

Oblig innlevering: Del 3

Teller 40% av semesteroppgaven totalt

Frist: 21.03.2011

I denne delen av oppgaven skal dere bygge videre på modellen som ble lagd i del 1, og lagrede prosedyrer som ble opprettet i del 2.

1. PL/SQL pakker, prosedyrer , funksjoner

I filen skiklubben3.sql er SQL DDL til å opprette en PL/SQL pakke som heter SKIKLUBBEN_PKG, som inkludere spesifikasjon til flere prosedyrer og funksjoner som hører til den nedre pakken. Oppgaven er å opprette selve prosedyrer og funksjoner i en 'package body'. Flere av disse har dere allerede opprettet i del 2, men med en annen prosedyrenavn (f.eks LEGG_TIL_NY_MEDLEM_SP i del 2; LEGG_TIL_NY_MEDLEM_PP i del 3. _PP er en 'pakkeprosedyre'. En kort beskrivelse av inkluderte prosedyrer og funksjoner:

1. **LEGG_TIL_NY_MEDLEM_pp**. Legg ny medlem i databasen. Må opprette og legge inn ny identifikator for medlemmen.
2. **LEGG_TIL_NY_SESONG_pp**. Legg ny sesong i databasen. Sesongen skal være en sammensetning av to år, feks, 2009-2010, 2010-2011, osv.
3. **LEGG_TIL_NY_ORGANISASJON_pp**. Legg ny organisasjon i databasen. Må opprette og legge inn ny identifikator for organisasjon.
4. **LEGG_TIL_NY_GRUPPE_pp**. Legg ny gruppe i databasen. Må spesifisere sesongen, og administratoren sin brukernavn. Må bruke brukernavn til å slå opp primærnøkkelen til tilsvarende medlem å legge inn denne verdien inn som fremmednøkkel.
5. **MEDLEM_SESONG_ALDER_pf**. 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).
6. **REGISTRER_AKTIVITET_pp**. Gitt dato, medlem brukernavn, gren type, anlegg, beskrivelse, sesong, distanse, konkurranse, legg inn ny langrenn aktivitet.
7. **REGISTRER_AKTIVITET_pp**. Samme navn som forrige, men andre parametere. Gitt dato, medlem brukernavn, gren type, anlegg, beskrivelse, sesong, antall hopp, konkurranse, k-punkt, og lengste hopp, legg inn ny hopp aktivitet.
8. **REGISTRER_AKTIVITET_pp**. Samme navn som forrige, men andre parametere. Gitt dato, medlem brukernavn, gren type, anlegg, beskrivelse, sesong, timer, minutter, og konkurranse, legg inn ny alpint aktivitet. Må beregne antal kilometer som tilsvarer tiden (1 time regnes som 4 kilometer).
9. **BLI_MED_I_GRUPPE_pp**. Gitt et brukernavn, gruppenavn, sesong gjør medlem til å bli medlem i gruppen.
10. **BEREGN_DISTANSE_pf**. Gitt et brukernavn og en sesong, beregn antall kilometer medlem har registrert i den sesongen.
11. **GI_UT_MERKE_pp**. Gitt et brukernavn og en sesong, beregn hva slags merke medlem skal få, og lagre dette i databasen.
12. **GI_UT_ALLE_MERKER_pp**. Gitt en sesong, gi til alle merkene de har fortjent i løpet av sesongen. Lagre i databasen medlem, sesong, og fortjente merke.
13. **POKAL_FORTJENERE_pp**. Gitt en sesong, finn alle medlemmer som har fortjent 3-års pokal, dvs, har fått gull siste tre år på rad.

14. **REGISTRER_GRUPPE_AKTIVITET_pp.** Gitt dato, gruppenavn, administrator brukernavn, gren type, anlegg, beskrivelse, sesong, distanse, konkurranse, legg inn ny langrenn aktivitet til alle medlemmer av gruppen.

Forklaring av parameterene finnes i skiklubben3.sql filen.

2. Database utløser (trigger)

Opprett utløser som oppdaterer antall kilometer en medlem har registrert, hver gang en ny aktivitet registreres. Utløseren kan bruke funksjonen `BEREGN_DISTANSE_PF` og lagre resultatet i hensiktsmessig sted i databasen.

3. Spørringer

Lag følgende sql-setninger for å hente ut data:

1. Lag en spørring som viser total kilometer for hver kommune innen en gitt fylke mellom to datoer (f.eks. mellom 1-feb-2005 og 15-feb-2005) i hver grentype. La utlistingen inneholde følgende kolonner: kommune, grentype, total kilometer.
2. Lag en spørring som viser total kilometer for en gitt gruppe. Total består av alle medlem sine kilometer. La utlistingen inneholde følgende kolonner: gruppe, sesong, total.
3. Lag en spørring som viser total kilometer i en gitt sesong for alle organisasjoner en gitt medlem har forhold til. Total består av alle kilometer av alle medlemmer som har forhold til de samme organisasjoner. La utlistingen inneholde følgende kolonner: organisasjon, sesong, total.

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 siden forrige innlevering.
 - SQL-script for opprettelse av pakker, prosedyrer, utløser og eventuelt andre Oracle objekter dere har etablert som del av løsningen.
 - SQL-script med spørringer for oppgave 3.
- 2) Et dokument levert i Fronter med flg. innhold
 - Redegjør for eventuelle forutsetninger dere har gjort.
- 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>