

Se que ya es tarde y quizás tu tengas todo hecho y aja, pero solo es algo que se me ocurrió...

Estuve pensando y creo que, bueno no se si ya avanzaste con el proyecto de estructura y talvez ya tengas una idea clara de que programar y como programar, probablemente hasta quizás ya hayas terminado el programa. De cualquier forma, quiero mostrarte este otro programa.

Hubiese querido decirte antes pero ayer estuve con el proyecto de tpi y no pude modificar las cosas, el programa es el que hicimos con mi grupo el año pasado, a nosotros nos toco conectar y buscar en una base de datos con arboles binarios, lo interesante de esta estructura es que los arboles binarios son considerados uno de los más eficiente y rápido a la hora de realizar búsquedas, de hecho, algunos de los buscadores están formados con esta estructura.

Entonces planteo que cuando como tu tema es simulación de colas de espera en el área de aduana en un aeropuerto, no especifica muy bien si son con aviones de carga o con personas, yo lo tome como personas por que igual aduana hace un registro de las maletas xd

Según un video que vi, primero el pasajero se registra y luego pasa con el personal de aduana a declarar lo que lleva en las maletas, le hacen un chequeo y todo eso, entonces ahí se forma una cola.

Entonces cuando tu vas al aeropuerto ya has hecho una reservación del vuelo, no solo llegas como hola voy a formarme, no entonces es ahí donde use la base de datos para tener un registro de los pasajeros, ya con eso utilice el árbol para hacer la búsqueda de la persona que llego a tomar el vuelo, entonces hago su registro y lo meto a la cola para que a continuación pase con aduana.

Mas o menos ese fue mi planteamiento y ya modifiqué el programa, pero por en sima si aun no has avanzado mucho en lo que tu habías pensado podrías trabajar en este y darle diseño y modificar algunas cosas que yo solo he puesto a la carrera.

A continuación, te muestro como es que funciona:

1. Primero estamos en la interfaz esperando que lleguen los pasajeros (La interfaz puede mejorarse claro si decides trabajar aquí, aunque para mejorarla hay que rehacer la estructura, porque no sé pero ese día andábamos bien locos con la marce para hacer así el frame xdx)

The screenshot shows a window titled 'Pasajeros'. It contains a search form with the following fields and buttons:

- Numero Pasaporte:** A text input field.
- Id:** A text input field.
- Nombre:** A text input field.
- Cantidad de Maletas:** A text input field.
- Buttons:** 'Buscar' (top right), 'Listar' (top right), 'Encolar' (middle right), and 'Terminar de Encolar' (bottom center).

The main area below the form is empty.

2. Luego en Numero de pasaporte se pondría el pasaporte del señor para verificar que si sea parte del vuelo xd y después se le da a buscar entonces si existe lo encuentra y nos pone el nombre y el pasaporte así mismo se nos habilita el ingresar cuantas maletas lleva el señor

The screenshot shows the 'Pasajeros' window after a search. The form fields are populated with the following data:

- Numero Pasaporte:** (Empty)
- Id:** 1
- Nombre:** Isabel Hayde
- Cantidad de Maletas:** (Redacted with a yellow box)

The 'Encolar' button is visible on the right. Below the form, the text 'Id buscado= 1' is displayed in the main area.

3. Después de eso se le da a encolar y manda ese señor a la cola y se sigue ingresando a esa persona a la cola y se continúa ingresando al resto de gente que llegue (Deje ese AreaBox para ir viendo que si se buscaran)

The 'Pasajeros' window contains the following fields and buttons:

- Numero Pasaporte: [Buscar] [Listar]
- Id:
- Nombre:
- Cantidad de Maletas: [Encolar]
- [Terminar de Encolar]

Below the form, a list of passengers is displayed with the following highlighted entries:

- Id buscado= 1
- Id buscado= 10
- Id buscado= 5
- Id buscado= 12
- Id buscado= 11

4. Luego se le da a terminar de Encolar y nos lleva a otra ventana donde se nos muestra el orden de llegada de las personas xd
 - a. Pasaporte 1 fue el primero en llegar

