



Høgskolen i Telemark

**EKSAMEN
5606 Internett**

08.05.2009

Tid:	5 timer
Målform:	Bokmål / nynorsk
Sidetall:	9 med framside
Hjelpemidler:	Alle trykte og skrevne
Merknader:	Ingen
Vedlegg:	A: Nettside og B: Database

Eksamensresultata blir offentliggjort på nettet, via Arena høgskole. I tillegg finn du eksamensresultatlistene på utsida av eksamenskontoret. Desse listene opplyser kun kandidatnummer og karakter, så hugs kandidatnummeret ditt frå eksamen.



Bokmål

Prosentsetningen ut for hver oppgave antyder vektning ved sensur.

Oppgaven bygger på et gjennomgående eksempel: Bedriften *XYZ Hytteutleie* leier ut hytter. Bedriften vil nå utvikle et nettsted med informasjon om hyttene og aktiviteter i nærområdet. Det skal også være mulig å søke etter hytter og å reservere hytter.

Oppgave 1 (30%)

1-a (20%)

Vedlegg A viser hjemmesiden til XYZ Hytteutleie. Lag en HTML-fil koblet til en CSS stilark-fil slik at resultatet blir som vist i vedlegget. Legg merke til følgende:

- Overskrifter er satt med font Verdana, annen tekst er satt med font Georgia.
- Skriftfarge og bakgrunnsfarge er endret.
- Innholdet er plassert i en boks med tykk, rød ramme. Det er ekstra luft både på innsiden (10px) og utsiden (20px) av denne rammen.
- Boksen fyller 80% av bredden til nettleservinduet.
- Teksten er blokkjustert (både rett venstremarg og rett høyremarg).
- Første bokstav i hvert avsnitt er ekstra stor.
- Listenummerering er tilpasset.
- Overskriftene i tabellen er blå, mens teksten i øvrige tabellceller er rød. Tabellen har ikke kantlinjer, men det er kantlinjer rundt hver tabellcelle, og det er ekstra luft (5px) på innsiden av disse kantlinjene.
- Når man hviler musmarkøren over hyperlenker blir understrekingen fjernet.

1-c (5%)

Forklar hvordan CSS kan brukes for å plassere elementer, som for eksempel en logo, i en bestemt posisjon på en nettside.

1-d (5%)

Forklar hva som må til for at en nettside skal tilfredsstillende XHTML-standarden, sammenlignet med HTML-standarden.

Oppgave 2 (50%)

Denne oppgaven går ut på å utvikle deler av en web-applikasjon for XYZ Hytteutleie ved hjelp av PHP, HTML og MySQL. Vedlegg B inneholder en beskrivelse av databasen.

Forberedelse

Hvis du vil teste PHP-koden du lager kan du bruke Wamp. Start Wamp-tjeneren og bygg hyttedatabasen ved hjelp av SQL-skriptet i vedlegg B. Du må skrive inn SQL-koden, men du velger selv hvor mange eksempel-rader du vil sette inn. SQL-skriptet kan så kjøres fra phpMyAdmin, som er tilgjengelig fra Wamp-menyen. Lagre alle PHP-skript på ny katalog C:\wamp\www\eksamen. Testing av PHP-skript kan da gjøres i nettleseren på adresse <http://localhost/eksamen>. **Lagre filene dine ofte!**



2-a (10%)

Web-løsningen skal inneholde et skjema for hyttereservasjon. Brukeren skal skrive inn sitt eget telefonnr, hyttenr, fra-uke og til-uke. Skjemadata skal sendes til PHP-skriptet `reservere.php` med metode POST (se oppgave 2-b). Skriv HTML-koden for dette skjemaet. Du velger selv hvordan skjemaet skal se ut.

2-b (15%)

Lag PHP-skriptet `reservere.php` for å lagre en hyttereservasjon i databasen. Telefonnr til kunden, hyttenr, fra-uke og til-uke blir overført som skjemadata med metode POST (se oppgave 2-a). Skriptet skal sjekke at fra-uke ikke er større enn til-uke (det er ikke mulig å reservere hytter fra slutten av ett år til begynnelsen av neste). Hvis dette er i orden skal skriptet skrive ut en kvittering som viser hyttenr og tidsrom. Det er ikke nødvendig å sjekke om hytta allerede er reservert.

