

**Kirjasto**  
**Relaatiotietokannat**  
**Kevät 2001**

Auvinen Annemari  
Niemi Anu  
Passoja Jonna  
Pulli Jari  
Tersa Tiina

**Harjoitustyö**  
**7.4.2001**

## Sisällysluettelo

<b>1. Yleiskuvaus</b> .....	<b>3</b>
<b>2. Vaatimukset</b> .....	<b>3</b>
2.1. Toiminnalliset .....	3
2.1.1. Sisäänkirjautuminen.....	3
2.1.2. Nimikkeiden lisääminen .....	3
2.1.3. Nimikkeiden poistaminen .....	3
2.1.4. Nimikkeiden muokkaaminen .....	3
2.1.5. Nimikkeiden haku .....	3
2.1.6. Käyttäjien lisääminen.....	3
2.1.7. Käyttäjien poistaminen .....	3
2.1.8. Käyttäjien muokkaaminen .....	3
2.1.9. Käyttäjien haku .....	3
2.1.10. Lainaaminen.....	4
2.1.11. Raportit .....	4
2.1.12. Hakuprofiilit.....	4
2.2. Rajoitteet .....	4
<b>3. Use Caset</b> .....	<b>5</b>
3.1. Käyttäjäprofiilit.....	5
3.2. Sisäänkirjautuminen.....	5
3.3. Poistuminen.....	5
3.4. Haku .....	5
3.5. Hakuprofiilin lisääminen käyttäjälle.....	5
3.6. Hakuprofiilin hakeminen .....	5
3.7. Lainaaminen.....	5
3.8. Nimikkeiden tallettaminen.....	5
3.9. Nimekkeiden poistaminen ja tietojen editointi .....	5
3.10. Raporttien luominen.....	5
<b>4. Käsitemalli</b> .....	<b>6</b>
4.1. ER-kaavio .....	6
4.2. Tietokantakaava .....	6
4.2.1. Toimistorakennus.....	6
4.2.2. Postinumero .....	7
4.2.3. Hakuprofiili.....	7
4.2.4. Huone .....	7
4.2.5. Henkilo.....	8
4.2.6. Puhelinnumero .....	8
4.2.7. Nimike.....	9
4.2.8. Avainsana.....	9
4.2.9. Kirja .....	9
4.2.10. Lehti .....	10
4.3. Tietokannan elementtien luonti- ja muokkauslauseet.....	10
4.3.1. Avainsana.....	10
4.3.2. Hakuprofiili.....	10
4.3.3. Henkilo.....	11
4.3.4. Huone .....	11

---

4.3.5.	Nimike.....	12
4.3.6.	Postinumero .....	12
4.3.7.	Puhelinnumero .....	12
4.3.8.	Toimistorakennus.....	13
4.3.9.	Kirja .....	13
4.3.10.	Lehti .....	13
4.3.11.	Indeksimalleja .....	13
4.3.12.	Testitiedon syöttölauseet.....	14
4.3.13.	Testitiedon päivittäminen ja poistaminen .....	15
4.3.14.	Näkymiä .....	15
4.3.15.	Käyttäjät .....	15
4.3.16.	Oikeudet .....	16
4.4.	PL/SQL prosedureja ja funktiot .....	17
4.4.1.	Proseduurit .....	17
4.4.2.	Funktiot .....	19

## 1. Yleiskuvaus

Kirjasto on järjestelmä, johon talletetaan tietoa kirjoista ja lehdistä. Järjestelmän on tarkoitus helpottaa kirjojen löytämistä ja saatavuutta normaaliin kirjastojärjestelmään verrattuna. Kirjat voivat sijaita missä tahansa yrityksen eri toimipisteissä kirjastotilassa tai työntekijöiden toimitiloissa ja järjestelmän avulla ne voidaan paikantaa, lainata tai varata.

## 2. Vaatimukset

### 2.1. *Toiminnalliset*

#### 2.1.1. Sisäänkirjautuminen

Tietokannassa olevien käyttäjän tunnuksen ja salasanan on vastattava käyttäjän itse syöttämiä arvoja. Pakollinen.

