

EKSAMEN

Emnekode: ITF10319	Emnenavn: Databasesystemer
Dato: 4.5.20	Eksamenstid: 4 timer + 15 minutter til innlevering i Inspera
Hjelpemidler: Alle, unntatt kommunikasjon	Faglærere: Tom Erik Høvring
Om gjennomføring av eksamen: Oppgavesettet består av 9 sider inklusiv denne forsiden. Merk: DEL 1 består av 4 oppgaver som du besvarer du i Word og leverer dokumentet (i PDF-format) i Inspera. DEL 2 består av 4 Multiple Choice spørsmål som du besvarer direkte i Inspera. Alle spørsmål på oppgavene skal besvares, og alle spørsmål teller likt i bedømmingen av eksamen. Alle hjelpemidler, unntatt kommunikasjon mellom kandidatene, er tillatt ved denne eksamenen. Eksamen skal være et selvstendig arbeid. Under eksamen er det ikke tillatt å kommunisere med andre personer om oppgaven, eller å dele utkast til besvarelse eller fullstendig besvarelse. Slik kommunikasjon er å anse som fusk. Innleverte eksamensbesvarelser blir kontrollert for plagiat. Innlogging Inspera og oppgavesett: Oppgavesettet blir tilgjengelig i Inspera kl. 9.00 på eksamensdagen. Logg inn på https://hiof.inspera.no med FEIDE-brukernavn og passord og last ned oppgavesettet (PDF-dokument som åpnes i et nytt vindu). Skrive besvarelse: På hver side i besvarelsen for DEL 1 skal du skrive ditt kandidatnummer (ikke navn!) og sidetall, slik: <i>side x av y</i> (for eksempel side 4 av 5). Innlevering i Inspera: Det gis 15 minutter tillegg i tid på eksamen til klargjøring av besvarelsen og innlevering i Inspera.	



Du besvarer oppgavene i DEL 1 i Word eller annet tekstbehandlingsprogram. Når du skal levere besvarelsen, lagrer du den (lagre som) PDF-fil. Så laster du opp PDF-filen i Inspera.

Husk å lagre besvarelsen for DEL 1 på din datamaskin som sikkerhetskopi før du leverer besvarelsen i Inspera!

Du besvarer oppgavene i DEL 2, Multiple Choice, direkte i Inspera.

Du vil kunne se **din samlede besvarelse med både DEL 1 og DEL 2** under *arkiv* i Inspera når eksamenstiden er utløpt.

For mer informasjon se <https://www.hiof.no/studier/eksamen/digital-eksamen/innlevering-av-oppgave.html>

Support:

Ta kontakt med eksamen-halden@hiof.no hvis du har spørsmål eller trenger veiledning om funksjonalitet i Inspera.

Sensurfrist: 25.5.20

Merk: Grunnet Covid 19-situasjonen kan det bli behov for forlengelse av sensurfristen. Studentene blir i så fall informert om utsettelsen.

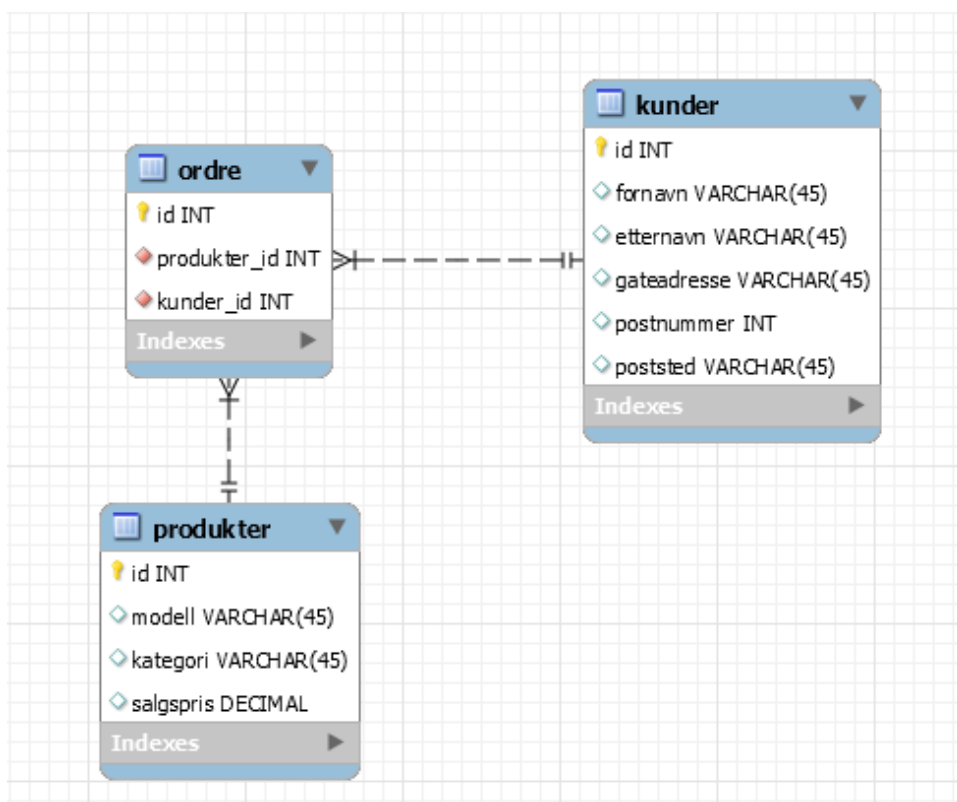
Karakterene blir publisert i Studentweb.

