**Laboratorio 10**

**Comunicación Android con base de datos en la nube**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tipo de trabajo:** | **X** | **Individual** |  | **Grupal** |

**Fecha:** 15/01/2020

**Paralelo:** 101

**Nombre:** Bolívar Núñez Montoya

**Objetivo:**

* Realizar una conexión remota entre una aplicación de Android y una base relacional para adquisición de datos desde Internet.

**Duración:**

Ciento veinte (120) minutos.

**Materiales y Herramientas:**

* Software Android Studio

**Introducción**

Una base de datos remota es una base de datos que se encuentra físicamente en otro sistema de servidor, a diferencia de una base de datos local que se encuentra físicamente en el sistema del servidor local.

VENTAJAS

* Eliminar costos y esfuerzos de reclutamiento de TI.
* Potente, se puede ajustar la instancia de la base para permitir recursos completos, usar cachés de memoria más grandes y permitir más ciclos de CPU.
* Estable, el servidor de la base no afectará la disponibilidad de un servidor web local.

DESVENTAJAS

* Velocidad, el tiempo de ida y vuelta de una solicitud de conexión o de consulta a la base es superior al tiempo de una base local.
* Seguridad, no puede ser controlado por un administrador propio, sino que depende exclusivamente de la seguridad otorgada por el proveedor.

**Investigación  
Pregunta 1: ¿Con qué otro método es posible acceder a datos de Internet desde una app móvil?**

Mediante punto de acceso Wifi, por puerto USB y mediante la compartición de datos de internet por Bluetooth.

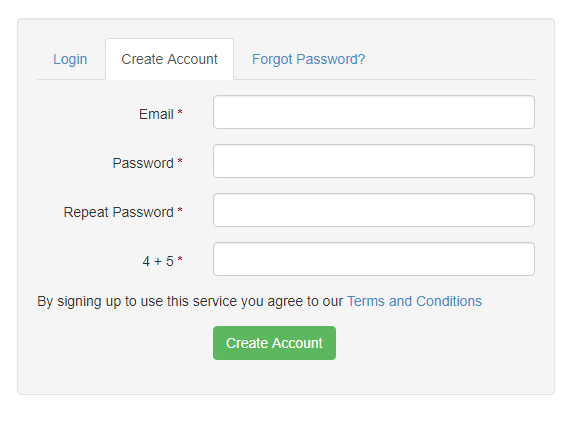
**Pregunta 2: Mencione al menos 3 páginas o servicios de internet que ofrecen el servicio de bases de datos remotas. (Exceptuando la utilizada en la práctica)**

<https://www.000webhost.com/> MongoDB PostgreSQL MariaDB

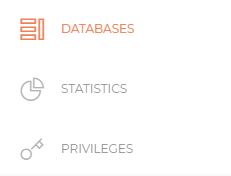
**Procedimiento**

**Paso 1:** Creación de una cuenta para el servicio de base de datos remoto

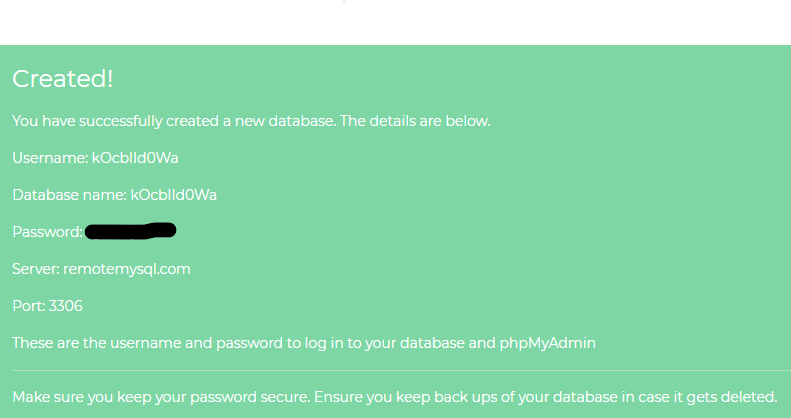
1. Acceder desde un navegador a la página <http://remotemysql.com>*.*
2. Seleccionar la opción *Login* y escoger la pestaña *Create Account*.



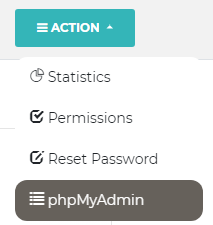
1. Llenar todos los campos e ingresar al sistema*.*
2. En el menú izquierdo seleccionar la opción *DATABASES* y crear una nueva base de datos.



1. Copiar la información referente a su nueva base de datos creada*.*



1. Dirigirse a la parte inferior y en el botón *Actions*, seleccionar *phpMyAdmin.*



1. Ingresar con las credenciales generadas*.*
2. Crear la tabla Lote con los siguientes campos:
   * id: int (PK)
   * descripcion: varchar50
   * precio\_unitario: float
   * cantidad: int
3. Crear la tabla Factura con los siguientes campos:
   * id: int (PK)
   * fecha\_emision: datetime
   * ruc\_cliente: varchar13
   * direccion\_cliente: varchar30
4. Crear la relación uno a muchos entre las tablas Lote y Factura.

