/\*

\* Pueda usar esta plantilla para la carga del reto a iMaster

\* Copie las clases de los paquetes Modelo, Vista, Controlador y Util

\* No incluya los import de los archivos .java solo las clases

\*/

// Importaciones necesarias en iMaster

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import java.util.ArrayList;

// Util (No modificar)

class JDBCUtilities {

private static final String DATABASE\_LOCATION = "ProyectosConstruccion.db";

public static Connection getConnection() throws SQLException {

String url = "jdbc:sqlite:"+DATABASE\_LOCATION;

return DriverManager.getConnection(url);

}

}

// Remplace en adelante por las clases de sus archivos .java

// Vista

public class ViewRequerimientos {

public static final ControllerRequerimientos controlador = new ControllerRequerimientos();

public static void requerimiento1(){

try {

ArrayList<Requerimiento\_1> listaRequemiento1;

listaRequemiento1 = controlador.consultarRequerimiento1();

for (Requerimiento\_1 req1 : listaRequemiento1) {

System.out.println(

req1.getFechaCompra() + " " +

req1.getCodigoCompra() + " " +

req1.getNombreProveedor() + " " +

req1.getValorPagado()

);

}

} catch (Exception e) {

System.err.println(e);

}

}

public static void requerimiento2(){

try {

ArrayList<Requerimiento\_2> listaRequerimiento2;

listaRequerimiento2 = controlador.consultarRequerimiento2();

for (Requerimiento\_2 req2 : listaRequerimiento2) {

System.out.println(

req2.getFechaInicio() + " " +

req2.getNombreCiudad() + " " +

req2.getNombreConstructora() + " " +

req2.getNombreLider() + " " +

req2.getCodigoTipo() + " " +

req2.getEstrato()

);

}

} catch (Exception e) {

System.err.println(e);

}

}

public static void requerimiento3(){

try {

ArrayList<Requerimiento\_3> listaRequerimiento3;

listaRequerimiento3 = controlador.consultarRequerimiento3();

for (Requerimiento\_3 req3 : listaRequerimiento3) {

System.out.println(

req3.getNombreAbreviado()

);

}

} catch (Exception e) {

System.err.println(e);

}

}

}

// Controlador

public class ControllerRequerimientos {

private final RequerimientoDao\_1 requerimientoDao\_1 = new RequerimientoDao\_1();

private final RequerimientoDao\_2 requerimientoDao\_2 = new RequerimientoDao\_2();

private final RequerimientoDao\_3 requerimientoDao\_3 = new RequerimientoDao\_3();

public ArrayList<Requerimiento\_1> consultarRequerimiento1()

throws SQLException {

return this.requerimientoDao\_1.requerimiento1();

}

public ArrayList<Requerimiento\_2> consultarRequerimiento2()

throws SQLException {

return this.requerimientoDao\_2.requerimiento2();

}

public ArrayList<Requerimiento\_3> consultarRequerimiento3()

throws SQLException {

return this.requerimientoDao\_3.requerimiento3();

}

}

// Modelo

// VO

public class Requerimiento\_1 {

// Operaciones de la clase requerimiento 1 -- su código

private String fechaCompra;

private String codigoCompra;

private String nombreProveedor;

private String valorPagado;

public Requerimiento\_1() {

}

public Requerimiento\_1(String fechaCompra, String codigoCompra, String nombreProveedor, String valorPagado) {

this.fechaCompra = fechaCompra;

this.codigoCompra = codigoCompra;

this.nombreProveedor = nombreProveedor;

this.valorPagado = valorPagado;

}

public String getFechaCompra() {

return fechaCompra;

}

public void setFechaCompra(String fechaCompra) {

this.fechaCompra = fechaCompra;

}

public String getCodigoCompra() {

return codigoCompra;

}

public void setCodigoCompra(String codigoCompra) {

this.codigoCompra = codigoCompra;

}

public String getNombreProveedor() {

return nombreProveedor;

}

public void setNombreProveedor(String nombreProveedor) {

this.nombreProveedor = nombreProveedor;

}

public String getValorPagado() {

return valorPagado;

}

public void setValorPagado(String valorPagado) {

this.valorPagado = valorPagado;

}

}

public class Requerimiento\_2 {

// Operaciones de la clase requerimiento 2 -- su código

private String fechaInicio;

private String nombreCiudad;

private String nombreConstructora;

private String nombreLider;

private String codigoTipo;

private String estrato;

public Requerimiento\_2() {}

public Requerimiento\_2(

String fechaInicio,

String nombreCiudad,

String nombreConstructora,

String nombreLider,

String codigoTipo,

String estrato

) {

this.setFechaInicio(fechaInicio);

this.setNombreCiudad(nombreCiudad);

this.setNombreConstructora(nombreConstructora);

this.setNombreLider(nombreLider);

this.setCodigoTipo(codigoTipo);

this.setEstrato(estrato);

}

public String getFechaInicio() {

return fechaInicio;

}

public void setFechaInicio(String fechaInicio) {

this.fechaInicio = fechaInicio;

}

public String getNombreCiudad() {

return nombreCiudad;

}

public void setNombreCiudad(String nombreCiudad) {

this.nombreCiudad = nombreCiudad;

}

public String getNombreConstructora() {

return nombreConstructora;

}

public void setNombreConstructora(String nombreConstructora) {

this.nombreConstructora = nombreConstructora;

}

public String getNombreLider() {

return nombreLider;