2-c (15%)

Web-applikasjonen inneholder et skjema for å søke etter hytter (skal ikke lages). I dette skjemaet kan brukeren fylle inn minimum antall sengeplasser og maksimumpris pr. uke. Lag et skript `hyttesok.php` som viser resultatet av hyttesøket, presentert som en HTML-tabell med kolonneoverskrifter. Du kan anta at skjemadata blir overført med metode GET. Det betyr at `hyttesok.php` kan testes ved å skrive skjemadata inn i URLen.

2-d (10%)

Forklar hvordan web-applikasjonen til XYZ Hytteutleie kan sikres mot såkalt SQL injection, gjerne med eksempelkode.

Oppgave 3 (20%)

3-a (10%)

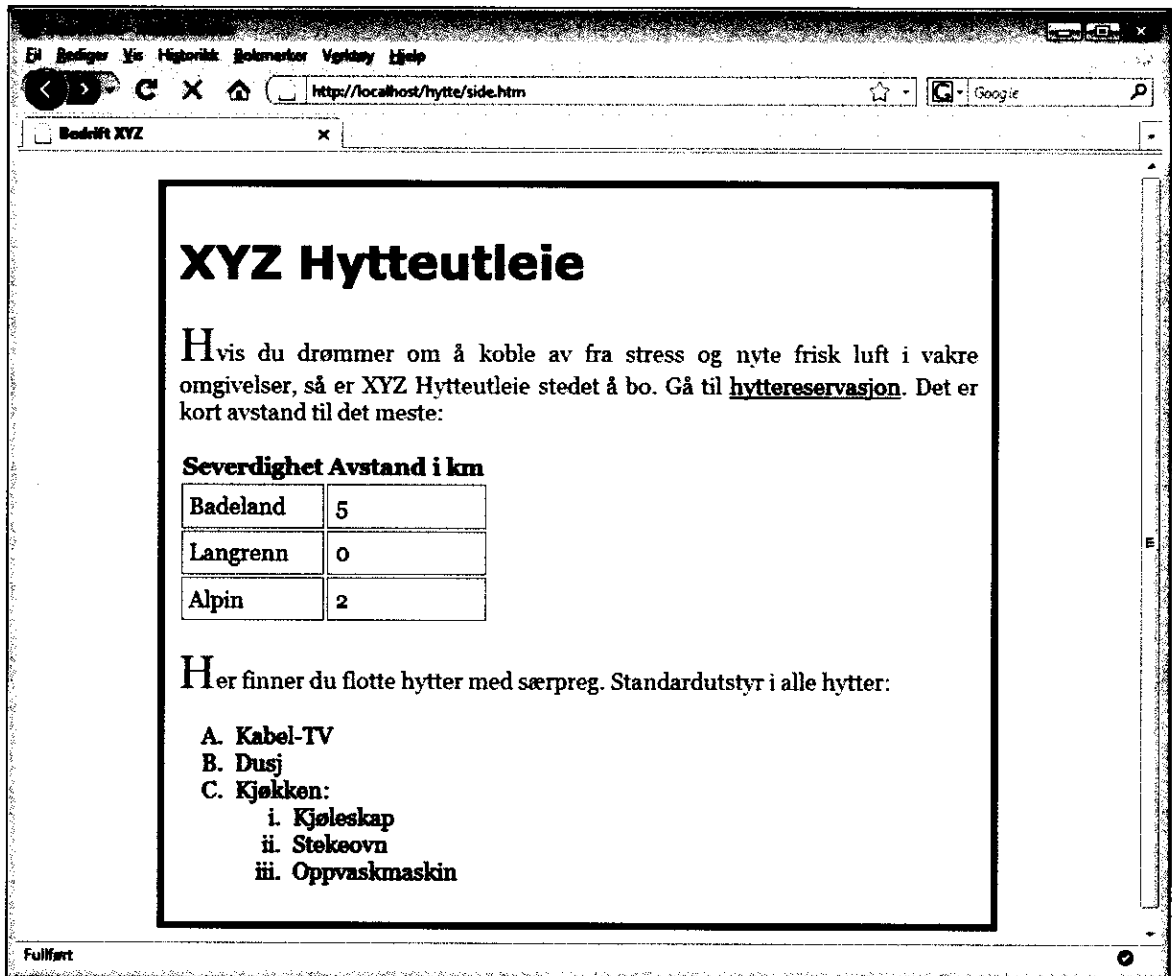
XYZ Hytteutleie ønsker å bygge ut nettstedet med informasjon om turløyper og aktiviteter i nærområdet. Flere ansatte skal lage nettsider. Alle nettsidene bør ha samme utforming. Bedriften vurderer å bruke rammer (HTML Frames). Drøft fordeler og ulemper ved en slik løsning.