#### 2.1.2. Nimikkeiden lisääminen

Tietokantaan on pystyttävä lisäämään uusia nimikkeitä. Pakollinen.

#### 2.1.3. Nimikkeiden poistaminen

Tietokannasta on pystyttävä poistamaan nimikkeitä. Pakollinen.

#### 2.1.4. Nimikkeiden muokkaaminen

Tietokannassa olevia nimikkeitä on pystyttävä muokata. Pakollinen.

#### 2.1.5. Nimikkeiden haku

Tietokannasta pitää pystyä hakemaan nimikkeitä kaikkien tietokenttien perusteella. Pakollinen.

#### 2.1.6. Käyttäjien lisääminen

Tietokantaan on pystyttävä lisäämään käyttäjiä. Pakollinen.

#### 2.1.7. Käyttäjien poistaminen

Tietokannasta on pystyttävä poistamaan nimikkeitä. Pakollinen.

#### 2.1.8. Käyttäjien muokkaaminen

Tietokannassa olevia nimikkeitä on pystyttävä muokata. Pakollinen.

#### 2.1.9. Käyttäjien haku

Tietokannasta pitää pystyä hakemaan nimikkeitä kaikkien tietokenttien perusteella. Pakollinen.

### **2.1.10. Lainaaminen**

Tietokantaan on pystyttävä tallentamaan tiedot lainauksista. Pakollinen.

### **2.1.11. Raportit**

Tietokannasta voi muodostaa erilaisia raportteja, kuten esimerkiksi tietoja käyttäjän lainoista. Kirjastonhoitaja voi katsoa myöskin статистиikkatietoja kirjoista. Valinnainen.

### **2.1.12. Hakuprofiilit**

Tietokantaan on pystyttävä tallentamaan käyttäjän määräämiä hakukriteereitä ja hakea profiilitiedot käytettäväksi. Valinnainen.

## **2.2. Rajoitteet**

Järjestelmässä ei ole havaittuja rajoitteita.

### 3. Use Caset

#### 3.1. Käyttäjäprofiilit

Työntekijä	Kuka tahansa järjestelmän käyttäjä
Kirjastonhoitaja	Kirjaston ylläpitäjä

#### 3.2. Sisäänkirjautuminen

#### 3.3. Poistuminen

#### 3.4. Haku

Tietokannasta haetaan nimikkeitä käyttäjän antamilla ehdoilla.

#### 3.5. Hakuprofiilin lisääminen käyttäjälle

#### 3.6. Hakuprofiilin hakeminen

#### 3.7. Lainaaminen

Tietokannasta tarkistetaan, voiko käyttäjä lainata nimikettä l. onko hän lainauskiellossa. Lisäksi tarkistetaan nimikkeen tila l. onko nimike varattu tai kadoksissa järjestelmän mielestä. Jos kirja on lainattavissa, tietokannan lisätään tieto lainauksesta kirjan id-tunnuksella käyttäjän id-tunnukselle.

