# **Spørsmål**

**Forklar i grove trekk hvordan man kobler seg til SQLite-databasen til en Android-app, og hvilke hensyn man bør ta.**

For å koble seg til en SQLite-database i en android-app så er normen å ha en adapter klasse som fungerer som et lag mellom bruker og database og som håndterer SQL logikk slik som metoder for innsetting, uthenting, oppdatering og sletting, samt åpning og stenging av database. Videre kan du ha en abstrakt hjelpe-klasse som benytter **SQLiteOpenHelper** som er bygd rundt best praksis for håndtering av database. Vi valgte dog å slå disse i sammen til en klasse i vår løsning. Ting å merke seg er at databasen kun er tilgjengelig for applikasjonen som opprettet den og at SQL-92 standarden ikke er støttet og det er derfor ikke foreign key contstraints, nøstede transaksjoner, right outer join og full outer join. Android har begrenset ressurser og det er derfor også veldig viktig å normalisere databasen.

**Hva er en AsyncTask, og hvorfor og når er denne nyttig? Hva gjør de vanligste metodene i AsyncTask?**

**AsyncTask** gjør at du kan utføre oppgaver i bakgrunnen slik at du ikke bruker tid på UI-tråden. Du trenger dessuten ikke manipulere tråder eller håndteringsklasser. **AsyncTask** er en hjelpeklasse og bør ideelt benyttes til korte operasjoner. Dersom du ønsker at tråden skal kjøre over lengre tid bør du benytte et annet API. **AsyncTask** må være en subklasse for å bli brukt og **doInBackground()** må overskrives. De vanligste metodene er **onPreExecute()** som benyttes for å gjøre eventuelle oppsett i forbindelse med koden som skal eksekveres, f eks instansiere variabler. **doInBackground()** som jeg har nevnt kjøres rett etter **onPreExecute()** og denne brukes til å kjøre koden som skal kjøre lenge. Parameterne fra **AsyncTask** blir sendt her og resultatet av koden må returneres. Videre har vi **onProgressUpdate()** som benyttes for å vise fremskritt for brukeren i UI mens **doInBackground()** kjører. Til slutt har vi **onPostExecute()** som kjøres etter **doInBackground()** er ferdig og reslutatet av koden blir sendt hit. Vi har også metoder som **onCancelled()** og **publishProgress()**, men disse er mindre vanlige.