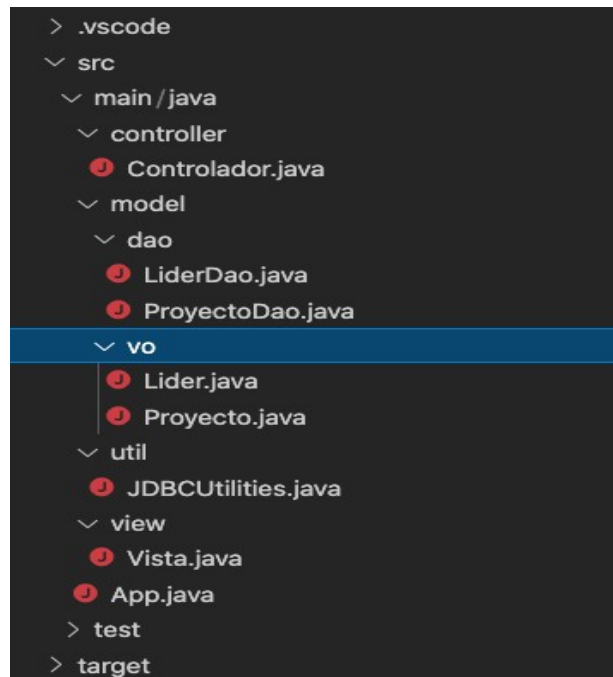


Reto 4

Descripción del problema:

La empresa de construcción en la cual fue contratado como DBA en el Reto 3 le ha asignado un nuevo rol como programador full stack para que escale la solución implementada durante el Reto 3. Se requiere que diseñe un software con patrón arquitectural MVC (Modelo Vista Controlador) el cual se conecte a la base de datos SQLite con la que estuvo trabajando en el Reto 3.

Estructura del proyecto:



[Descargue la estructura del proyecto aquí](#)

El controlador es el encargado de realizar la conexión a la base de datos, crear objetos tipo Proyecto y Lider a partir de los datos obtenidos en la BD, el modelo es la representación de dichos datos y la vista es la encargada de mostrar en consola los datos solicitados en cada requerimiento.

Requerimientos

Mostrar en consola la siguiente información:

1. De la tabla 'Proyecto' obtener la fecha de inicio del proyecto, número de habitaciones y número de baños donde la constructora encargada de ejecutar los proyectos es 'Pegaso'. Mostrar en consola de la siguiente manera:

```
Fecha_Inicio: 2019-09-09 - Numero_Habitaciones: 1 - Numero_Banos: 1
Fecha_Inicio: 2020-12-07 - Numero_Habitaciones: 1 - Numero_Banos: 2
Fecha_Inicio: 2021-04-04 - Numero_Habitaciones: 2 - Numero_Banos: 3
Fecha_Inicio: 2019-09-09 - Numero_Habitaciones: 4 - Numero_Banos: 1
Fecha_Inicio: 2020-06-15 - Numero_Habitaciones: 4 - Numero_Banos: 1
```



2. Obtener fecha inicio de los proyectos, número de habitaciones, número de baños, nombre y primer apellido del líder de cada proyecto y por último el estrato donde se llevará a cabo la ejecución del proyecto que es llevado a cabo por la constructora 'Pegaso'. Limitar el número de registros obtenidos a 50 (deberá interactuar con las tablas: Proyecto, Lider y Tipo). Mostrar en consola de la siguiente manera:

```
Fecha_Inicio: 2019-09-09 - Numero_Habitaciones: 1 - Numero_Banos: 1 - Nombre_Lider: Libia - Apellido_Lider: Pelaez - Estrato_Proyecto: 3
Fecha_Inicio: 2020-12-07 - Numero_Habitaciones: 1 - Numero_Banos: 2 - Nombre_Lider: Paul - Apellido_Lider: Sucerquia - Estrato_Proyecto: 3
Fecha_Inicio: 2021-04-04 - Numero_Habitaciones: 2 - Numero_Banos: 3 - Nombre_Lider: Juana - Apellido_Lider: Perez - Estrato_Proyecto: 3
Fecha_Inicio: 2019-09-09 - Numero_Habitaciones: 4 - Numero_Banos: 1 - Nombre_Lider: Bart - Apellido_Lider: Rivera - Estrato_Proyecto: 3
Fecha_Inicio: 2020-06-15 - Numero_Habitaciones: 4 - Numero_Banos: 1 - Nombre_Lider: Jack - Apellido_Lider: Arteaga - Estrato_Proyecto: 6
Fecha_Inicio: 2020-06-15 - Numero_Habitaciones: 1 - Numero_Banos: 3 - Nombre_Lider: Nadia - Apellido_Lider: Pelaez - Estrato_Proyecto: 5
```

3. Obtener la sumatoria total de habitaciones construidas por cada constructora . Mostrar en consola de la siguiente manera:

```
Total_Habitaciones: 202 - Constructora: Arquitectura S.A.
Total_Habitaciones: 184 - Constructora: Edificios y Edificios
```

4. Obtener el nombre y primer apellido de todos los líderes de la constructora 'Pegaso'. Mostrar en consola de la siguiente manera:

```
Nombre_Lider: Libia - Apellido_Lider: Pelaez
Nombre_Lider: Paul - Apellido_Lider: Sucerquia
Nombre_Lider: Juana - Apellido_Lider: Perez
Nombre_Lider: Bart - Apellido_Lider: Rivera
Nombre_Lider: Jack - Apellido_Lider: Arteaga
Nombre_Lider: Nadia - Apellido_Lider: Pelaez
Nombre_Lider: Juana - Apellido_Lider: Falcon
```

5. Obtener la sumatoria total de habitaciones construidas por cada constructora y mostrar únicamente los registros que tengan una sumatoria total en habitaciones construidas mayor a 200.0, así mismo ordenar por el nombre de la constructora de forma ascendente . Mostrar en consola de la siguiente manera:

```
Total_Habitaciones: 202 - Constructora: Arquitectura S.A.
Total_Habitaciones: 212 - Constructora: Pegaso
```

Notas:

- Recuerde que las consultas a los requerimientos anteriores ya se tienen del reto 3.
- Los ejemplos de las salidas son solo los registros iniciales de las respectivas respuestas que se debe devolver en los requerimientos.
- Recuerde que cada una de las clases debe ser codificada en una clase (archivo independiente), pero se deben de cargar juntas en iMaster y el main no debe ser subido.
- Los packages de la parte superior de cada clase deben de ser eliminados al momento de subir la solución a iMaster