#### 3.8. Nimikkeiden tallettaminen

#### 3.9. Nimekkeiden poistaminen ja tietojen editointi

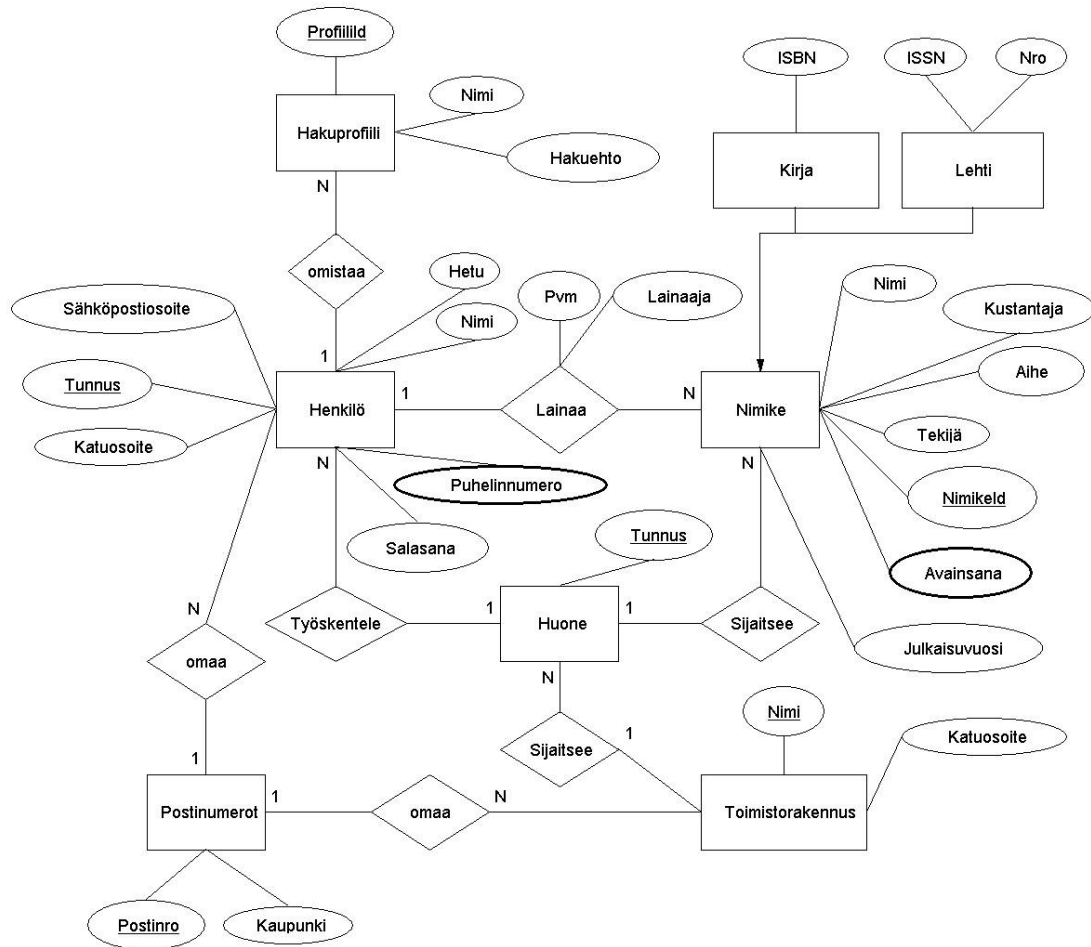
Tietokannasta haetaan poistettava tai muokattava nimike, jolle muokattaessa asetetaan uudet arvot. Poistettaessa hävitetään kaikki nimikkeeseen liittyvät tiedot tietokannasta. Jos teos on merkitty lainatuksi, niin laina poistetaan. Nimike voidaan poistaa, jos fyysinen teos on huonokuntoinen tai kadonnut kokonaan.

#### 3.10. Raporttien luominen

Tietokannasta haetaan esim. kaikki lainassa olevat nimikkeet ja järjestetään ne nimikkeen lainauspäivän mukaan.

## 4. Käsitelmä

### 4.1. ER-kaavio



### 4.2. Tietokantakaava

#### 4.2.1. Toimistorakennus

Sarake	Kuvaus	Null	Tyyppi	Mahdolliset arvot
<u>Nimi</u>	Toimistorakennuksen nimi	NOT NULL	VARCHAR2(30)	
Katuosoite	Toimistorakennuksen katuosoite	NOT NULL	VARCHAR2(100)	
Postinro	Toimistorakennuksen postinumero	NOT NULL	VARCHAR2(5)	FK: Postinumero(Postinro)

#### 4.2.2. Postinumero

Sarake	Kuvaus	Null	Tyyppi	Mahdolliset arvot
<u>Postinro</u>	Postinumero	NOT NULL	VARCHAR2(5)	numeroita
Kaupunki	Postitoimipaikka	NOT NULL	VARCHAR2(50)	

