

CLASE CONEXIÓN

**public class Conexion {**

//los propiedades static pueden ser accedidas directamente con el nombre de la clase

// sin tener que declarar un objeto

public static String ip = "192.168.1.158";

public static String baseDeDatos="bdtienda";

public static String driver="com.mysql.jdbc.Driver"; //driver o manejador de la base de datos

public static String usuario="sergio"; //usuario debe estar registrado con permisos en la base de datos

public static String password="12345"; //contraseña

public static Connection databaseConnection; //objeto para hacer la conexion

public static Statement statement; //objeto para hacer la consulta

**public static String hacerConexion() throws Exception{**

try{

//Cargar la clase driver

Class.forName(driver);

//Crear la conexion a traves de DriverManager

String sourceURL = "jdbc:mysql://"+ip+"/"+baseDeDatos;

databaseConnection=DriverManager.getConnection(sourceURL,usuario,password);

statement=databaseConnection.createStatement();

//Si se estable conexion

//System.out.println("---------Conexion establecida----------");

return "---------Conexion establecida----------";

//Si no se establece la conexion

}catch ( ClassNotFoundException cnfe ){

System.err.println(cnfe);

System.out.println("Conexion fallida sergio");

return "---------Conexion fallida----------";

}

catch( SQLException cnfe){

System.err.println(cnfe);

System.out.println("Conexion fallida sergio");

return "---------Conexion fallida----------";

}

}

**public static void cerrarConexion(){**

//cerrar la conexion

try{

databaseConnection.close();

System.out.println("---------Conexion cerrada----------");

}catch(SQLException ex){

System.err.println(ex.getSQLState());

System.err.println(ex.getMessage());

System.out.println("Error al cerrar la conexion");

}

**}**

//para los reportes y obtener los registros

**public static ResultSet ejecutaConsultaSQL(String Query){**

ResultSet rs=null;

try{

//Ejecutar Query SQL

rs= statement.executeQuery(Query);

//Obtener los metadatos de la tabla

ResultSetMetaData metadatos=rs.getMetaData();

//Mostrar el numero de columnas

int columnas=metadatos.getColumnCount();

System.out.println("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

System.out.println(Query);

System.out.println("Esta tabla tiene: " + columnas+ " columnas");

//Mostrar el nombre de los campos de la tabla

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

System.out.print(metadatos.getColumnName(i)+" ");

}

System.out.println("\n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

System.out.println("\*\*\*\*\*\*\*\*R E G I S T R O S\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

//Mostrar los datos de la tabla

while(rs.next()){

System.out.println(" ");

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

System.out.print(rs.getString(i)+" ");

}

**}**

System.out.println("\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

rs.beforeFirst(); // Ojo: si no lo pongo, no comienzo en el primero y

//necesito retornar un rs que inicie en el primero registro

}catch(SQLException ex){

System.err.println(ex.getSQLState());

System.err.println(ex.getMessage());

System.out.println("Error al hacer consulta");

}

return rs;

}

//se usa para agregar borrar y modificar

**public static void ejecutarActualizacionSQL(String Query) {**

try{

statement.executeUpdate(Query);

System.out.println("Se ejecuto la actualizacion correctamente");

}catch(SQLException ex){

System.err.println(ex.getSQLState());

System.err.println(ex.getMessage());

System.out.println("Error al ejecutar la actualizaion");

}

}

**} //Cierro llave principal de la CLASE CONEXIÓN**

CLASE ROPA

**public class Ropa {**

**//1.-atributos**

private int id=0;

private String descripcion="";

private double precio= 0.0;

private int cantidad = 0;

private String temporada="";

//atributo especial para generar ID

public static int ID\_unico=0;

**//2.-metodos constructor**

**public Ropa(){**

this.id= ID\_unico++;

**}**

**//3 .- METODOS SET Y GET PARA LA ENCAPSULACION**

//3.-Encapsulacion de los atributos (metodos get y set)

// public void setID(int dato){

// this.id = dato;

// }

public void setDescripcion(String dato){

this.descripcion=dato;

}

public void setPrecio(double dato){

this.precio=dato;

}

public void setCantidad(int dato){

this.cantidad=dato;

}

public void setTemporada(String dato){

this.temporada=dato;

}

public void setID(int id){

this.id = id;

}

public String getTemporada(){

return this.temporada;

}

public int getID(){

return this.id;

}

public String getDescripcion(){

return this.descripcion;

}

public double getPrecio(){

return this.precio;

}

public int getCantidad(){

return this.cantidad;

}

**//4.-Metodos del negocio**

public String obtenerInformacion(){

//proceso

String informacion = "id camisa: " + this.id + "\n" +

"precio: " + this.precio + "\n" +

"cantidad: " + this.cantidad + "\n" +

"descripcion: " + this.descripcion;

//salida

return informacion;

}

**}**