# Konteeksamen m LØSN i 2DAN101-1 21V Databaser og nettverk

Høyskolen i Innlandet, Spillskolen

Dato: 17.3.2021

Tid: 09:00 til 13:00

Tillatte hjelpemidler:

* Lærebok og andre skriftlige dokumenter
* Egen datamaskin.

Det forutsettes at du har følgende programmer (eller tilsvarende) installert på PCen din: WAMP (Apache, MySQL eller MariaDB), MySQL Workbench, samt en php-editor (f.eks. Notepad++ eller VSCode, men Notepad kan også brukes). Også MongoDB må være installert. Du kan bruke lokal MongoDB med kommandovindu, eller med Compass.

I Inspera må det leveres én zip-fil med navn **xxx.zip** der **xxx** er kandidatnummer.

Det er ikke lov til å bruke internett, unntatt for å gå inn på Inspera. Praktisk utførelse av dette:

* Du kobler deg opp mot Inspera for å hente ut oppgavetekst.
* Så slår du av internett på PCen (SLÅ PÅ FLYMODUS)
* Når oppgaven skal leveres, slår du på internett for å bruke Inspera

Et par ting du må passe på:

* Når du lager en tabell, sett alltid **Engine=InnoDB**.
* Når Workbench lager en index, er det i scriptet med ordet VISIBLE etter hver index. MariaDB og MySQL liker ikke den, så den må slettes (kommer an på versjon av databasene)

**OPPGAVEN**

En database for flervalgstester, beregnet på studenter, har datafelter som vist i tabellen under. Vi vil sjekke ut hvordan disse dataene bør lagres, både i MongoDB og i en SQL-database.

Tabell 1 nedenfor viser de datafeltene som MÅ være med.

Dataene kan selvfølgelig lagres i en SQL-database som Tabell 1. Men i oppgave 4 skal du lage en bedre datastruktur, ved å dele Tabell 1 opp i flere tabeller. Studenter (de som tar testene) skilles ut i en egen tabell, og testene skilles ut i en egen tabell (eller flere). Føl fri til å innføre så mange tabeller du føler er nødvendig. Du kan også legge til felter, hvis det er nødvendig.

Selve spørsmålene ligger ikke i databasen, i ei fil angitt i feltet **SpmFil**. Hver test består av **AntallSpm,** og spørsmål nummerert under **SpmNr**, med 4 svaralternativer, hvorav bare ett er riktig. Studenten velger ett av alternativene og svaret lagres under **AvgittSvar**. Det riktige svaralternativet er lagret under **KorrektSvar**.

Noen data fra en test er gitt i tabellen under.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **StudentID** | **StudentNavn** | **TestID** | **SpmFil** | **AntallSpm** | **SpmNr** | **AvgittSvar** | **KorrektSvar** |
| 1101 | Birger Bang | T1 | Intro.sql | 5 | 1 | 2 | 2 |
| 1101 | Birger Bang | T1 | Intro.sql | 5 | 2 | 3 | 4 |
| 1101 | Birger Bang | T1 | Intro.sql | 5 | 3 | 4 | 4 |
| 1101 | Birger Bang | T1 | Intro.sql | 5 | 4 | 2 | 2 |
| 1101 | Birger Bang | T1 | Intro.sql | 5 | 5 | 2 | 2 |
| 1102 | Sigrid Sund | T4 | Intro.sql | 8 | 1 | 1 | 2 |
| 1102 | Sigrid Sund | T4 | Intro.sql | 8 | 2 | 4 | 4 |
| 1102 | Sigrid Sund | T4 | Intro.sql | 8 | 3 | 4 | 4 |
| 1102 | Sigrid Sund | T4 | Intro.sql | 8 | 4 | 3 | 2 |
| 1102 | Sigrid Sund | T4 | Intro.sql | 8 | 5 | 3 | 3 |
| 1102 | Sigrid Sund | T4 | Intro.sql | 8 | 6 | 2 | 2 |
| 1102 | Sigrid Sund | T4 | Intro.sql | 8 | 7 | 1 | 1 |
| 1102 | Sigrid Sund | T4 | Intro.sql | 8 | 8 | 1 | 2 |

**Tabell 1**

1. Åpne et **Mongo**-vindu og opprett en database **Flervalg** med en dokumentsamling **Test**.

