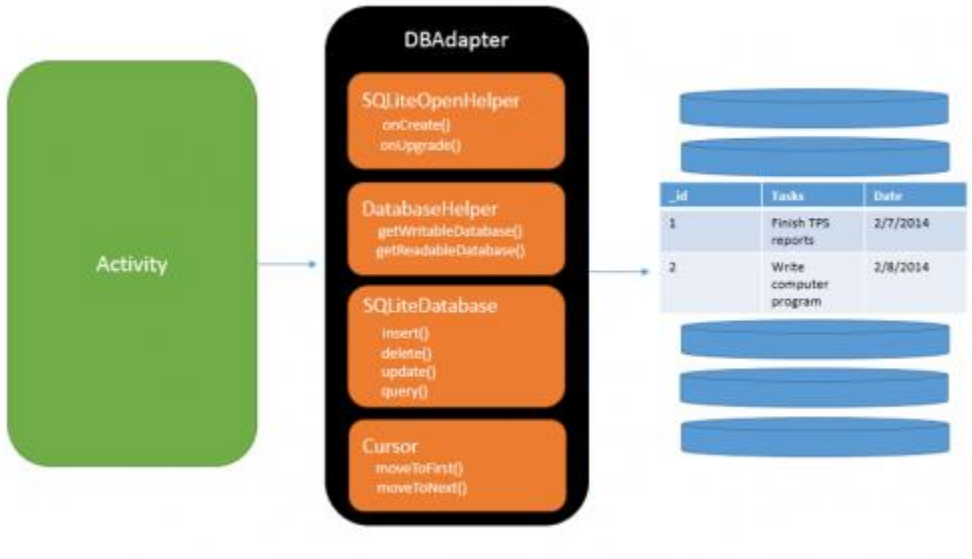
## Design Pattern

### Database Access Object (DAO)




- The Data Access Object (DAO) provides an abstract interface between the database and the logic managers. The CRUD (Create, Retrieve, Update, Delete) operations are performed by DAO classes on the database. In MediPoint all the major functionalities have DAO objects which perform necessary CRUD operations for the manager classes instead of these classes directly changing the database. The advantages of using DAO pattern are:
  - It creates a necessary seperation between the two important parts of the application i.e. Database and Logic Managers.
  - It is easy to modify the logic for managers in future without affecting the database schema.
  - It is easy to modify the database schema or add more relations to the database without affecting the overall logic. For new additions, logic can be easily modified because there is a clear seperation.

### Layered Architecture

- In MediPoint a 3-tier layered architecture has been implemented. The three layers are:
  - Presentation Layer : The user interface (UI) which displays data to the user and accepts input from the user.
  - Business Logic : handles data validation, business rules and task-specific behaviour.

    - Data Access logic : communicates with the database by constructing SQL queries and executing them via the relevant API.

- The benefits of layered architecture is Scalability since each tier can scale horizontally. For example, you can simply make a new presentation layer for making a web application since the current presentation layer for MediPoint is conigured for mobile devices.