



## Parte III

Dados los ficheros Cine.java, Pelicula.java y Sala.java de la parte anterior, se pide:

- Crear una base de datos en MySQL para representar las tablas de las clases Cine, Pelicula y Sala.
- Definir los métodos necesarios para su conexión.
- Define los métodos necesarios para realizar las siguientes acciones con la base de datos. **OJO:** define los métodos **SOLO** para la siguiente lista. Puedes modificar los métodos originales como creas conveniente; por ejemplo, la función BuscarPeliPorActor puede recibir el nombre de un actor y devolver un vector con los nombres de las películas (en lugar de recibir un objeto y devolver un vector de objetos). **IMPORTANTE**, en la BDA pondremos toda la información necesaria para que el sistema funcione (no usaremos mockups):
  - Para la clase **Cine**:
    - ListarPoblacionesConCine: accede a la BDA y devuelve un vector con todas las poblaciones que hay en la provincia que se pasa como parámetro.
    - BuscarPeliPorActor: accede a la BDA y devuelve en un vector el conjunto de películas que coinciden con el actor que se pasa como argumento.
    - HaySesion: dada una película y un día, accede a la BDA y devuelve un vector con las sesiones que hay de la película para ese día.
    - ComprarEntrada: dada una sesión y un número de entradas, se comprueba en la BDA que quedan entradas disponibles. En caso afirmativo, se reducen las entradas disponibles en esa sesión en base a la cantidad comprada. Al finalizar, la BDA se actualiza con el número de entradas disponibles.
  - Para la clase **Pelicula**:
    - addActor: se introduce en la BDA un nuevo actor para la película.
- Define casos de prueba para probar toda la funcionalidad del punto anterior. Las pruebas las puedes hacer, bien comparando variables o bien comparando datasets. Para el último caso, deberás usar la función para pasar el contenido de la BDA a un dataset de forma automática. Es posible que para hacer algunas de estas pruebas debas añadir más funcionalidad que la proporcionada en los ficheros iniciales. En concreto, se pide:
  - Para la clase **Cine**:
    - ListarPoblacionesConCine: probar que las poblaciones devueltas en el vector coincide con las poblaciones existentes en la BDA.
    - BuscarPeliPorActor: probar que las película devueltas en el vector coinciden con las películas existentes en la BDA.
    - HaySesion: probar que el vector de sesiones devueltas coincide con los datos existentes en la BDA.
    - ComprarEntrada: probar que para una compra mayor de las plazas existentes, devuelve falso. Además, probar que en caso de que hayan entradas suficientes, la BDA se actualiza restando el número de entradas compradas.
  - Para la clase **Pelicula**:
    - addActor: probar que el actor nuevo se ha introducido en la BDA.