The 'Cola' window displays the 'Revision Pasajero Isabel Hayde' form with the following details:

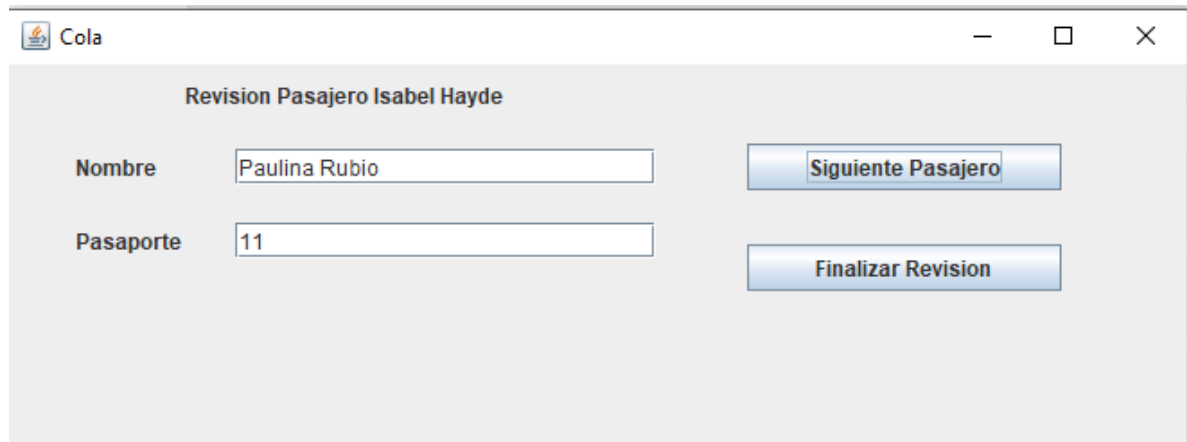
- Nombre: [Siguiente Pasajero]
- Pasaporte: [Finalizar Revision]

- b. Se le da siguiente Pasajero y el pasajero 10 fue el siguiente que ingresamos

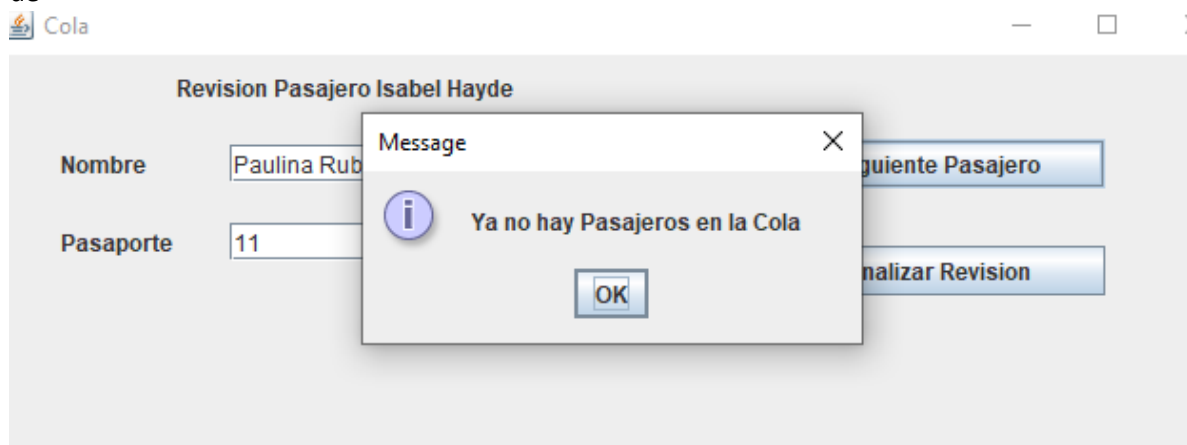
The 'Cola' window displays the 'Revision Pasajero Isabel Hayde' form with the following details:

