**SQLite vs Firebase**

**SQLite** este o bază de date relațională open-source, folosită pentru a efectua operațiuni de baze de date pe dispozitive Android, cum ar fi stocarea, manipularea sau preluarea datelor persistente din baza de date. SQLite este o bază de date locală pe dispozitivul Android (datele stocate / procesate pe un dispozitiv) cu interfață SQL. Deci, nu este nevoie să efectuați nicio sarcină de configurare sau administrare a bazei de date.

**Avantaje :**

* Excelent pentru dezvoltare și testare.
* Încarcă doar datele necesare, în loc să citească întregul fișier și să le țină în memorie.
* Nu este nevoie de instalare si configurare.
* Actualizează conținutul în mod continuu, astfel încât se pierde puțin în caz de avarie.
* Este portabil pe toate sistemele de operare pe 32 de biți și pe 64 de biți.
* Poate fi utilizat cu toate limbajele de programare fără nici o problemă de compatibilitate.

**Dezavantaje** :

* Este utilizat pentru a gestiona solicitările HTTP de trafic mic sau mediu; nu este construit pentru aplicații la scară largă.
* Dimensiunea bazei de date este restricționată la 2 GB în majoritatea cazurilor.
* Nu oferă acces la rețea deoarece este fără server.

In timp ce, **FireBase** este un serviciu cloud, fiind o bază de date NoSQL. Date stocate / procesate într-un cloud, avand nevoie de o conexiune la internet pentru a le accesa. FireBase este potrivit pentru aplicații în timp real.

**Avantaje :**

* Stocajul în cloud este atât de ușor disponibil peste tot.
* Dacă aplicația este actualizată de o mulțime de utilizatori, serviciul FireBase este capabil să gestioneze actualizările de date în timp real între dispozitive.
* Nu trebuie să ne ingrijoram pentru hardware!

**Dezavantaje:**

* Formatul de stocare este complet diferit de cel al SQL (Firebase folosește JSON), astfel încât să nu poți migra cu ușurință.
* Instrumentele de raportare nu vor fi nicăieri în apropierea celor standard SQL.
* Limitat la 100 de conexiuni și 1 GB de stocare!
* In funcție de serverul folosit, în timp vizionarilor, se pare că există multe perturbări în ultima perioadă.

**SQL vs NoSQL**

Bazele de date **SQL** utilizează un limbaj de interogare structurat (SQL) pentru definirea și manipularea datelor. Pe de o parte, aceasta este extrem de puternică: SQL este una dintre cele mai versatile și utilizate pe scară largă opțiuni disponibile, ceea ce face o alegere sigură și mai ales excelent pentru interogări complexe.

**Avantaje**:

* Management rapid.
* Integrare cu diferite instrumente de dezvoltare, de exemplu cu IntelliJ Idea.
* Este disponibil gratuit.
* Există o varietate de interfețe de utilizator care pot fi implementate.
* Poate fi lucrat cu alte baze de date, inclusiv DB2 și Oracle.

**Dezavantaje :**

* Poate fi restrictiva.
* Necesita utilizarea unor scheme predefinite pentru a determina structura datelor înainte de a lucra cu acestea.
* Toate datele trebuie să urmeze aceeași structură.
* Nu există suport pentru XML.

In timp ce, bazele de date **NoSQL** câștigă in prezent popularitate ca fiind cel mai bun model alternativ de gestionare a bazelor de date.

**Avantaje** :

* Scalabilitate flexibila.
* Stocheaza cantitati masive de date.
* Modelele NoSQL au nevoie de o gestionare mult mai mică a experților, deoarece au deja capacități de reparare auto și distribuție a datelor, mai puține cerințe de administrare și transformare, precum și proiectări simplificate de date.
* Este ușor de instalat.

**Dezavantaje** :

* Modelele NoSQL au fost create datorită aplicațiilor web moderne 2.0 2.0. Și din această cauză, majoritatea funcțiilor NoSQL sunt concentrate să răspundă acestor cerințe, ignorând cerințele aplicațiilor făcute fără aceste caracteristici, prin urmare, ajung să ofere mai puține funcții analitice pentru aplicațiile web normale.
* Fiecare firmă ar trebui să fie sigură că, în cazul în care o funcție cheie din sistemul lor de baze de date nu reușește, acestea vor avea asistență competentă nelimitată oricând. Toți furnizorii de modele raționale au depășit kilometrul suplimentar pentru a oferi această asigurare și s-au asigurat că asistența lor este disponibilă 24 de ore, ceea ce nu este încă un pas garantat de furnizorii NoSQL.
* NoSQL sunt foarte noi în comparație cu bazele de date relaționale, ceea ce înseamnă că sunt mult mai puțin stabile și pot avea mult mai puține funcționalități.

Informatii Ex3:

* Link tutorial: <https://abhiandroid.com/database/sqlite>
* Link Git: <https://github.com/cristianapetreaca/DAPM_lab/tree/master/Lab8>