Paso 2:Creación de la aplicación móvil

1. Crear un nuevo proyecto con el nombre *lab10.*
2. Diseñar un layout que contenga un textview(sin texto) que abarque el ancho de la aplicación y en la parte inferior un botón que sea Consultar.
3. En la clase main crear los atributos para cada parámetro de conexión a la base y un atributo de tipo textview, el cual debe ser asignado al componente que se diseñó en el literal anterior.

|  |  |
| --- | --- |
|  | private String serverIP = "remotemysql.com";  private String port = "3306";  private String userMySQL = " ";  private String pwdMySQL = " ";  private String database = " ";  private String[] datosConexion = null;  private TextView consulta; |

1. Crear una nueva clase llamada *AsyncQuery* que herede de la clase *AsyncTask<String[],Void,String[]>*
2. Crear los atributos de la clase.

private Connection conexionMySQL;  
private Statement st = null;  
private ResultSet rs = null;

1. Sobrescribir la función *doInBackground* con el siguiente código:

protected String[] doInBackground(String[]... datos) {  
 String sql = datos[0][5];  
 String resultadoSQL = "";  
 String[] totalResultadoSQL = null;  
 int numColumnas = 0;  
 int numFilas = 0;  
 String SERVIDOR = datos[0][0];  
 String PUERTO = datos[0][1];  
 String BD = datos[0][2];  
 String USUARIO = datos[0][3];  
 String PASSWORD = datos[0][4];  
  
 try{  
 conexionMySQL = DriverManager.*getConnection*("jdbc:mysql://" + SERVIDOR + ":" + PUERTO + "/" + BD,  
 USUARIO,PASSWORD);  
  
 st = conexionMySQL.createStatement();  
 rs = st.executeQuery(sql);  
 rs.last();  
 numFilas = rs.getRow();  
 if(numFilas == 0)  
 {  
 resultadoSQL = "No se ha producido ningún resultado. Revise la consulta realizada.\n";  
 }else  
 {  
 for(int i=1;i<=numColumnas;i++){  
 if(i>1) resultadoSQL += ",";  
 resultadoSQL += rs.getMetaData().getColumnName(i);  
 }  
 resultadoSQL += "\n";  
 rs.beforeFirst();  
 while (rs.next())  
 {  
 numColumnas = rs.getMetaData().getColumnCount();  
 for(int i=1;i<=numColumnas;i++){  
 if(i>1) resultadoSQL += ",";  
 resultadoSQL += rs.getString(i);  
 }  
 resultadoSQL += "\n";  
 }  
 }  
 totalResultadoSQL = new String[]{ resultadoSQL,String.*valueOf*(numFilas),String.*valueOf*(numColumnas)};  
  
 }catch(SQLException ex)  
 {  
 Log.*d*("Error de conexion", ex.getMessage());  
 }  
 finally  
 {  
 try  
 {  
 if(rs != null)  
 {  
 rs.close();  
 }  
 st.close();  
 conexionMySQL.close();  
 } catch (SQLException e)  
 {  
 e.printStackTrace();  
 }  
 }  
 return totalResultadoSQL;  
}

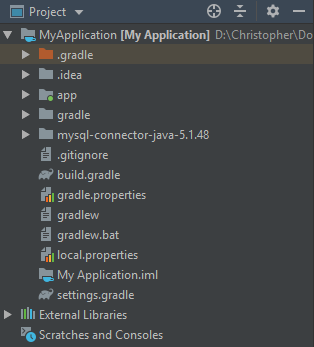
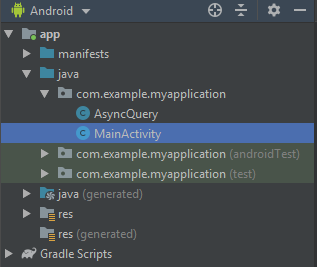
1. Crear una función llamada *mostrarResultados* en la clase Main para asignarle al botón.

public void mostrarResultados(View view)  
{  
 String[] resultadoSQL = null;  
 try{  
 datosConexion = new String[]{  
 serverIP,  
 port,  
 database,  
 userMySQL,  
 pwdMySQL,  
 "SELECT \* FROM Factura;"  
 };  
 String driver = "com.mysql.jdbc.Driver";  
 Class.*forName*(driver).newInstance();  
 resultadoSQL = new AsyncQuery().execute(datosConexion).get();  
 Toast.*makeText*(MainActivity.this,"Conexión Establecida", Toast.*LENGTH\_LONG*).show();  
  
 String resultadoConsulta = resultadoSQL[0];  
 String numFilas = resultadoSQL[1];  
 String numColumnas = resultadoSQL[2];  
 consulta.setText(resultadoConsulta + "\nNúmero de filas devueltas: " +  
 numFilas + "\nNúmero de columnas devueltas: " + numColumnas);  
 }catch(Exception ex)  
 {  
 Toast.*makeText*(this, "Error al obtener resultados de la consulta Transact-SQL: "  
 + ex.getMessage(), Toast.*LENGTH\_LONG*).show();  
 }  
}

