



Høgskolen i Telemark

EKSAMEN

6065 DATABASER OG WEB

04.05.2015

Tid:	9-13
Målform:	Bokmål/nynorsk
Sidetall:	5 med forside
Hjelpemidler:	Et maskinskrevet notat på maksimalt 2 A4-sider, satt med enkel linjeavstand og skriftstørrelse 12 (eller større).
Merknader:	Eksamen blir gjennomført på PC uten tilgang på Internett, men med Firefox, MySQL Workbench, XAMPP og Sublime.
Vedlegg:	Et SQL-skript eksamen.sql, HTML-filer oppg1a.html og oppg1f.html, samt et PHP-skript oppg1f.php. Alle filene er lagret på mappe c:\hit.

Eksamensresultatene blir offentliggjort på StudentWeb.



Fakultet for allmennvitenskapelige fag



Bokmål

Prosentatsen ut for hver oppgave antyder vekt ved sensur. Deloppgaver er vektet likt. Prøv å svare på så mange spørsmål som mulig. Lykke til!

Oppgave 1 (60 %)

Tabellene under er tenkt brukt i en web-applikasjon for gjennomføring av konferanser. Primærnøkler er understreket og fremmednøkler er merket med ei stjerne.

- rom(romkode, ant_plasser)
- person(epost, fornavn, etternavn)
- presentasjon(pid, romkode*, dato, kl_fra, kl_til, epost*, tittel)
- deltakelse(pid*, epost*)

Databasen lagrer informasjon om én enkelt konferanse. Konferansen kan pågå et antall dager, og det er et antall presentasjoner (foredrag) i ulike rom hver dag (parallele sesjoner). **Navn og epost til både foredragsholdere og deltakere blir lagret i tabellen person.** Kolonnen **presentasjon.epost** viser til **foredragsholder**. Kolonnene ant_plasser og pid inneholder heltall, dato inneholder datoer, kl_fra og kl_til inneholder klokkeslett. Øvrige kolonner inneholder tekster. I tabellen presentasjon så er pid auto-nummerert.

Et SQL-skript eksamen.sql som oppretter tabellene og setter inn eksempeldata er lagt inn på hver PC under mappe c:\hit. På samme mappe finner du også HTML-filer oppg1a.html og oppg1f.html, samt et PHP-skript oppg1f.php. Forberedelse:

- Start XAMPP, kjør SQL-skriptet i phpMyAdmin og gjør deg kjent med tabellene.

Noen JavaScript-tips:

- Uttrykket s.substring(2, 5) gir "lko" hvis s er lik "velkommen".
- Funksjonen Number konverterer verdier til tall.
- Funksjonen isNaN sjekker om en verdi er et tall.
- Uttrykket ptn.test(verdi) returnerer true hvis det regulære uttrykket ptn matcher verdi.

Du skal nå løse følgende programmeringsoppgaver:

- a. Ta utgangspunkt i filen oppg1a.html. Den inneholder et HTML-skjema for å legge til en ny presentasjon. Du skal *ikke* gjøre endringer i denne filen. Du skal gjøre dette:
 - Legg til CSS-regler (på filen oppg1a.css) som sørger for at eventuelt innhold i elementet med **id lik melding** blir vist med **rød farge**, at alle **obligatoriske ledetekster** (label inne i div med class lik «obligatorisk») blir vist med fet skrift, og at det blir lagt til litt luft mellom hvert inndatafelt.
 - Skriv JavaScript-kode (på filen oppg1a.js) som sjekker inndata fra tekstfeltene for dato og klokkeslett, der datoer skal skrives på formen ÅÅÅÅ-MM-DD og klokkeslett på formen TT:MM. Du skal sjekke at både dato og klokkeslett er lovlig, men du kan anta at februar alltid har 28 dager. Kontrollen med klokkeslett skal gjøres kun ved hjelp av regulære uttrykk.