}

public void setNombreLider(String nombreLider) {

this.nombreLider = nombreLider;

}

public String getCodigoTipo() {

return codigoTipo;

}

public void setCodigoTipo(String codigoTipo) {

this.codigoTipo = codigoTipo;

}

public String getEstrato() {

return estrato;

}

public void setEstrato(String estrato) {

this.estrato = estrato;

}

}

public class Requerimiento\_3 {

// Operaciones de la clase requerimiento 3 -- su código

private String nombreAbreviado;

public Requerimiento\_3() {}

public String getNombreAbreviado() {

return nombreAbreviado;

}

public void setNombreAbreviado(String nombreAbreviado) {

this.nombreAbreviado = nombreAbreviado;

}

public Requerimiento\_3(String nombreAbreviado) {

this.setNombreAbreviado(nombreAbreviado);

}

}

// DAO

public class RequerimientoDao\_1 {

public ArrayList<Requerimiento\_1> requerimiento1() throws SQLException {

ArrayList<Requerimiento\_1> result = new ArrayList<Requerimiento\_1>();

Connection conn = null;

PreparedStatement stmt = null;

ResultSet rs = null;

try {

String sql =

"SELECT c.Fecha AS 'Fecha Compra',c.Proveedor AS 'Nom. Proveedor', c.ID\_Compra AS 'Codigo Compra' , c.Pagado " +

"FROM Compra c ORDER BY c.ID\_Compra DESC LIMIT 15;";

conn = JDBCUtilities.getConnection();

stmt = conn.prepareStatement(sql);

rs = stmt.executeQuery();

while (rs.next()) {

Requerimiento\_1 requerimiento1 = new Requerimiento\_1();

requerimiento1.setFechaCompra(rs.getString("Fecha Compra"));

requerimiento1.setNombreProveedor(rs.getString("Nom. Proveedor"));

requerimiento1.setCodigoCompra(rs.getString("Codigo Compra"));

requerimiento1.setValorPagado(rs.getString("Pagado"));

result.add(requerimiento1);

}

rs.close();

stmt.close();

} catch (SQLException ex) {

System.out.println(

"Error en la consulta del requerimiento 1 " + ex.getMessage()

);

} finally {

if (conn != null) {

conn.close();

}

}

return result;

}

}

public class RequerimientoDao\_2 {

public ArrayList<Requerimiento\_2> requerimiento2() throws SQLException {

ArrayList<Requerimiento\_2> result = new ArrayList<Requerimiento\_2>();

Connection conn = null;

PreparedStatement stmt = null;

ResultSet rs = null;

try {

String sql =

"SELECT p.Fecha\_Inicio AS 'Fec. Ini.', p.Ciudad AS 'Nombre Ciudad'," +

"p.Constructora AS 'Nom. Constructora', l.Nombre ||' '|| l.Segundo\_Apellido AS 'Nombre Lider'," +

"t.Codigo\_Tipo AS 'Codigo Tipo',t.Estrato " +

"FROM Proyecto p " +

"INNER JOIN Lider l ON l.ID\_Lider = p.ID\_Lider " +

"INNER JOIN Tipo t ON t.ID\_Tipo = p.ID\_Tipo " +

"WHERE p.Fecha\_Inicio BETWEEN '2020-01-01' AND '2020-07-09' " +

"AND p.Ciudad = 'Ibague'";

conn = JDBCUtilities.getConnection();

stmt = conn.prepareStatement(sql);

rs = stmt.executeQuery();

while (rs.next()) {

Requerimiento\_2 requerimiento2 = new Requerimiento\_2();

requerimiento2.setFechaInicio(rs.getString("Fec. Ini."));

requerimiento2.setNombreCiudad(rs.getString("Nombre Ciudad"));

requerimiento2.setNombreConstructora(rs.getString("Nom. Constructora"));

requerimiento2.setNombreLider(rs.getString("Nombre Lider"));

requerimiento2.setCodigoTipo(rs.getString("Codigo Tipo"));

requerimiento2.setEstrato(rs.getString("Estrato"));

result.add(requerimiento2);

}

rs.close();

stmt.close();

} catch (SQLException ex) {

System.out.println(

"Error en la consulta del requerimiento 2 " + ex.getMessage()

);

} finally {

if (conn != null) {

conn.close();

}

}

return result;

}

}

public class RequerimientoDao\_3 {

public ArrayList<Requerimiento\_3> requerimiento3() throws SQLException {

// Su código

ArrayList<Requerimiento\_3> result = new ArrayList<Requerimiento\_3>();

Connection conn = null;

PreparedStatement stmt = null;

ResultSet rs = null;

try {

String sql =

"SELECT SUBSTR(l.Nombre,1,3) || SUBSTR(l.Primer\_Apellido ,1,3) || SUBSTR(l.Segundo\_Apellido ,1,3) " +

"AS 'Abrev.' FROM Lider l ";

conn = JDBCUtilities.getConnection();

stmt = conn.prepareStatement(sql);

rs = stmt.executeQuery();

while (rs.next()) {

Requerimiento\_3 requerimiento3 = new Requerimiento\_3();

requerimiento3.setNombreAbreviado(rs.getString("Abrev."));

result.add(requerimiento3);

}

rs.close();

stmt.close();

} catch (SQLException ex) {

System.out.println(

"Error en la consulta del requerimiento 3 " + ex.getMessage()

);

} finally {

if (conn != null) {

conn.close();

}

}

return result;

}

}