

The Realm Database

Realm is a mobile database that runs directly inside phones, tablets or wearables.

- `npm i realm`

Features

- **Mobile-first:** Realm is the first database built from the ground up to run directly inside phones, tablets and wearables.
- **Simple:** Data is directly [exposed as objects](#) and [queryable by code](#).
- **Fast:** It is faster than even raw SQLite on common operations, while maintaining an extremely rich feature set.

Realm Studio is primarily a graphical user interface (GUI) tool for inspecting and interacting with Realm databases. It allows you to explore, edit, and query your Realm database (filter, sort, etc.), but it is not used to fetch or retrieve data in real-time from an Android device unless the app has already created and populated the Realm file.

Storage and Update Speed

- **Storage Efficiency:** Realm provides efficient storage by using a binary format that is highly optimized for mobile environments. The database is embedded directly in the app, reducing the need for external connections and thus improving the app's performance. Realm's compact file size also helps save storage space on mobile devices.
- **Update Speed:** One of Realm's standout features is its fast read/write speeds, especially for common database operations like inserting, updating, or deleting data. Due to its native integration with mobile environments and the absence of an intermediary layer like a network call to a remote database, updates to the database occur almost instantaneously. This speed makes Realm ideal for applications that need to respond quickly to user input or offline data changes.

User-Friendliness

Realm is highly user-friendly for developers. It eliminates the need to write complex SQL queries, as data is modeled as native objects. Instead of managing tables and rows, developers can work directly with objects, improving productivity and reducing boilerplate code.

Advantages of Realm

1. **Performance:** Realm is much faster than traditional SQLite databases for most operations, especially for inserting, updating, and querying data.
 2. **Offline Support:** Since the database is embedded within the app, users can continue using the app without an internet connection. Realm syncs with the backend when the connection is restored.
 3. **Easy Integration:** Realm is simple to set up and integrate into both Android and iOS apps. It provides native APIs, which makes it easier for developers to work with the database.
 4. **Real-Time Sync:** Through MongoDB Atlas Device Sync, Realm allows data to be synchronized across devices and platforms in real-time, making it ideal for apps that require consistent data across multiple devices.
 5. **Schema-less Database:** Realm does not require strict schema definitions and allows for flexibility in how data is stored and queried.
-

Disadvantages of Realm

1. **File Size Limitations:** While Realm's binary storage format is efficient, it may not be ideal for extremely large datasets, especially when the database grows in size over time.
2. **Learning Curve for Complex Queries:** Although Realm provides a simple interface for basic operations, more complex queries (like advanced filtering and joins) can be challenging to implement compared to SQL-based databases.
3. **Limited Querying Flexibility:** Unlike SQL databases, Realm's querying capabilities may not be as flexible, especially for apps that require complex, multi-table joins or cross-table searches.
4. **Platform Lock-in:** Realm is primarily designed for mobile apps, and while it does offer some backend integration (via MongoDB Atlas), it's not as widely supported in other platforms (such as server-side or web applications) as traditional databases like PostgreSQL or MySQL.
5. **Not a Full Replacement for Traditional Databases:** While Realm is excellent for mobile apps, it may not always be suitable for apps with extensive relational database needs or complex server-side logic.

Realm POC Application:

- 1). Query operations - sorting and search (filter)
- 2). Basic operations (CRUD) - Add, Fetch, Update, Delete

Stored nearly 500 JSON records in this application. All operations, including fetch, add, update, delete, search, and sort, are functioning properly. The data is accessible in Realm Studio after pulling the .realm file to a local machine. Otherwise, the data remains stored on the mobile device, and all operations are updated instantly within the app.

For 500 records, the .realm file size is approximately 5-7 MB, and it is pulled to the local machine in about 0.007 seconds. The data is stored in binary format.

```
C:\Users\Asus>adb pull /storage/emulated/0/Android/data/com.realm/files/myrealm.realm
/storage/emulated/0/Android/data/com.realm/files/myrealm.r...1 file pulled, 0 skipped. 7.1 MB/s (49152 bytes in 0.007s)
```