

# Bases de Datos II – mayo 2022

NOTA: carilla como máximo para cada pregunta. Hay que aprobar la parte de JDBC, el resto hace media geométrica.

## Teoría

1. ¿Qué son las transacciones? ¿Qué propiedades tienen? ¿Quién es el responsable de cada propiedad?
2. Explicar aislamiento de instantáneas en el sistema de concurrencia. Definición y tipos (primera actualización gana y primer compromiso gana).
3. Explicar el uso del registro histórico en el sistema de recuperación y poner un ejemplo de su funcionamiento.
4. Optimización de consultas. Poner el esquema y explicarlo teniendo en cuenta todas las fases.
5. Almacenes y minería de datos. (explicar todo lo que supieras)

## JDBC

1. Qué es el autocommit, para qué sirve y escribir un ejemplo de su uso.
2. Ventajas del Preparedstatement con respecto a los otros. Ejemplo de uso.
3. Escribir un pseudocódigo de cómo harías para acceder a una tabla de una base de datos de la cual no conoces su estructura, pero conoces el nombre y como se haría para mostrar por pantalla los contenidos de esta tabla. (el enunciado era más o menos así pero no nos acordamos 100%) (Hablar de los metadatos)
4. Tipos de statement. (no olvidarse del Callable que nosotros nos olvidamos todos)

Statement, prepared, callable

## 3. Código en java para el ejercicio

```
public void meterDatos() {  
    //Obtener Datos  
  
    Connection con = DriverManager.getConnection(); //obtenemos la conexión  
  
    String consulta = "Select * from NombreTabla"; //conocemos el nombre de la tabla  
  
    Statement stmt = con.createStatement(consulta); //creamos el statement  
  
    ResultSet rs = stmt.executeQuery();  
  
    DefaultTableModel m = new DefaultTableModel(); //creamos modelo de tabla  
  
    ResultSetMetaData meta = rs.getMetaData(); //obtenemos las características de la tabla a través de los metaDatos  
  
    int numCol = meta.getColumnCount();  
  
    for (int i = 1; i <= numCol; i++) {
```

```
        m.addColumn(meta.getColumnName(i)); //añadimos las columnas a la tabla
    }
    while (rs.next()) { //recorremos las filas de la tabla
        Object[] obj = new Object[numCol]; //creamos un array que guardará los valores de las
        columnas
        for (int i = 0; i < numCol; i++) {
            obj[i] = rs.getObject(i + 1); //metemos los valores de las columnas
        }
        m.addRow(obj); //metemos las fila en la tabla
    }
    tabla.setModel(m); //tabla es una JTable, le asignamos el modelo que hemos creado
}
```