3-b (10%)

I hvilke situasjoner er det behov for å bruke sesjonsvariabler i PHP? Beskriv en utvidelse av web-løsningen til XYZ Hytteutleie som vil kreve dette. Gi eksempler på relevant PHP-kode.



Vedlegg A: Nettside



Vedlegg B: Database

Databasen til XYZ Hytteutleie består av to tabeller: Hytte og Reservasjon. Om hver hytte er det lagret et unikt hyttenr, en kort beskrivelse, antall sengeplasser og ukepris. Hyttene leies ut på ukebasis (lørdag til lørdag). En reservasjon gjelder én hytte. Om hver reservasjon er det lagret et unikt reservasjonsnr, hyttenr, kundens telefonnr, fra-uke og til-uke.

SQL-koden under lager databasetabellene og setter inn noen rader med eksempeldata. I tabellen Reservasjon er primærnøkkelen ResNr autonummerert. Databasemotor InnoDB sørger for kontroll med fremmednøkler (ENGINE = InnoDB). For testing kan du bruke Wamp og følgende oppkoblingsparametre:

- Tjener: "localhost"
- Brukernavn: "root"
- Passord: "" (tomt)
- Database: "hytteutleie" (hvis du bruker skriptet under)



```
CREATE DATABASE hytteutleie;

USE hytteutleie;

CREATE TABLE Hytte
(
  HytteNr      INTEGER,
  Beskrivelse  VARCHAR(100),
  Sengeplasser SMALLINT,
  Ukepris      DECIMAL(8,2),
  CONSTRAINT HyttePK PRIMARY KEY(HytteNr)
) ENGINE=InnoDB;

CREATE TABLE Reservasjon
(
  ResNr      INTEGER AUTO_INCREMENT,
  TlfNr      VARCHAR(15),
  HytteNr    INTEGER,
  FraUke     SMALLINT,
  TilUke     SMALLINT,
  CONSTRAINT ReservasjonPK PRIMARY KEY (ResNr),
  CONSTRAINT ReservasjonHytteFK FOREIGN KEY (HytteNr)
  REFERENCES Hytte(HytteNr)
) ENGINE=InnoDB;

INSERT INTO Hytte(HytteNr,Beskrivelse,Sengeplasser,Ukepris)
VALUES (1,'Enkel hytte, sørsiden av fjellet',4,5500);

INSERT INTO Hytte(HytteNr,Beskrivelse,Sengeplasser,Ukepris)
VALUES (2,'Solrik hytte, rett ved turløypene',6,8000);

INSERT INTO Hytte(HytteNr,Beskrivelse,Sengeplasser,Ukepris)
VALUES (3,'Moderne hytte med alle fasiliteter',4,11500);

INSERT INTO Reservasjon(TlfNr,HytteNr,FraUke,TilUke)
VALUES ('12345678',3,3,5);

INSERT INTO Reservasjon(TlfNr,HytteNr,FraUke,TilUke)
VALUES ('44332211',1,4,6);

INSERT INTO Reservasjon(TlfNr,HytteNr,FraUke,TilUke)
VALUES ('87654321',2,5,5);

INSERT INTO Reservasjon(TlfNr,HytteNr,FraUke,TilUke)
VALUES ('18273645',3,6,6);

INSERT INTO Reservasjon(TlfNr,HytteNr,FraUke,TilUke)
VALUES ('81726354',2,6,9);

INSERT INTO Reservasjon(TlfNr,HytteNr,FraUke,TilUke)
VALUES ('45632781',3,7,7);

COMMIT;
```



Prosentsetningen ut for kvar oppgåve antyder vektning ved sensur.

Oppgåva bygger på eit gjennomgåande døme: Bedrifta *XYZ Hytteutleige* leiger ut hytter. Bedrifta vil no utvikle ein nettstad med informasjon om hyttene og aktivitetar i nærområdet. Det skal òg være mogleg å søke etter hytter og å reservere hytter.