DEL 1, oppgave 1-4, besvares i Word og leveres inn som PDF-fil i Inspira

Oppgavesettet tar utgangspunkt i et databasesystem for en nyoppstartet nettbutikk som skal prøve å ta over markedet for salg av PC-komponenter m.m.

Oppstartsbedriften Komp.net er en nettbutikk som selger PC-deler, TVer, spillkonsoller m.m. De har lagd følgende databasemodell for oppstarten av nettstedet dems som skal holde styr på kunder, produkter og ordre. Dette er kun et førsteutkast som skal testes ut før de bestemmer seg om hvilke utvidelser dette systemet trenger.

Se bildet under for tabellstruktur. Alle kolonner markert med gule lyspærer er primærnøkler, alle kolonner markert med røde firkanter er fremmednøkler til andre tabeller. Fremmednøklerne er navngitt som <tabell>_<kolonne>.



Oppgave 1

- Finn ut hvilket produkt som er det dyreste i denne nettbutikken.
- Finn ut hvilket produkt det har blitt solgt mest av i denne nettbutikken.
- Finn ut hvilken bruker som har handlet for mest penger i denne nettbutikken.
- Finn hvilken kunde som har handlet flest av det samme produktet.
- Finn ut innenfor hvilken kategori det har blitt solgt flest produkter. Altså av alle produkter som har blitt solgt, hvilken kategori er hyppigst representert.

Oppgave 2

A) Hvilken normalform er Kunder-tabellen på? Dersom den ikke er fullstendig normalisert, gjør nødvendige endringer slik at den blir normalisert.

B) Denne strukturen gjør det veldig tungvint å håndtere at en bruker kan kjøpe flere produkter i samme ordre. Gjør nødvendige endringer slik at en ordre kan inneholde mer enn et produkt, den må både håndtere flere forskjellige produkter, og man må kunne spesifisere antallet av hvert produkt.

Oppgave 3

Systemet må utvides for å kunne dekke alle nettbutikkens behov. Du skal i denne oppgaven utvide databasesystemet vi har jobbet med hittil. Husk å ta med de originale tabellene kunder, ordre, og produkter, samt endringer du har måttet gjøre i tidligere oppgaver.

- Nettsiden trenger også å lagre produsentene av hvert produkt de har slik at de kan implementere et filter som sorterer etter produsent. De fleste produsenter produserer mer enn et produkt.
- Det er ønskelig at kundene skal kunne gi tilbakemelding på produkter, da i form av en anmeldelse på en tall-skala (fra 1 til 10) og en kort tilbakemelding på opptil 1000 tegn, hver kunde kan maksimalt anmelde et produkt én gang.
- Nettstedet har nå kun informasjon om produkter de har solgt, det er også ønskelig å lage en database som kan loggføre alle produktene de kjøper inn. De trenger altså en måte å lagre bestillinger på, og hver bestilling må kunne inneholde flere produkter og med et spesifisert antall per produkt. Det vil også nå være ønskelig å skille mellom salgspris og innkjøpspris.
- Et problem var at brukere som lette etter skjermkort ikke kunne se alle skjermkortene når de filtrerte på nettsiden, da noen skjermkort hadde blitt lagret med kategori 'grafikkort' eller 'sjermkort' (altså med skrivefeil). Gjør justeringer slik at dette ikke skjer i fremtiden.