- Nombre: [Siguiente Pasajero]
- Pasaporte: [Finalizar Revision]

- c. Y asi se sigue hasta que llegamos al ultimo pasajero que es el 11



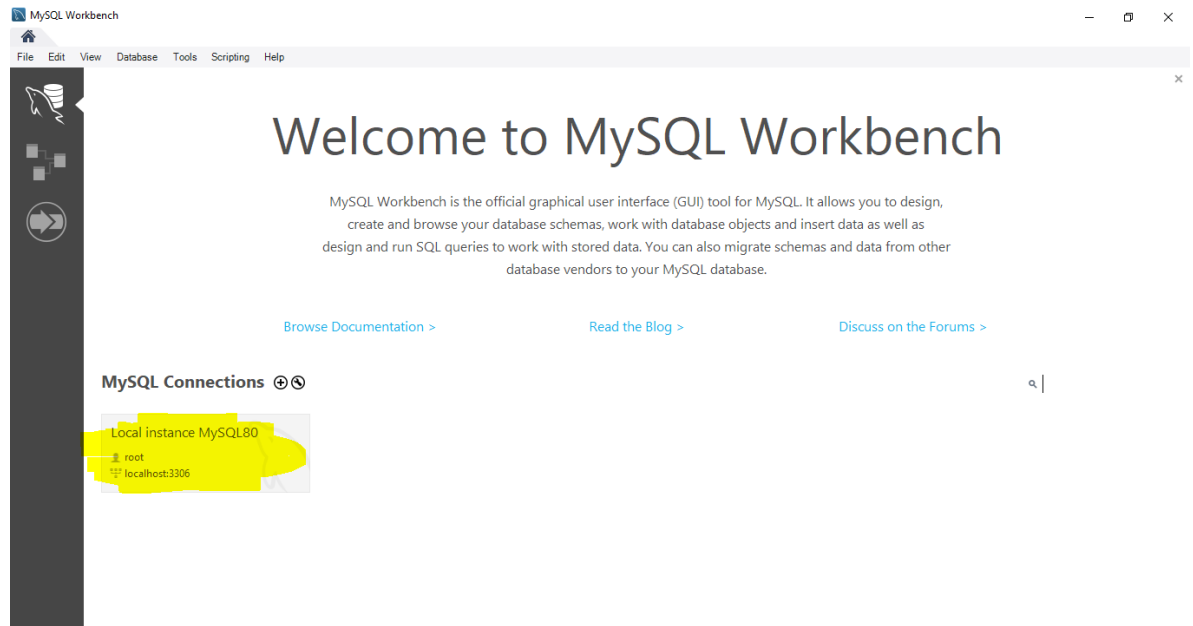
- d. Después de el ya no hay nadie y si se le da a siguiente pasajero nos da el mensaje de