Lag et **JSON**-objekt med dataene gitt i **Tabell 1**, og sett dataene inn i den nye MongoDB databasen.

LØSN

**>use Flervalg;**

**>db.createCollection("Test");**

**db.Flervalg.insert( [**

**{"StudentID":1101,"StudentNavn":"Birger Bang","TestID":1,"SpmFil":"Intro.sql", "AntallSpm":5, "SpmNr":1, "AvgittSvar":2, "KorrektSvar":2},**

**{"StudentID":1101,"StudentNavn":"Birger Bang","TestID":1,"SpmFil":"Intro.sql","AntallSpm":5, "SpmNr":2, "AvgittSvar":3, "KorrektSvar":4},**

**{"StudentID":1101,"StudentNavn":"Birger Bang","TestID":1,"SpmFil":"Intro.sql","AntallSpm":5, "SpmNr":3, "AvgittSvar":4, "KorrektSvar":4},**

**{"StudentID":1101,"StudentNavn":"Birger Bang","TestID":1,"SpmFil":"Intro.sql","AntallSpm":5, "SpmNr":4, "AvgittSvar":2, "KorrektSvar":2},**

**{"StudentID":1101,"StudentNavn":"Birger Bang","TestID":1,"SpmFil":"Intro.sql","AntallSpm":5, "SpmNr":5, "AvgittSvar":2, "KorrektSvar":2},**

**{"StudentID":1101,"StudentNavn":"Sigrid Sund","TestID":1,"SpmFil":"Intro.sql","AntallSpm":5, "SpmNr":1, "AvgittSvar":2, "KorrektSvar":2},**

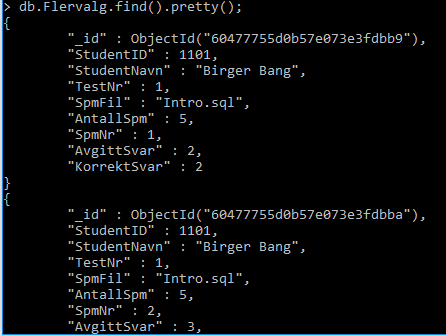
**{"StudentID":1101,"StudentNavn":"Sigrid Sund","TestID":1,"SpmFil":"Intro.sql","AntallSpm":5, "SpmNr":2, "AvgittSvar":4, "KorrektSvar":4},**

**{"StudentID":1101,"StudentNavn":"Sigrid Sund","TestID":1,"SpmFil":"Intro.sql","AntallSpm":5, "SpmNr":3, "AvgittSvar":4, "KorrektSvar":4},**

**{"StudentID":1101,"StudentNavn":"Sigrid Sund","TestID":1,"SpmFil":"Intro.sql","AntallSpm":5, "SpmNr":4, "AvgittSvar":3, "KorrektSvar":2},**

**{"StudentID":1101,"StudentNavn":"Sigrid Sund","TestID":1,"SpmFil":"Intro.sql","AntallSpm":5, "SpmNr":5, "AvgittSvar":1, "KorrektSvar":2}**

**]);  
db.Flervalg.insert([...]);** der **[...]** er hele uttrykket for dokumentsamlingen gitt foran.



Herfra dreier spørsmålene seg om reasjonsdatbaser (**SQL**).

1. Hva ligger det i begrepet **Supernøkkel**?  
   Finn en supernøkkel for tabellen i **Tabell 1**.  
   Hva er den minimale supernøkkelen for tabellen i **Tabell 1**?

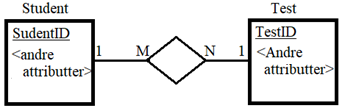
Angi en mulig primærnøkkel for tabellen i **Tabell 1**.

