

Obligatorisk innlevering 1 – SQL

Grupper opptil 4 personer. Dere legger dere selv til i en av de opprettede gruppene. Innlevering som PDF-dokument. Eventuell kode skal legges inn som tekst (og ikke screenshot) slik at den kan testes. Angi også hvilken database (bruker/studentnummer) hvor innleveringen er lagret.

I innleveringene vil det også vil bli lagt vekt på ”akademisk skriving”, dvs., hvordan man skriver besvarelser og avhandlinger på en strukturert og ”god” måte.

IMRD- oppsettet (Intro, Method, Research, Discussion, har vært introdusert i andre kurs tidligere) i den grad det passer for oppgavene. Dette gjelder særlig oppgaver som kan løses på mer enn en måte, og de ulike måtene å løse på kan ha visse fordeler og ulemper som må veies opp mot hverandre før man kan velge hvordan man vil løse oppgaven. IMRD-strukturen bør brukes slik:

1. Introduksjon
 - Beskrive problemstillingen (»kravspesifikasjonen») i oppgaven på en oversiktlig måte.
 - Beskrive (del)oppgavene som skal løses. Beskriv begge deloppgavene under ett.
 - Pass på å bruke godt språk, og logisk struktur som er lett lesbart og pent oppsatt.
2. Metode
 - Beskriv metoden(e) som brukes for å løse (del)oppgavene
 - For begge oppgavene kan dere med fordel strukturere svarene i en matrise, for systematisk sammenligning, f.eks. slik:

Eksempel:

	Fordel	Ulempe
Mulighet 1
Mulighet 2
...		

Dette kan også struktureres som mange delseksjoner, en for hver rad i tabellen, om det er bedre.

3. Resultat
 - Beskriv resultatet (»svaret») på deloppgavene etter å ha benyttet metoden fra 2.
 - Dette blir to ulike svar, så del gjerne inn i flere seksjoner (delkapitler) slik at hvert tema omtales i sin egen seksjon.
4. Diskusjon
 - Oppsummer resultatene, funnene og vurderingene dere har gjort.

Problembeskrivelse

Det er behov for et system som registrerer passeringer av biler for bompenger. For hver bil skal det lagres registreringsnummer, eier, eiers adresse, epost og telefonnummer. For hver passering skal det registreres hvilken bil som passerte, tidspunkt, og hvilken “bompenggebod”. Kommenter valgene dere tar for å klargjøre hvordan dere tenker/resonerer.

Oppgave

- a) Skriv SQL-kode for å opprette tabellene. Dere må selv vurdere hvilke kolonner som trengs, og hvilke datatyper som passer.
- b) Skrive SQL-kode for å legge inn test-data slik at de påfølgende deloppgavene gir mening.
- c) Det er ikke lenger behov for å lagre telefonnummer. Skrive SQL-kode for å fjerne dette.
- d) Hvordan håndterer løsningen deres tilfeller hvor det er en passering, men skiltet kunne ikke bli lest? Er det andre måter å håndtere dette på?
- e) Skriv en SQL-spørring som lister ut all informasjon, inkludert eier og epost for alle passeringer. Også passeringer som ikke har registrert registreringsnummer skal bli tatt med. (Dvs. skiltet kunne ikke bli lest.)
- f) Skriv en SQL-spørring som lister ut all informasjon, inkludert eier og epost for alle passeringer. Bare passeringer som har registrert registreringsnummer skal bli tatt med. (Dvs. skiltet kunne bli lest og vi kan sende faktura.)
- g) Skriv samme spørring som i deloppgave f) som relasjonsalgebra.
- h) Skriv en SQL-spørring som viser antall passeringer som er registrert på ulike registreringsnummer.
- i) Skriv en SQL-spørring som finner siste passering for bil med registreringsnummer AA10000.
- j) Skriv en SQL-spørring som viser antall passeringer uten registrert registreringsnummer (dvs., antall passeringer vi ikke får betalt for.)
- k) Kan løsningen deres håndtere biler med utenlandske registreringsnummer? Hvorfor/hvorfor ikke. Er det noen mulige problemstillinger knyttet til dette?
- l) Denne oppgaven er tenkt løst med to tabeller: Bil og Passering. Er det andre tabeller dere tror kunne vært nyttig?

Huskliste/krav for hva besvarelsen skal (minst) inneholde

- ☒ Oppgaven kan bli besvart helhetlig (istedenfor punktvis) i den grad dette er enklere. Men alle punktene skal besvares.
 - Primærnøkler (som du mener er mest hensiktsmessig)
 - Fremmednøkler
 - Datatyper
 - Null/not null
 - Eventuelle fremmednøkler og on update/delete krav.
 - **Redegjørelse for de valgene som er tatt. Det blir lagt vekt på både praktisk utførelse og teoretisk forståelse**
 - **Besvarelsen skal være ”konsistent”, dvs. skal være for den ”samme” databasen, og de valgene man har tatt i en del av besvarelsen skal ikke ”motsies” av de valgene man har tatt i en annen del av besvarelsen.**