurar la integridad y la unicidad de los datos en una tabla.
6. Clave foránea: Es un campo en una tabla que se relaciona con la clave primaria de otra tabla. Se utiliza para establecer relaciones entre tablas y mantener la integridad referencial en una base de datos.
7. Consulta: Es una solicitud de información a una base de datos. Permite recuperar, filtrar y manipular datos almacenados en una base de datos para obtener información específica.
8. Índice: Es una estructura de datos que se crea en una o varias columnas de una tabla para acelerar la búsqueda y recuperación de datos. Mejora el rendimiento de las consultas en una base de datos.
9. Normalización: Es el proceso de diseñar una base de datos de manera eficiente para eliminar redundancias y asegurar la integridad y la consistencia de los datos. Se divide en diferentes formas normalizadas, como la primera forma normal (1NF), segunda forma normal (2NF) y tercera forma normal (3NF).
10. Transacción: Es una unidad de trabajo en una base de datos que involucra una o varias operaciones, como inserciones, actualizaciones o eliminaciones de datos. Las transacciones se utilizan para asegurar la integridad y la consistencia de los datos en una base de datos.

**Que es Multitarea en java**

La multitarea, también conocida como concurrencia, se refiere a la capacidad de un programa para realizar varias tareas simultáneamente. En Java, la multitarea se puede lograr mediante la implementación de hilos (threads). Un hilo es una secuencia de instrucciones que se ejecuta de forma independiente en el contexto de un programa.

package ejemplo.de.hilos;

public class EjemploMultitarea {

public static void main(String[] args) {

// Crear dos objetos de la clase HiloEjemplo

HiloEjemplo hilo1 = new HiloEjemplo("Hilo 1");

HiloEjemplo hilo2 = new HiloEjemplo("Hilo 2");

// Iniciar los hilos

hilo1.start();

hilo2.start(); }

// Clase que extiende de Thread para crear un hilo

static class HiloEjemplo extends Thread {

private String nombre;

public HiloEjemplo(String nombre) {

this.nombre = nombre; }

@Override

public void run() {

for (int i = 1; i <= 5; i++) {

System.out.println("Hola desde " + nombre + ", contador: " + i);

try {

// Dormir el hilo por un segundo para simular una tarea

Thread.sleep(1000);

} catch (InterruptedException e) {

e.printStackTrace();

}

}

}

}

}

**Que son los HILOS Threads- en programación (java)**

Los hilos (threads) en programación se refieren a la capacidad de un programa para realizar múltiples tareas de forma concurrente y simultánea. Un hilo es una secuencia de instrucciones que se ejecuta de manera independiente dentro de un programa, compartiendo los mismos recursos del sistema, como la memoria y los archivos, con otros hilos en ejecución.

En Java, los hilos se implementan utilizando la clase **Thread**, que es parte de la biblioteca estándar de Java. Los hilos permiten que un programa realice tareas en paralelo, lo que puede mejorar la eficiencia y el rendimiento de una aplicación al aprovechar los recursos del sistema de manera más efectiva.

Los hilos en Java tienen las siguientes características:

1. Multitarea: Los hilos permiten que una aplicación realice múltiples tareas de forma concurrente. Cada hilo puede ejecutar su propio flujo de ejecución independiente, lo que permite que varias tareas se realicen simultáneamente.
2. Compartición de recursos: Los hilos comparten el mismo espacio de memoria y los mismos recursos del sistema, como archivos y conexiones de red, lo que permite la comunicación y la cooperación entre los hilos.
3. Cambio de contexto: El sistema operativo o la máquina virtual de Java se encargan de administrar la ejecución de los hilos, alternando entre ellos en pequeños intervalos de tiempo. Este proceso se conoce como cambio de contexto, y permite que los hilos se ejecuten aparentemente al mismo tiempo.
4. Sincronización: Debido a que los hilos comparten recursos, puede haber problemas de sincronización si dos o más hilos acceden y modifican los mismos datos simultáneamente. Es importante utilizar técnicas de sincronización, como bloqueos (locks) o monitores, para asegurar la consistencia y la integridad de los datos compartidos.

Los hilos son una herramienta poderosa en la programación en Java para lograr la multitarea y la concurrencia en aplicaciones. Sin embargo, también pueden introducir complejidad y desafíos adicionales, como condiciones de carrera y problemas de sincronización. Por lo tanto, es importante comprender bien los conceptos de hilos y utilizar las mejores prácticas de programación concurrente para asegurar un funcionamiento correcto y seguro de los hilos en una aplicación Java.