droidcon academy

Exploring Room Fundamentals

Belal Khan

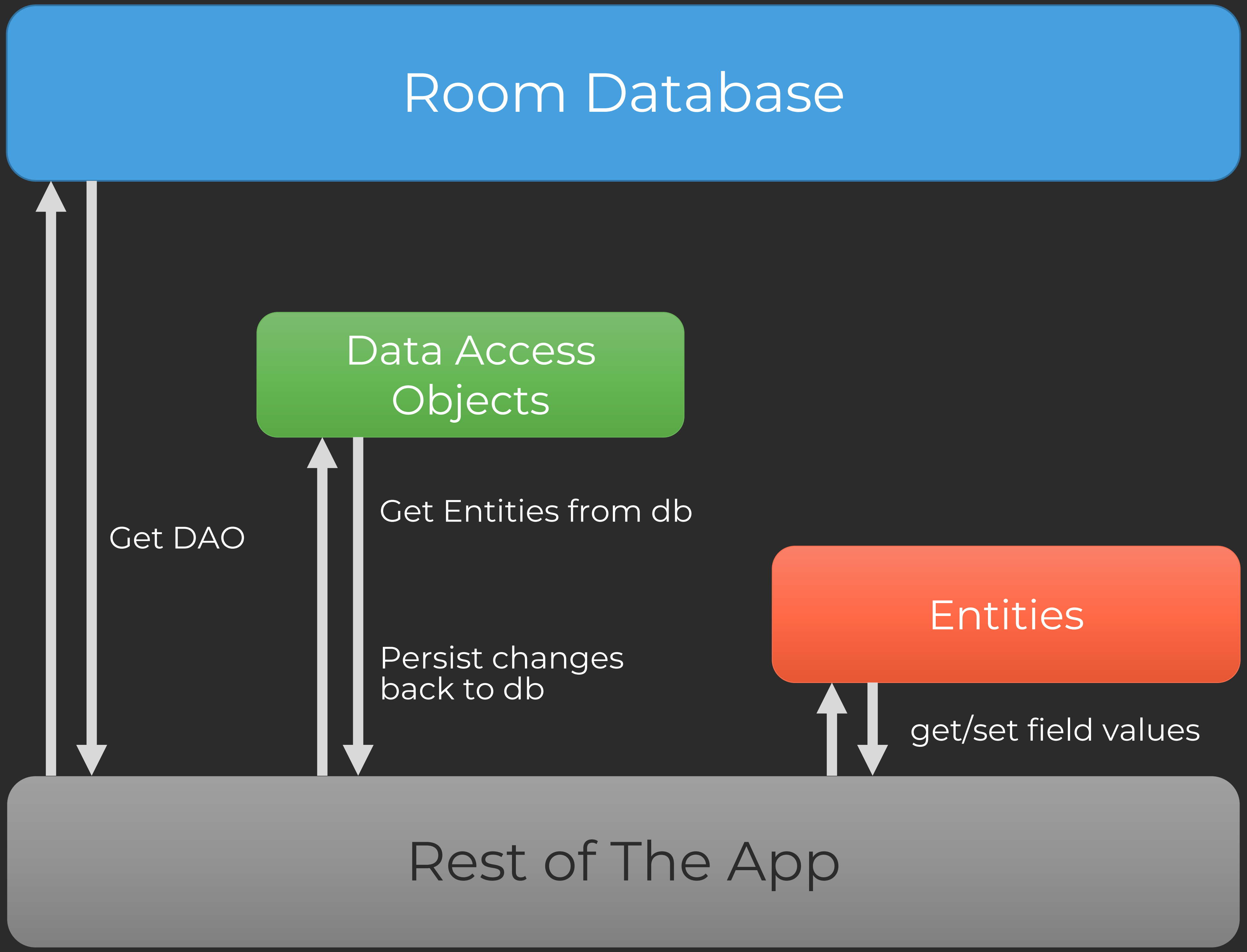
@probelalkhan

in/probelalkhan

Overview

- Room Architecture
- Entities
- Daos
  - Create, Read, Update and Delete Operations
- Database

Room Architecture



Room Database

Data Access Objects

Get DAO

Get Entities from db

Persist changes back to db

Entities

get/set field values

Rest of The App

Entity

- Entities are objects that will be persisted.

```
@Entity
data class User(
    @PrimaryKey val id:
Int,

    val firstName: String?,
    val lastName: String?
)
```

Dao

- Provides abstract access to the database.

```
@Dao
interface UserDao {
    @Insert
    fun insertAll(vararg users: User)

    @Update
    fun update(user: User)

    @Delete
    fun delete(user: User)

    @Query("SELECT * FROM user")
    fun getAll(): List<User>
}
```

Database

- The app’s main access point to persisted data.