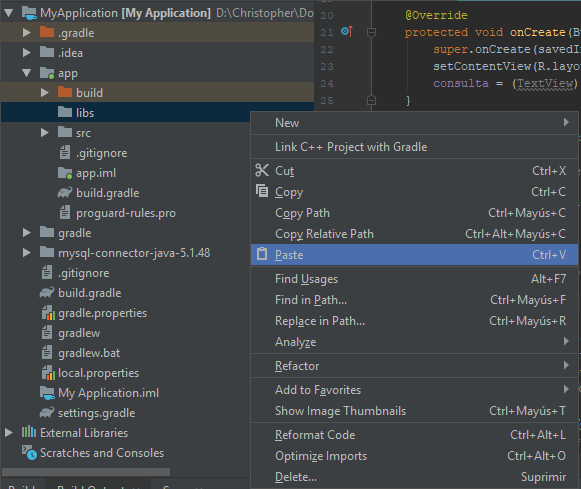
1. Dar permisos de acceso a Internet en el archivo *Manifest*.

Paso 3:Agregar la librería del conector MySQL

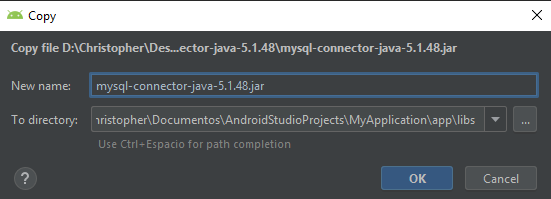
1. Descargar la librería desde el Sidweb *mysql-connector-java-5.1.48.jar*
2. Copiar el archivo (*CTRL+C*)
3. En el panel izquierdo cambiar el modo de vista de *Android* a *Project*



1. Clic derecho sobre la carpeta *app -> libs* y seleccionar pegar



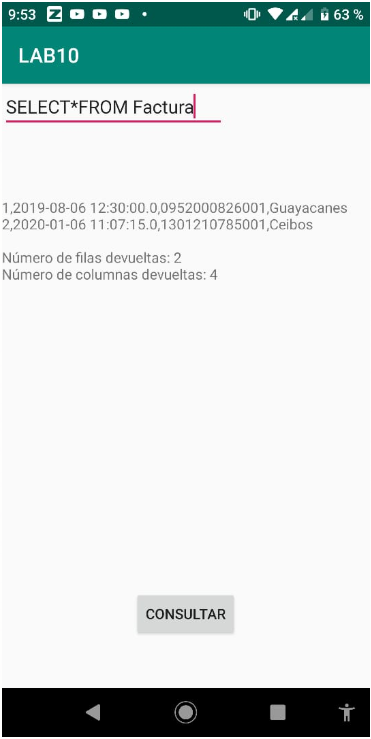
1. Clic en OK y ejecutar la aplicación



**Actividad 1:** Presente una captura del dispositivo real ejecutando la aplicación “lab10”.

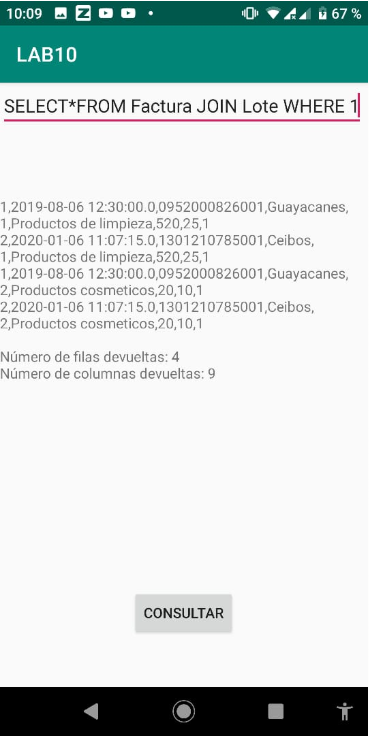


Actividad 2:Ingrese un campo EditText, el cual contendrá una consulta SQL ingresada por el usuario y cuando se presione el botón, la app ejecutará dicho comando SQL en lugar de ejecutar “SELECT \* FROM Factura;”



Actividad 3:En el componente EditText, ingresar el comando SQL para mostrar los datos de la factura con id=1 y todos los lotes correspondientes a dicha factura.

SELECT\*FROM Facturas JOIN Lote WHERE 1



**RETO (15 puntos extras): Modifique el diseño para que los resultados de la consulta no se muestren en un TextView, sino dentro de una tabla.**

**Conclusiones y Recomendaciones**

Conclusiones:

* + Se logro utilizar una base de datos remota para la adquisición de datos de Internet desde una aplicación móvil.
  + A pesar de que la base de datos es remota, el tiempo de respuesta es algo lento incluso teniendo una base de datos con solo dos tablas y un par de datos almacenados en cada una de ellas.

Recomendaciones:

* + Se puede utilizar este medio como base de datos remota para un proyecto básico, ya que el tiempo de respuesta depende de que tan pesada y compleja es la base de datos.
  + No ingresar variables de tablas cuando se quiere hacer una consulta ya que el código no valida según las variables sino según los datos.