

## IS-203

### Oblig innlevering: Del 2

**Frist: 28.2.2011**

I denne delen av oppgaven skal dere bygge videre på modellen som ble lagd i del 1. Om nødvendig kan dere remodellere databasen.

## 1. PL/SQL prosedyrer , funksjoner

I filen skiklubben2.ddl finner du spesifikasjon til flere prosedyrer som har med skiklubben å gjøre. Oppgaven er å opprette selve prosedyrer. En kort beskrivelse av inkluderte prosedyrer:

**LEGG\_TIL\_NY\_MEDLEM\_sp.** Legg ny medlem i databasen. Må opprette og legge inn ny identifikator for medlemmen. identifikatoren blir sendt tilbake fra prosedyren. Prosedyren kan bli tilsendt navn til skole, barnehage, og idrettslag.

**LEGG\_TIL\_NY\_SESONG\_pp.** Legg ny sesong i databasen. Sesongen skal være en sammensetning av to år, feks, 2009-2010, 2010-2011, osv.

**LEGG\_TIL\_NY\_ORGANISASJON\_pp.** Legg ny organisasjon i databasen. Må opprette og legge in ny identifikator for organisasjon.

**LEGG\_TIL\_NY\_GRUPPE\_pp.** Legg ny gruppe i databasen. Må spesifisere sesongen, og administratoren sin brukernavn. Må bruke brukernavn til å slå opp primærnøgelen til tilsvarene medlem å legge inn denne verdien inn som fremmednøkkel.

**MEDLEM\_SESONG\_ALDER\_sp.** Gitt en sesong og en medlem brukernavn, returner alderen som skal brukes for å bestemme hvilken skimerke medlem skal få. Beregn alderen etter foretningsregler på PPT side (G).

**REGISTRER\_AKTIVITET\_pp.** Gitt dato, medlem brukernavn, gren type, anlegg, beskrivelse, distanse, konkurranse, legg inn ny langrenn aktivitet (ikke hopp eller alpint ; det kommer senere). Prosedyren skall opprette og legge inn aktivitet identifikator (primærnøkkelverdi).

Aktiviteten kan ikke registreres på forhånd. Datoen må være <= SYSDATE

Distanse må være >= 0, og kan ikke være null.

**BEREGN\_DISTANSE\_sp.** Gitt et brukernavn og en sesong, beregn antall kilometer medlem har registret i den sesongen. Beregningen skal baseres på registrert aktivitetdistanse. Prosedyren returnerer antall kilometer til prosedyren som kalte den.

**GI\_UT\_MERKE\_sp.** Gitt et brukernavn og en sesong, beregn hva slags merke medlem skal få. Prosdyren må beregne antall kilometer, og sjekke antall mot den som kreves av medlemmer på medlem sin alder. Returner kun den beste merke fortjent, dvs, returner 'sølv', ikke både 'sølv' og 'bronse'.

**GI\_UT\_ALLE\_MERKER\_sp.** Gitt en sesong, gi til alle som har fortjent det merkene de har fortjent i løpet av sesongen. Lagre i databasen medlem, sesong, og fortjente merke.

Oppgaven er å fullføre prosedyrene ved å legg til PL/SQL kode som skal utføre ønsket funksjonaliteten.

Det er viktig å

1. ikke endre grensesnitt til prosedyrene. Den eneste måten å endre grensesnitt på er å forhandle om det med alle andre some satser på grensesnittet (andre studenter og meg). Poenget er å forstå at hver enkelt programmist har ikke rett til å bestemme for seg selv å entre det som andre satser på.
2. ikke glemme om feil behandling. Prosedyren må utføres uten feil, selv om data som blir sendt gjemmon parameterene passer ikke i databasen, eller data i databasen mangles, osv. Hvis det oppstår feil, meldingene burde være logiske og hjelpsomme. Jeg venter EXCEPTION deler i hver prosedyre.

## 2. Spøringer

Lag følgende sql-setninger for å hente ut data:

1. Lag en spørring som viser antall kilometer for hver gren for hver kommune i en gitt sesong. Resultatet av spørringen skal være en tabell med folgene kolonner: kommune, gren, antall kilometer. Sorter resultatet fra flest til færrest antall kilometer.
2. Lag en spørring som viser antall kilometer for hver medlem som er med på en gitt skole. Resultatet av spørringen skal være en tabell med folgene kolonner: skole, medlem fornavn, medlem etternavn, antall kilometer. Sorter resultatet fra flest til færrest antall kilometer.

## Generelt

Innleveringen skal bestå av:

- 1) Lag følgende SQL-script som vedlegg til besvarelsen:
  - SQL-script for opprettelse av database-tabeller og constraints etter eventuelle endringer som er gjort i datamodellen.
  - SQL-script for opprettelse av prosedyrer eventuelt andre Oracle objekter dere har etablert som del av løsningen.
  - SQL-script med spøringer for oppgave 2.
- 2) Et MS Word dokument levert i Fronter med flg. innhold
  - Redegjør for eventuelle forutsetninger dere har gjort.
  - Lag et revidert ER-diagram dersom dere har endret datamodellen.
- 3) Alle objekter skal være opprettet på gruppens Oracle-bruker på kursets database server (IS203/dbadb på oracle.ist.unomaha.edu)

- 4) Hver medlem i gruppen skal også levere refleksjon. Svar på to spørsmål, blant de som finnes på side <http://escalate.ac.uk/resources/reflection/09.html>