Oppgåve 1 (30%)

1-a (20%)

Vedlegg A syner heimesida til XYZ Hytteutleige. Lag ei HTML-fil kopla til ei CSS stilark-fil slik at resultatet blir som i vedlegget. Legg merke til følgjande:

- Overskrifter er satt med font Verdana, annan tekst er satt med font Georgia.
- Skriftfarge og bakgrunnsfarge er endra.
- Innhaldet er plassert i ein boks med tjukk, raud ramme. Det er ekstra luft både på innsida (10px) og utsida (20px) av denne ramma.
- Boksen fyller 80% av bredda til nettlesarvindauget.
- Teksten er blokkjustert (både rett venstremarg og rett høgremarg).
- Første bokstav i kvart avsnitt er ekstra stor.
- Listenummereringa er tilpassa.
- Overskriftene i tabellen er blå, mens teksten i andre tabellceller er raud. Tabellen har ikkje kantliner, men det er kantliner rundt kvar tabellcelle, og det er ekstra luft (5px) på innsida av desse kantlinene.
- Når ein kviler musmarkøren over hyperlenker blir understrekinga fjerna.

1-c (5%)

Forklar korleis CSS kan nyttast for å plassere element, som til dømes ein logo, i ein bestemt posisjon på ei nettside.

1-d (5%)

Forklar kva som må til for at ei nettside skal tilfredsstillе XHTML-standarden, samanlikna med HTML-standarden.

Oppgåve 2 (50%)

Denne oppgåva går ut på å utvikle deler av ein web-applikasjon for XYZ Hytteutleige ved hjelp av PHP, HTML og MySQL. Vedlegg B skildrar databasen.

Førebuing

Viss du vil teste PHP-koden du lager kan du nytte Wamp. Start Wamp-tenaren og bygg hyttedatabasen ved hjelp av SQL-skriptet i vedlegg B. Du må skrive inn SQL-koden, men du vel sjølv kor mange rader du vil sette inn. SQL-skriptet kan så køyrast frå phpMyAdmin, som er tilgjengeleg frå Wamp-menyen. Lagre alle PHP-skripta på ny katalog C:\wamp\www\eksamen. Du kan i så fall teste PHP-skripta i nettlesaren på adresse <http://localhost/eksamen>. **Lagre filene dine ofte!**



2-a (10%)

Web-løysinga skal innehalde eit skjema for hytterreservasjon. Brukaren skal skrive inn sitt eige telefonnr, hyttenr, frå-veke og til-veke. Skjemadata skal sendast til PHP-skriptet `reservere.php` med metode POST (sjå oppgåve 2-b). Skriv HTML-koden for dette skjemaet. Du vel sjølv korleis skjemaet skal sjå ut.

2-b (15%)

Lag PHP-skriptet `reservere.php` for å lagre ein hytterreservasjon i databasen. Telefonnr til kunden, hyttenr, frå-veke og til-veke blir overført som skjemadata med metode POST (sjå oppgåve 2-a). Skriptet skal sjekke at frå-veke ikkje er større enn til-veke (det er ikkje mogleg å reservere hytter frå slutten av eit år til starten av neste). Viss dette er i orden skal skriptet skrive ut ein kvittering som syner hyttenr og tidsrom. Det er ikkje naudsynt å sjekke om hytta alt er reservert.

2-c (15%)

Web-applikasjonen inneheld eit skjema for å søke etter hytter (skal ikkje lages). I dette skjemaet kan brukaren fylle inn minimum tal på sengeplassar og maksimumpris pr. veke. Lag eit skript `hyttesok.php` som syner resultatet av hyttesøket, presentert som ein HTML-tabell med kolonneoverskrifter. Du kan anta at skjemadata blir overført med metode GET. Det betyr at du kan teste `hyttesok.php` ved å skrive skjemadata inn i URLen.

2-d (10%)

Forklar korleis web-applikasjonen til XYZ Hytteutleige kan sikrast mot såkalla SQL injection, gjerne med døme på programkode.

Oppgåve 3 (20%)

3-a (10%)

XYZ Hytteutleige ønskjer å byggje ut nettstaden med informasjon om turløyper og aktivitetar i nærområdet. Fleire tilsette skal lage nettsider. Alle nettsidene bør ha same utforming. Bedrifta vurderer å nytte rammer (HTML Frames). Drøft fordelar og ulemper ved ei slik løysing.