1. Hvilken avhengigheter har vi i **Tabell 1**?

LØSN:

**StudentId→StudentNavn   
TestID→SpmFil og AntallSpm  
TestID og SpmNr→AvgittSvar og KorrektSvar**

1. Figuren under er et skjematisk **ER**-diagram (en såkalt konseptuell modell, som bare viser hovedtrekkene i datamodellen) for en **SQL**-database **Flervalg**

****

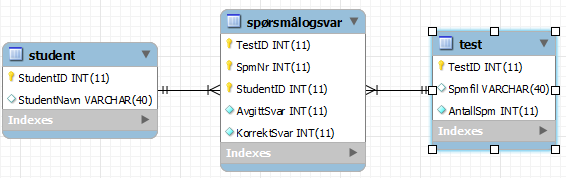
Altså: en **Student**-tabell inneholder studenter med **StudentID** som primærnøkkel, samt andre attributter. Og **Test**-tabellen inneholder primærnøkkelen **TestID** samt andre attributter.

Forklar de øvrige symbolene, og lag tabeller som kompletterer **ER**-diagrammet.  
Alle attributtene i tabellen i **Tabell 1** skal få plass.

1. Skriv **SQL**-setninger som oppretter tabellene fra **ER**-diagrammet ditt.  
   Legg vekt på å få med fremmednøkler.  
   Husk å skrive **ENGINE = InnoDB** etter hver tabelldefinisjon.

LØSN

Forslag til fullstendig ER-diagram:

  
CREATE TABLE IF NOT EXISTS Student (

StudentID INT NOT NULL,

StudentNavn VARCHAR(40),

PRIMARY KEY (StudentID)

) ENGINE=InnoDB;

CREATE TABLE IF NOT EXISTS Test (

TestID INT NOT NULL,

Spmfil VARCHAR(40),

AntallSpm INT NOT NULL,

PRIMARY KEY (TestID)

) ENGINE=InnoDB;

CREATE TABLE IF NOT EXISTS SpørsmålOgSvar (

TestID INT NOT NULL,

SpmNr INT NOT NULL,

StudentID INT NOT NULL,

AvgittSvar INT NOT NULL,

KorrektSvar INT NOT NULL,

CONSTRAINT PRIMARY KEY (TestID, SpmNr, StudentID),

CONSTRAINT FOREIGN KEY(TestID) REFERENCES Test(TestID),

CONSTRAINT FOREIGN KEY(StudentID) REFERENCES Student(StudentID)

) ENGINE=InnoDB;

1. Lag et ER-diagram for databasen din i **MySQL** **Workbench**, og lim inn et bilde av dette  
   ER-diagrammet i besvarelsen.

LØSN: Se løsnig oppg. 5

1. Sett inn dataene i **Tabell 1** i tabellene du har laget, bruk for eksempel **MySQL Workbench**.

LØSN:

INSERT INTO Student VALUES(1101, 'Birger Bang');

INSERT INTO Student VALUES(1102, 'Sigrid Sund');

INSERT INTO Student VALUES(1103, 'Arne Arntsen');

INSERT INTO Student VALUES(1104, 'Anne Anonsen');

INSERT INTO Test VALUES(1, 'IntroduksjonTilDatabaser.sql', 5);

INSERT INTO Test VALUES(2, 'SpørringerMotEnTabell.sql', 12);

INSERT INTO Test VALUES(3, 'LageOgBrukeTabeller.sql', 16);

INSERT INTO Test VALUES(4, 'AvanserteSpørringer.sql', 8);

INSERT INTO SpørsmålOgSvar VALUES(1, 1, 1101, 2, 2);

INSERT INTO SpørsmålOgSvar VALUES(1, 2, 1101, 3, 4);

INSERT INTO SpørsmålOgSvar VALUES(1, 3, 1101, 4, 4);