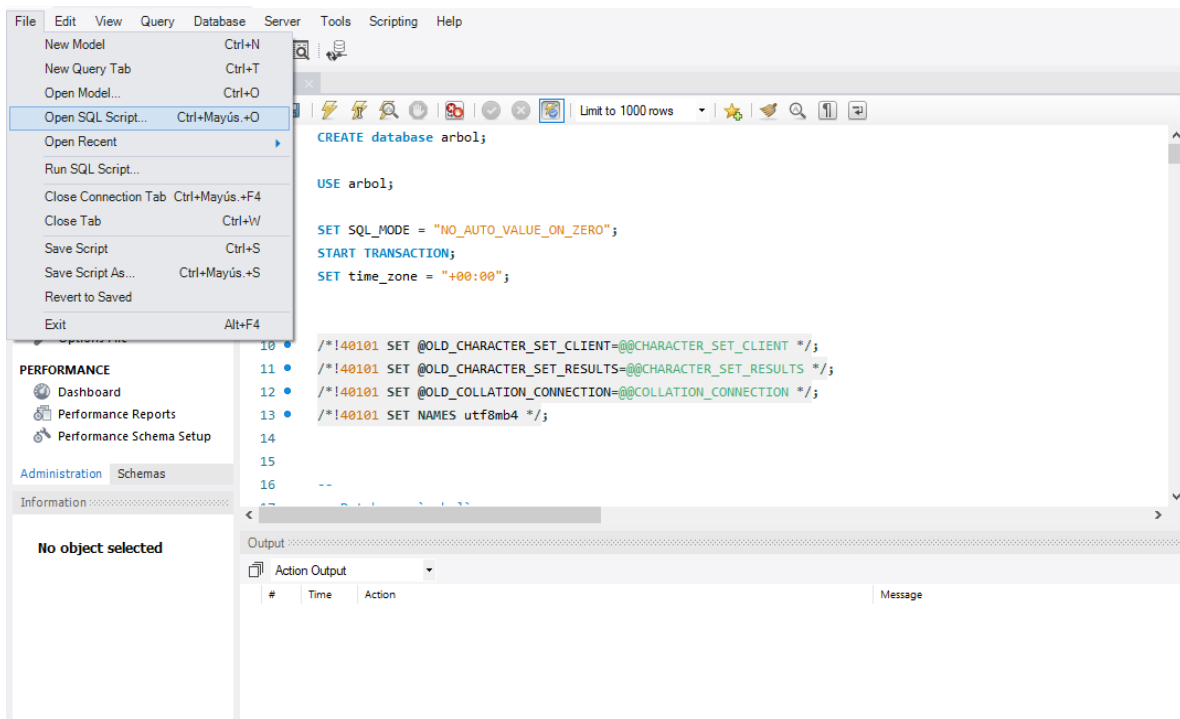
- e. Y de ahí cerrar el programa, finalizar revisión y ya eso hace jajaja xd

Si quieres probar y agarrar cosas de ahí para la idea que estes haciendo puedes hacerlo y probar para poder usarlo necesitar tener MySQL que es la base de datos que se utilizo :c

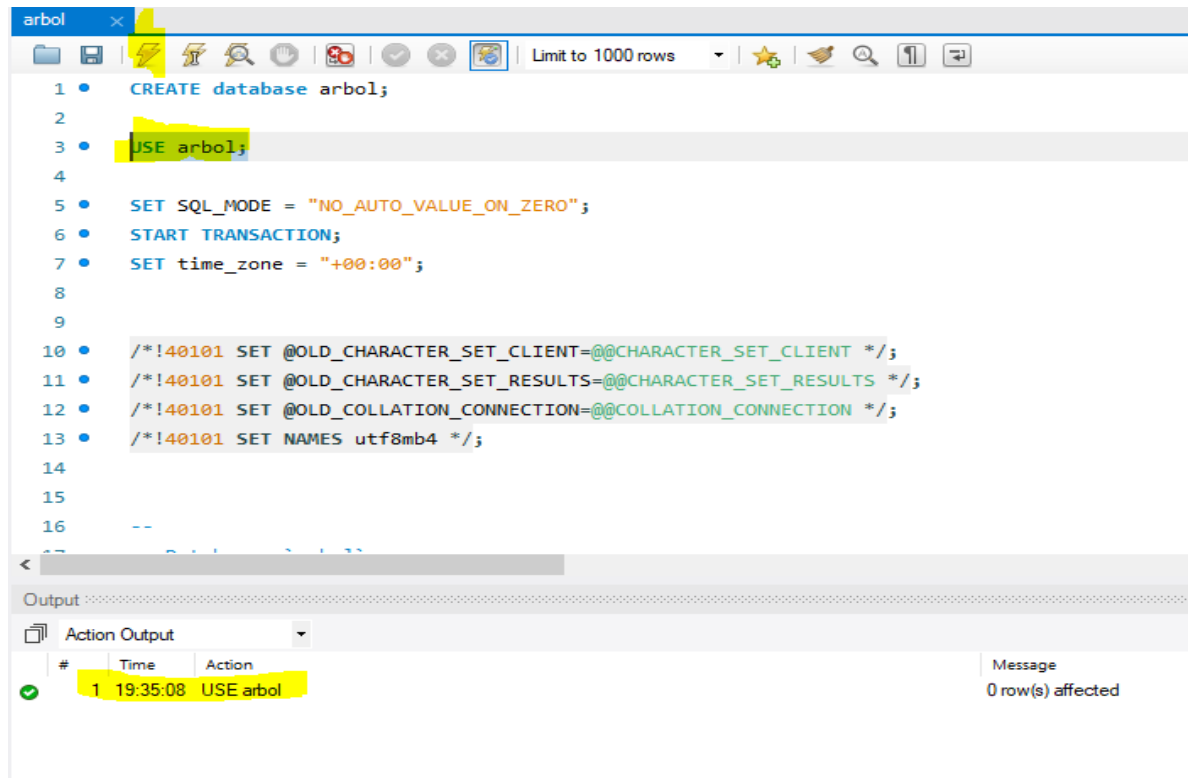
1. Cuando estes aquí dale clic donde señalo, ingresas tu contraseña y aja



