

TEMA: USO DE ROOM EN ANDROID

1. Configurar Grandle. En el archivo Build.Grandle (Module) dentro de la etiqueta de “dependecies”

```
implementation 'android.arch.persistence.room:runtime:1.0.0'  
annotationProcessor 'android.arch.persistence.room:compiler:1.0.0'
```

2. Crear un paquete room en la aplicación y dentro de ella incluir otros paquetes y clases que se necesiten.
3. Crear una clase Constantes para crear constantes que se necesiten:

```
public class Constante {  
    public static final String NOMBRE_DB="dietaDB";  
    public static final String NOMBRE_TABLA_ALIMENTO="alimento";  
    etc  
}
```

4. Crear cada entidad con las anotaciones de Room y los constructores, sets, gets y métodos que se necesiten.

```
@Entity(tableName=Constante.NOMBRE_TABLA_ALIMENTO)  
public class Alimento {  
  
    @PrimaryKey(autoGenerate=true)  
    @ColumnInfo(name="id_alimento")  
    private int id_alimento;  
    @ColumnInfo(name="nombre")  
    private String nombre;  
    ..  
    etc  
    ..  
}
```

5. Crear interfaces llamados dao (ejemplo daoAlimento) con los métodos que se necesiten y anotaciones de room

```
@Dao  
public interface InterfaceDaoAlimento {  
  
    @Insert  
    public void crearAlimento(Alimento al);  
    @Delete  
    public void borrarAlimento(Alimento al);  
    @Update  
    public void modificarAlimento(Alimento al);  
    @Query("SELECT * FROM alimento WHERE id_alimento LIKE :id")  
    public Alimento verAlimento(int id);  
    @Query("SELECT * FROM alimento WHERE nombre LIKE :nombre")  
    public Alimento verAlimento(String nombre);  
    @Query("SELECT * FROM alimento")  
    public List<Alimento> verAlimentos();  
  
}
```

6. Crear una clase abstracta para crear y usar una instancia de la base de datos.

```
@Database(entities={Alimento.class},version=1)
public abstract class AppDB extends RoomDatabase {

    public static AppDB INSTANCE;

    public abstract InterfaceDaoAlimento daoAlimento();

    public static AppDB getAppDB(Context context){

        if(INSTANCE==null){

            //Usar allowMainThreadQueries() solo para pruebas

            INSTANCE=Room.databaseBuilder(context.getApplicationContext(),AppDB.class,
            Constante.NOMBRE_DB).allowMainThreadQueries().build();

        }
        return INSTANCE;

    }

    public void destroyInstance(){
        INSTANCE=null;
    }

}
```

7. Probar la base de datos en el Main

```
Alimento alimento=new Alimento("tomate");

//no funciona si está en el hilo principal a no ser que lo especifiquemos
AppDB.getAppDB(getApplicationContext()).daoAlimento().crearAlimento(alimento);

List<Alimento> alimentos=
AppDB.getAppDB(getApplicationContext()).daoAlimento().verAlimentos();

Log.d("dietapp","hola mundo");

for (Alimento al:alimentos) {

    Log.d("dietapp", al.getNombre());

}
```