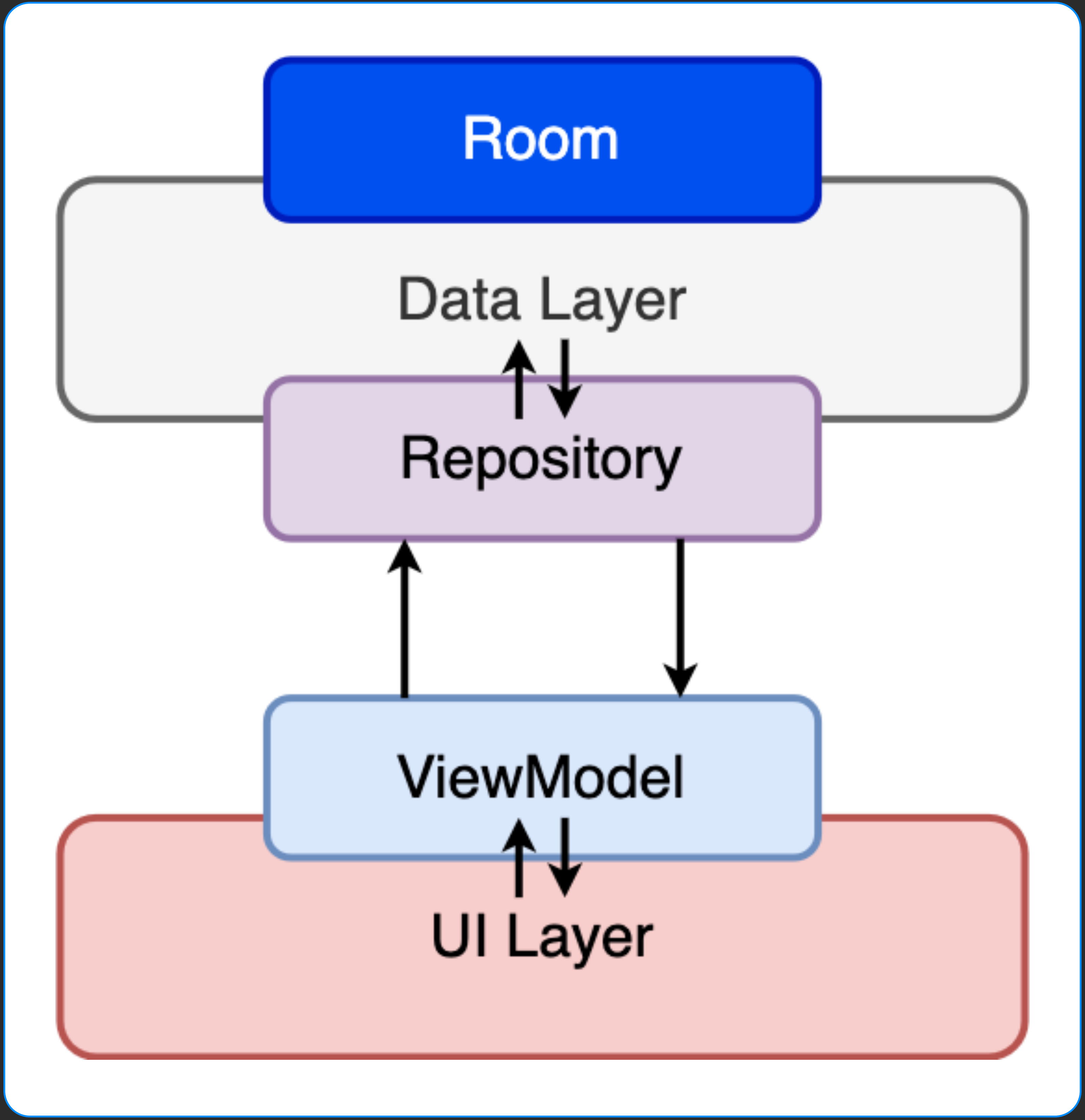
```
@Database(entities = [User::class], version = 1)
abstract class AppDatabase : RoomDatabase() {
    abstract fun userDao(): UserDao
}
```

Database

- Building the database using databaseBuilder.

```
val db =
   Room.databaseBuilder(
       applicationContext,
       AppDatabase::class.java, "database-name"
   ).build()
```

Easy Invoice Architecture



Summary

- Room Database Architecture
- Creating Database Tables with Entities
- Performing Create, Read, Update and Delete Operations with DAOs
- Creating Database Instance
- Easy Invoice Architecture

Setting Up Room Database

Up Next