2. Luego le das en open script sql y abres el archivo de arbol.sql



3. Luego te va salir y tienes que sombrear donde dice USE arbol y luego le das clic al rayito y abajo te saldrá así en verde entonces ya puedes ir a netbeans abrir el archivo



4. En el archivo BTreeRenderer.java tienes que encontrar el método que te señalo ahí y cambiar la contraseña a la contra que hayas puesto en mysql

```
BTTreeRenderer.java X
Source History
428
429 public void deleteKey(Integer key) {
430     String strVal = mTreeTest.getBTree().delete(key);
431     println("Delete key = " + key + " | value = " + strVal);
432 }
433
434
435 public void addKey(Integer key) {
436     String usuario = "root";
437     String clave = "1234";
438     String url = "jdbc:mysql://localhost:3306/arbol?serverTimezone=UTC";
439     Connection con;
440     Statement stmt;
441     ResultSet rs;
442     ArrayList<String> nuevos = new ArrayList<String>();
443
444     try {
445         Class.forName("com.mysql.cj.jdbc.Driver");
446     } catch (ClassNotFoundException ex) {
447         Logger.getLogger(BTreeRenderer.class.getName()).log(Level.SEVERE, null, ex);
448     }
449
450     try {
451         con= DriverManager.getConnection(url, usuario,clave);
452         stmt= con.createStatement();
453         rs= stmt.executeQuery("SELECT nombre FROM personasc");
454
455         while (rs.next()) {
```

5. Haces lo mismo en un método que esta un poco mas abajo, cambias la contraseña

```
BTTreeRenderer.java X
Source History
458     } catch (SQLException ex) {
459         Logger.getLogger(BTreeRenderer.class.getName()).log(Level.SEVERE, null, ex);
460     }
461     Random randomGenerator = new Random();
462     int index = randomGenerator.nextInt(nuevos.size());
463     println("Se añadio el usuario con id = " + key);
464     mTreeTest.getBTree().insert(key, nuevos.get(index));
465 }
466
467
468 public final void generateTestData() {
469     String usuario = "root";
470     String clave = "1234";
471     String url = "jdbc:mysql://localhost:3306/arbol?serverTimezone=UTC";
472     Connection con;
473     Statement stmt;
474     ResultSet rs;
475
476     try {
477         Class.forName("com.mysql.cj.jdbc.Driver");
478     } catch (ClassNotFoundException ex) {
479         Logger.getLogger(BTreeRenderer.class.getName()).log(Level.SEVERE, null, ex);
480     }
481
482     try {
483         con= DriverManager.getConnection(url, usuario,clave);
484         stmt= con.createStatement();
485         rs= stmt.executeQuery("SELECT nombre FROM personasc");
```

Y ahora si podrás ver como funciona

Como te digo no es obligación, solo te doy mi idea, pero tu y tu grupo son libres de hacer como quieran

Pd: Salen 4 advertencias porque no se :v y hay vario código que quite o bueno lo deje comentado porque no lo veo necesario para este caso xd