INSERT INTO SpørsmålOgSvar VALUES(1, 4, 1101, 2, 2);

INSERT INTO SpørsmålOgSvar VALUES(1, 5, 1101, 2, 2);

INSERT INTO SpørsmålOgSvar VALUES(1, 1, 1102, 2, 2);

INSERT INTO SpørsmålOgSvar VALUES(1, 2, 1102, 4, 4);

INSERT INTO SpørsmålOgSvar VALUES(1, 3, 1102, 4, 4);

INSERT INTO SpørsmålOgSvar VALUES(1, 4, 1102, 3, 2);

INSERT INTO SpørsmålOgSvar VALUES(1, 5, 1102, 1, 2);

1. Skriv en **SQL**-setning som skriver ut alle studentene som er registrert i databasen.

LØSN: SELECT DISTINCT StudentNavn FROM Student;

Skriv så en **SQL**-setning som skriver ut alle studentene sammen med hvilke tester de har tatt.

LØSN: SELECT S.StudentNavn, COUNT(S.StudentNavn) AS Antall

FROM Student AS S, SpørsmålOgSvar AS SS

1. Skriv en **SQL**-setning som finner antall riktige svar for hver av studentene som deltar på en gitt prøve.

LØSN: SELECT S.StudentNavn, COUNT(S.StudentNavn) AS Antall

FROM Student AS S, SpørsmålOgSvar AS SS

1. Skriv et **VIEW** basert på en av spørringene du har skrevet foran.
2. Lag ei enkel web-side som presenterer data fra databasen gitt ved det **VIEW**et du skrev foran.
3. Forklar hva ei MAC adresse er, og hvordan denne brukes.

LØSN:

MAC = Media Access Control address. Det er ei 48 eller 64 bits adresse på ethvert nettverkskort. Den skal være unik på verdensbasis. I nettverk kalles den en fysisk adresse, ettersom den er fysisk koblet til nettverkskortet/PCen. MAC-adressen brukes til adressering internt i et nettverk. Switcher bruker MAC-adressa til adressering, i motsetning til routere som bruke IP-adressa.

1. En PC har IPv4 adressen 132.16.254.1/24. Hvilke av Pc-ene a) til e) er i samme nettverk som denne?

132.16.254.254 b) 132.16.1.1 c)132.16.255.254 d)132.16.254.128 e) 32.16.254.1

LØSN: a og d

1. Forkort denne IPv6 adressen 20a1: 00b8: 0bbf: 000e:0000:0000:0000:0d00

LØSN: *Svar: 20a1:b8:bbf:e::d00*

1. Gitt adresse 2001:0DB8:0000:CD30::/60

Hvilke av de to (ingen, en eller begge) skrivemåtene nedenfor er en gyldig skrivemåte for denne adressen. Begrunn svaret:

1. 2001:0DB8::CD30:0:0:0:0/60 Gyldig. Det er bedre å bruke :: på slutten, men denne er gyldig
2. 2001:0DB8:0:CD3/60 Ugyldig, har bare 4 grupper og ingen ::
3. Adressetype Link-Local unicast er definert slik: FE80::/10. Hvilke av adressene nedenfor er en gyldig Link-Local unicast adresse
   1. FF80::/64 =1111 1111 1000 0000….
   2. FE91::7/48 = 1111 1110 1001….
   3. FE80:FE80:0000/64 = 1111 1110 1000…

De 10 første bitene skal være lik det som er angitt i FE80::/10 -> Skrevet binært: 1111 1110 1000……

* 1. FF80::/64 =1111 1111 1000 0000…. Ikke denne. De 10 første bit stemmer ikke
  2. FE91::7/48 = 1111 1110 1001…. Dette er en gyldig Link-Local unicast adesse
  3. FE80:FE80:0000/64 = 1111 1110 1000… Ikke denne. Her stemmer de 10 første bitene, men det er ei ugyldig IPV6 adresse fordi den ikke er 128 bit lang.

Svar: b.