- b. Lag et PHP-skript for å behandle skjemaet fra oppgave 1a. Skriptet skal først sjekke at rom og foredragsholder finnes. Hvis dette er i orden skal presentasjonen lagres i databasen og en kvittering skrives ut, ellers skal det kun skrives ut en passende feilmelding. Sørg også for å sikre skjemaet mot SQL injection.
- c. Lag et PHP-skript som viser programmet for en gitt dag, sortert først etter rom og så etter klokkeslett. For hver presentasjon skal tittelen bli vist som en h2-overskrift, og øvrige opplysninger i et avsnitt rett under. Du kan anta at datoen blir sendt med som en GET-parameter.
- d. Lag et PHP-skript som genererer og setter inn testdata i tabellen rom. La antall rom være GET-parameter. Navn på rommene skal bestå av en tilfeldig bokstav etterfulgt av en bindestrek og så et tall fra 1 og oppover, f.eks. B-1, B-2, B-3, B-4 for antall rom lik 4. Antall plasser i rommene skal være et tilfeldig heltall mellom 5 og 300.
- e. Lag en trigger som kontrollerer at det er ledig plass ved påmelding av ny deltaker. Triggeren skal kaste et unntak hvis det er fullt. Forklar også hvordan denne triggeren kan testes.
- f. Lag en HTML-side med et AJAX-basert søkeskjema for å finne presentasjoner. Brukeren skal kunne skrive inn deler av tittelen på en presentasjon. Hvis det finnes presentasjoner som matcher søkeordet skal disse titlene bli vist i en punktliste. Her kan du bygge videre på ferdiglaget kode i filene oppg1f.html og oppg1f.php.

Oppgave 2 (20 %)

Svar kort på følgende deloppgaver:

- a. Forklar forskjellene mellom enkle og doble apostrofer i PHP.
- b. Hva inneholder de assosiative tabellene \$_GET og \$_POST og hva brukes de til?
- c. Forklar hvordan man bruker sesjonsvariabler i PHP og gi et eksempel på bruk av slike variabler.
- d. Hva menes med SQL injection og hvordan kan man sikre seg mot dette?
- e. Nevn noen bruksområder for rad-triggere og forklar innholdet i OLD og NEW.
- f. Forklar hva et regulært uttrykk er og gi et eksempel på hva det kan brukes til.
- g. Hva er forskjellen på et velformet og et gyldig XML-dokument?
- h. Forklar hva som skjer i et AJAX-kall.
- i. Hvilke fordeler ser du ved bruk av ORM?
- j. Forklar sammenhengen mellom begrepene SOAP, WSDL og XML.

Oppgave 3 (20 %)

Gi et mer utfyllende svar på følgende deloppgaver:

- a. Forklar hvordan en web-applikasjon utviklet med PHP, JavaScript, SQL og CSS blir utført. Legg vekt på å beskrive hvordan ulike deler av programkoden blir utført på tjenersiden og klientsiden. Altså, hva skjer på databasetjeneren? Hva skjer på web-tjeneren og hva gjøres av nettleseren på klient-maskinen?
- b. Skriv en innføring i jQuery. Du kan anta at leseren har god forståelse for HTML, CSS og JavaScript.



Prosentatsen ut for kvar oppgåve antyder vekt ved sensur. Deloppgåver er vekta likt. Freist å svare på så mange spørsmål som mogleg. Lykke til!

Oppgåve 1 (60 %)

Tabellane under er tenkt nytta i ein web-applikasjon for gjennomføring av konferansar. Primærnøklar er understreka og framandnøklar er merkte med ei stjerne.

- rom(romkode, ant_plasser)
- person(epost, fornavn, etternavn)
- presentasjon(pid, romkode*, dato, kl_fra, kl_til, epost*, tittel)
- deltakelse(pid*, epost*)

Databasen lagrar informasjon om ein einskild konferanse. Konferansen kan strekke seg over fleire dagar, og det er og fleire presentasjonar (foredrag) i ulike rom kvar dag (parallele sesjonar). Namn og epost til både foredragshaldarar og deltakarar blir lagra i tabellen person. Kolonnen presentasjon.epost viser til foredragshaldar. Kolonnane ant_plasser og pid inneheld heiltal, dato inneheld datoar, kl_fra og kl_til inneheld klokkeslett. Dei andre kolonnane inneheld tekstar. I tabellen presentasjon så er pid auto-nummerert.

Eit SQL-skript eksamen.sql som opprettar tabellane og set inn døme på data er lagt inn på kvar PC under mappe c:\hit. På same mappe finn du og HTML-filer oppg1a.html og oppg1f.html, samt eit PHP-skript oppg1f.php. Førebuing:

- Start XAMPP, køyr SQL-skriptet i phpMyAdmin og gjer deg kjent med tabellane.

Nokre JavaScript-tips:

- Uttrykket s.substring(2, 5) gir "lko" viss s er lik "velkommen".
- Funksjonen Number konverterer verdiar til tal.
- Funksjonen isNaN sjekkar om ein verdi er eit tal.
- Uttrykket ptn.test(verdi) returnerer true viss det regulære uttrykket ptn matcher verdi.