Oppgave 4

A) Nå som du har redesignet databasen, kan du forklare hvorfor man ikke trenger å lagre hvor mange av hvert produkt man har på lager – og hvordan man likevel kan vite hvor mange man har på lager.

B) Hvordan kan vi sørge for at en bruker kun får lov til å se på data i databasen vår, men ikke legge til, slette eller oppdatere data? Man trenger ikke å svare med syntaks, et kort forklaring med egne ord er nok.

Syntaksliste

Select-setning

SELECT kolonne(r), <string>

FROM tabell(er)

<**INNER, OUTER, LEFT, RIGHT**> **JOIN** tabell **ON** kriterie

WHERE kriterie(r)

IN | BETWEEN | AND | OR | XOR

GROUP BY kolonne(r)

HAVING gruppekriterie(r)

ORDER BY kolonne **ASC/DESC**

Øvrige funksjoner

- **CONCAT**(kolonne, <string>) - Slår sammen kolonneverdier og eller strenger, tar så mange parametre man vil.
- **FLOOR**(kolonne) - Runder av en kolonneverdi nedover til nærmeste heltall.
- **ROUND**(kolonne, desimaler) - Runder av en kolonneverdi til spesifisert antall desimaler.
- **POWER**(grunntall, eksponent) - For å lage en potens.
- **RAND()** - Returnerer et tilfeldig tall, eller en tilfeldig oppføring i en spørring

Aggregatfunksjoner:

SUM - Summerer verdiene av en kolonne for en spørring

MIN – Henter den laveste verdien fra en spørring

MAX - Henter den høyeste verdien fra en spørring

AVG – Henter gjennomsnittsverdien fra en spørring

COUNT - Henter antall resultater fra en spørring

Insert-setning

INSERT INTO tabell (kolonne1, kolonne2, kolonne3 ...)

VALUES (verdi1, verdi2, verdi3 ...)

Update-setning

UPDATE tabell

SET kolonne1 = verdi1, kolonne2 = verdi2

WHERE kriterie(r)

Delete-setning

DELETE

FROM tabell

WHERE kriterie(r)

Views

CREATE VIEW navn

AS <select-setning>

DEL 2, Fire (4) Multiple Choice-spørsmål besvares direkte i Inspira. MERK: Du får spørsmålene og svaralternativene i tilfeldig rekkefølge i Inspira.

Hva er hovedformålet med views?

Velg ett alternativ

- Views gjør det lettere å oppdatere oppføringer i databasen.
- Views blir brukt som et sikkerhetstiltak som kan skjule enkelte kolonner fra brukeren.
- Views gjør spørringer raskere ved å forhåndslagre dem.

Hva er den største fordel/ hovedegenskapen til GraphDB?

Velg ett alternativ

- Den er den raskeste databasetypen, men kan kun håndtere tall som datatype.
- Man kan legge med informasjon direkte på relasjoner mellom noder, i tillegg til informasjonen nodene allerede har.
- Den kan lagre og tegne grafer mye raskere enn andre databasetyper. Grafene kan også oppdateres i real-time.

Hva er hovedgrunnen til at vi bruker fremmednøkler?

Velg ett alternativ

- For å sikre referanseintegriteten, altså at man vet at dataene man kobler noe opp mot finnes.
- For å muliggjøre sammenslåing av to eller flere tabeller i en spørring.
- For at det skal gå raskere å hente data fra to eller flere tabeller i én spørring.

Et problem som kan oppstå er følgende: Butikken har kun et eksemplar igjen av et produkt, og to stk kjøper den samtidig, troende at det fortsatt finnes på lager. Dermed har to kunder betalt for det samme produktet, men kun en av dem kan få produktet. Hvordan kan vi unngå dette?

Velg ett alternativ

- Ved å gjøre om databasen vår til en NoSQL database.
- Ved å ta back-up av databasen hver dag, og lage et script som ser over disse daglig.
- Ved å ta i bruk ACID-protokollen.