3-b (10%)

I kva for situasjonar er det behov for å nytte sesjonsvariablar i PHP? Skildre ei utviding av web-løysinga til XYZ Hytteutleige som vil krevje dette. Gje døme på relevant PHP-kode.



Vedlegg A: Nettside

XYZ Hytteutleie

Hvis du drømmer om å koble av fra stress og nyte frisk luft i vakre omgivelser, så er XYZ Hytteutleie stedet å bo. Gå til [hyttereservasjon](#). Det er kort avstand til det meste:

Severdighet Avstand i km

Badeland	5
Langrenn	0
Alpin	2

Her finner du flotte hytter med særpreg. Standardutstyr i alle hytter:

- A. Kabel-TV
- B. Dusj
- C. Kjøkken:
 - i. Kjøleskap
 - ii. Stekeovn
 - iii. Oppvaskmaskin

Fullført

Vedlegg B: Database

Databasen til XYZ Hytteutleige inneheld to tabellar: Hytte og Reservasjon. Om kvar hytte er det lagra eit unikt hyttenr, ei kort omtale, talet på sengeplassar og vekepris. Hyttene blir leigde ut på vekebasis (laurdag til laurdag). Ein reservasjon gjeld ei hytte. Om kvar reservasjon er det lagra eit unikt reservasjonsnr, hyttenr, kundens telefonnr, frå-veke og til-veke.

SQL-koden under lager databasetabellane og set inn nokre rader med data. I tabellen Reservasjon er primærnøkkelen ResNr autonummerert. Databasemotor InnoDB syt for kontroll med framandnøklar (ENGINE = InnoDB). For testing kan du nytte Wamp og følgjande oppkoblingsparametar:

- Tjener: "localhost"
- Brukarnamn: "root"
- Passord: "" (tomt)
- Database: "hytteutleige" (viss du nyttar skriptet under)



```
CREATE DATABASE hytteutleige;

USE hytteutleige;

CREATE TABLE Hytte
(
  HytteNr      INTEGER,
  Omtale       VARCHAR(100),
  Sengeplassar SMALLINT,
  Vekepris     DECIMAL(8,2),
  CONSTRAINT HyttePK PRIMARY KEY(HytteNr)
) ENGINE=InnoDB;

CREATE TABLE Reservasjon
(
  ResNr      INTEGER AUTO_INCREMENT,
  TlfNr      VARCHAR(15),
  HytteNr    INTEGER,
  FraaVeke   SMALLINT,
  TilVeke    SMALLINT,
  CONSTRAINT ReservasjonPK PRIMARY KEY (ResNr),
  CONSTRAINT ReservasjonHytteFK FOREIGN KEY (HytteNr)
  REFERENCES Hytte(HytteNr)
) ENGINE=InnoDB;

INSERT INTO Hytte(HytteNr,Omtale,Sengeplassar,Vekepris)
VALUES (1,'Enkel hytte, sørsiden av fjellet',4,5500);

INSERT INTO Hytte(HytteNr,Omtale,Sengeplassar, Vekepris)
VALUES (2,'solrik hytte, rett ved turløypene',6,8000);

INSERT INTO Hytte(HytteNr,Omtale,Sengeplassar, Vekepris)
VALUES (3,'Moderne hytte med alle fasiliteter',4,11500);

INSERT INTO Reservasjon(TlfNr,HytteNr,FraaVeke,TilVeke)
VALUES ('12345678',3,3,5);

INSERT INTO Reservasjon(TlfNr,HytteNr,FraaVeke,TilVeke)
VALUES ('44332211',1,4,6);

INSERT INTO Reservasjon(TlfNr,HytteNr,FraaVeke,TilVeke)
VALUES ('87654321',2,5,5);

INSERT INTO Reservasjon(TlfNr,HytteNr,FraaVeke,TilVeke)
VALUES ('18273645',3,6,6);

INSERT INTO Reservasjon(TlfNr,HytteNr,FraaVeke,TilVeke)
VALUES ('81726354',2,6,9);

INSERT INTO Reservasjon(TlfNr,HytteNr,FraaVeke,TilVeke)
VALUES ('45632781',3,7,7);

COMMIT;
```