Du skal nå løyse desse programmeringsoppgåvene:

- a. Ta utgangspunkt i fila oppg1a.html. Den inneheld eit HTML-skjema for å legge til ein ny presentasjon. Du skal *ikkje* gjere endringar i denne fila. Du skal gjere dette:
 - Legg til CSS-reglar (på fila oppg1a.css) som syt for at eventuelt innhald i elementet med id lik melding blir vist med raud farge, at alle obligatoriske ledetekstar (label inne i div med class lik «obligatorisk») blir vist med feit skrift, og at det blir lagt til litt luft mellom kvart inndatafelt.
 - Skriv JavaScript-kode (på fila oppg1a.js) som sjekkar inndata frå tekstfeltene for dato og klokkeslett, der datoar skal vere på forma ÅÅÅÅ-MM-DD og klokkeslett på forma TT:MM. Du skal sjekke at både dato og klokkeslett er lovlege, men du kan anta at februar alltid har 28 dagar. Kontrollen med klokkeslett *skal* gjerast berre ved hjelp av regulære uttrykk.



- b. Lag eit PHP-skript for å handsame skjemaet frå oppgåve 1a. Skriptet skal først sjekke at rom og foredragshaldar finst. Viss dette er i orden skal presentasjonen lagrast i databasen og en kvittering skrivast ut, elles skal det berre skrivast ut ein passende feilmelding. Syt og for å sikre skjemaet mot SQL injection.
- c. Lag eit PHP-skript som viser programmet for ein gitt dag, sortert først etter rom og så etter klokkeslett. For kvar presentasjon skal tittelen bli vist som ein h2-overskrift, og dei andre opplysningane samla i eit avsnitt rett under. Du kan anta at datoen blir sendt med som ein GET-parameter.
- d. Lag eit PHP-skript som genererer og set inn testdata i tabellen rom. La talet på rom være GET-parameter. Namn på romma skal starte med ein tilfeldig bokstav, så ein bindestrek og så eit tal frå 1 og oppover, til dømes B-1, B-2, B-3, B-4 for tal på rom lik 4. Talet på plassar i romma skal være eit tilfeldig heiltal mellom 5 og 300.
- e. Lag ein trigger som sjekkar om det er ledig plass ved påmelding av ny deltakar. Triggeren skal kaste eit unntak viss det er fullt. Forklar og korleis ein kan teste denne triggeren.
- f. Lag ei HTML-side med eit AJAX-basert søkeskjema for å finne presentasjonar. Brukaren skal kunne skrive inn deler av tittelen på ein presentasjon. Viss det finnes presentasjonar som matcher søkeordet skal desse titlane bli vist i ei punktliste. Her kan du byggje vidare på ferdiglaga kode i filene oppg1f.html og oppg1f.php.

Oppgåve 2 (20 %)

Svar kort på følgjande deloppgåver:

- a. Forklar forskjellane mellom enkle og doble apostrofer i PHP.
- b. Kva inneheld dei assosiative tabellane \$_GET og \$_POST og kva vert dei nytta til?
- c. Forklar korleis ein nyttar sesjonsvariablar i PHP og gje eit døme på korleis ein nyttar slike variablar.
- d. Kva meiner ein med SQL injection og korleis kan ein sikre seg mot dette?
- e. Nemn nokre bruksområder for rad-triggere og forklar innhaldet i OLD og NEW.
- f. Forklar kva eit regulært uttrykk er og gje eit døme på kva ein kan nytte det til.
- g. Kva er forskjellen på eit velforma og eit gyldig XML-dokument?
- h. Forklar kva som skjer i eit AJAX-kall.
- i. Kva for fordelar ser du ved bruk av ORM?
- j. Forklar sammenhengen mellom omgrepa SOAP, WSDL og XML.

Oppgåve 3 (20 %)

Gje eit meir utfyllande svar på følgjande deloppgåver:

- a. Forklar korleis ein web-applikasjon utvikla med PHP, JavaScript, SQL og CSS blir utført. Legg vekt på å beskrive korleis ulike delar av programkoden blir utført på tenarsida og klientsida. Altså, kva skjer på databasetenaren? Kva skjer på web-tenaren og kva gjer nettlesaren på klient-maskina?
- b. Skriv ein innføring i jQuery. Du kan anta at lesaren har god forståing for HTML, CSS og JavaScript.