

Ejercicio 1 - Entradas de baloncesto (5 puntos)

Tenemos una aplicación que permite gestionar la venta de entradas para los partidos de un torneo de baloncesto a través de una plataforma online.

La empresa organizadora quiere probar la carga del sistema bajo condiciones de estrés, por lo que nos ha solicitado una simulación del proceso de venta de entradas.

Nuestro programa simulará la venta de 100 entradas para 10 clientes simultáneos durante 5 minutos (300 segundos). Cada cliente intentará comprar una entrada cada 5 segundos.

Los clientes se identificarán con un número de cliente (secuencial).

Importante:

Para realizar la simulación debéis utilizar intervalos de tiempo proporcionales a los indicados en el enunciado, de forma que nos permitan lanzar la simulación y observar el comportamiento de la aplicación sin tener que esperar 5 minutos cada vez.

Los clientes, para cada compra, solicitarán un número de entrada **(aleatorio)** que intentan comprar.

- Si la entrada está disponible, se le asignará y se mostrará un mensaje de confirmación.
- Si la entrada ya ha sido vendida, se mostrará un mensaje indicando esta circunstancia. El cliente se quedará realizando una espera no activa. Cuando salga de la espera, volverá a intentar comprar una entrada diferente, repitiendo el proceso hasta que consiga comprar una entrada.

Tips:

Los mensajes deben ser lo suficiente claros para identificar el número de cliente, la entrada solicitada y el resultado de la operación.

El programa debe ser capaz de gestionar la concurrencia de los clientes y la disponibilidad de las entradas.

La simulación finalizará cuando todas las entradas estén vendidas o bien cuando haya transcurrido el tiempo de simulación.

Antes de finalizar, cada cliente mostrará las entradas que ha conseguido comprar, indicando el número de cada una de las entradas adquiridas y el total de entradas conseguidas.

Cuando todos los clientes hayan finalizado, la aplicación mostrara un resumen con el número de entradas vendidas, el número de colisiones que se han producido y el tiempo total de ejecución.

Ayuda:

Para calcular el tiempo de la simulación podéis utilizar el método `currentTimeMillis()` de la clase `System`.

Con este método podéis calcular el tiempo de ejecución de la simulación tomando un valor al iniciar la simulación y un valor al finalizarla. Restando ambos valores obtendréis el tiempo total de ejecución.

Ejercicio 2 - Gestión de usuarios (5 puntos)

Queremos hacer una aplicación C/S que simule el funcionamiento de un cliente y un servidor de base de datos. La aplicación mantendrá un conjunto de registros de usuarios y controlará las altas y bajas que realizamos, guardando esta información en memoria.

🔴 Instrucciones:

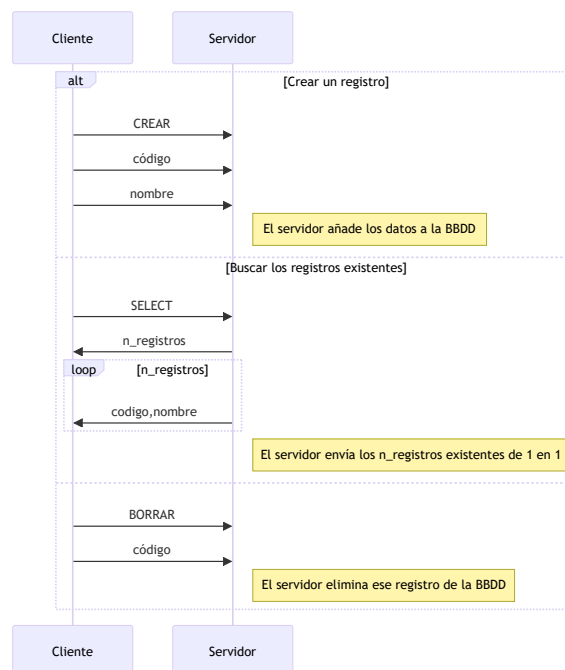
Hay datos precargados en memoria para realizar las pruebas

Abre los proyectos BDUsuariosServidor y BDUsuariosCliente

- En el cliente os doy hecha la parte de conexión con el servidor. Toda la lógica del protocolo la tenéis que programar vosotros.
- En el servidor
 - Clase Usuario: No hay que cambiar nada
 - Clase BDUsuarioServidor: Completad el código marcado con TO-DO

Añade las clases y el código necesario para crear un servidor multihilo que cumpla el siguiente protocolo:

Diagrama de secuencia



Todos los mensajes se pueden procesar en cualquier momento (**sin estados**)

Descripción del protocolo

CREAR

- Cliente
 - El cliente solicitará, por teclado, un código numérico y un nombre de usuario.
 - El cliente envía 3 mensajes al servidor, uno con el comando (CREAR), otro con el código y otro con el nombre
- Servidor
 - Esperará a recibir, en un mensaje, el código del usuario en la BD

- Esperará a recibir, en otro mensaje, el nombre del usuario en la BD
 - El servidor añade la información recibida en la BD (*colección de datos en memoria*)
 - No es necesario comprobar errores como códigos de usuario duplicados

SELECT

- Cliente
 - El cliente envía un mensaje al servidor con el comando (SELECT)
 - El cliente espera un mensaje con el número de respuestas que le enviará el servidor.
 - El cliente espera tantos mensajes como se le haya indicado en el mensaje anterior e irá mostrando por consola la información recibida con el formato:
`Código: xxxxx / Usuario: yyyyyy`
- Servidor
 - El servidor contesta con una línea que indica cuantos empleados hay en la BD.
 - A continuación envía un mensaje por cada registro que hay en la BD, con el formato:
`codigo,nombre`

BORRAR

- Cliente
 - El cliente solicitará, por teclado, un código numérico
 - El cliente envía 2 mensajes al servidor, uno con el comando (BORRAR) y otro con el código.
- Servidor
 - Esperará a recibir, en un mensaje, el código del usuario en la BD
 - El servidor elimina el registro de la BD
 - No es necesario comprobar errores como si el código no existe