#### 4.2.3. Hakuprofiili

Sarake	Kuvaus	Null	Tyyppi	Mahdolliset arvot
<u>ProfiiliID</u>	Hakuprofiilin id	NOT NULL	NUMBER	
Nimi	Hakuprofiilin nimi	NOT NULL	VARCHAR2(20)	
Hakuehto	Hakuprofiilin hakuehdot	NOT NULL	VARCHAR2(100)	
Henkilo	Hakuprofiilin tekijän tunnus	NOT NULL	NUMBER	FK: Henkilo(Tunnus)

#### 4.2.4. Huone

Sarake	Kuvaus	Null	Tyyppi	Mahdolliset arvot
<u>HuoneenTunnus</u>	Huoneen tunnus	NOT NULL	VARCHAR2(20)	
<u>Toimistorakennus</u>	Toimistorakennus, jossa huone sijaitsee	NOT NULL	VARCHAR2(30)	FK: Toimistorakennus(Nimi)



#### 4.2.5. Henkilo

Sarake	Kuvaus	Null	Tyyppi	Mahdolliset arvot
<u>Tunnus</u>	Henkilön yksilöivä tunnus	NOT NULL	NUMBER	
Salasana	Henkilön salasana	NOT NULL	VARCHAR2(12)	
Hetu	Henkilön sosiaaliturvatunnus	NOT NULL	VARCHAR2(11)	6 numeroa, +,- tai A, 3 numeroa, numero tai kirjain
Nimi	Henkilön nimi	NOT NULL	VARCHAR2(100)	
Katuosoite	Henkilön katuosoite	NOT NULL	VARCHAR2(100)	
Postinro	Henkilön postinumero	NOT NULL	VARCHAR2(5)	FK: Postinumero(Postinro)
Huone	Henkilön työhuoneen tunnus	NOT NULL	VARCHAR2(20)	FK: Huone(HuoneenTunnus)

#### 4.2.6. Puhelinnumero

Sarake	Kuvaus	Null	Tyyppi	Mahdolliset arvot
<u>Henkilo</u>	Henkilön tunnus	NOT NULL	NUMBER	FK. Henkilo(Tunnus)
<u>Numero</u>	Henkilön puhelinnumero	NOT NULL	VARCHAR2(20)	

#### 4.2.7. Nimike

Sarake	Kuvaus	Null	Tyyppi	Mahdolliset arvot
<u>NimikeID</u>	Nimikkeen id	NOT NULL	NUMBER	
Nimi	Nimikkeen nimi	NOT NULL	VARCHAR2(100)	
Tekija	Nimikkeen tekijä	NOT NULL	VARCHAR2(100)	
Aihe	Nimikkeen aihe		VARCHAR2(20)	
Kustantaja	Nimikkeen kustantaja	NOT NULL	VARCHAR2(40)	
Julkaisuvuosi	Nimikkeen julkaisuvuosi	NOT NULL	NUMBER	>0
Huone	Huone, jossa nimike sijaitsee	NOT NULL	VARCHAR2(20)	FK : Huone(HuoneenTunnus)
Lainauspvm	Lainauspäivämäärä		DATE	
Lainaja	Nimikkeen lainaajan tunnus		NUMBER	FK: Henkilo(Tunnus)

#### 4.2.8. Avainsana

Sarake	Kuvaus	Null	Tyyppi	Mahdolliset arvot
<u>Nimike</u>	Nimikkeen id	NOT NULL	NUMBER	FK: Nimike(NimikeID)
<u>Avainsana</u>	Nimikkeen avainsana	NOT NULL	VARCHAR2(20)	

#### 4.2.9. Kirja

Sarake	Kuvaus	Null	Tyyppi	Mahdolliset arvot
<u>Nimike</u>	Nimikkeen id	NOT NULL	NUMBER	FK: Nimike(NimikeID)
ISBN	Kirjan tunnistenumero	NOT NULL	VARCHAR2(30)	

#### 4.2.10. Lehti

Sarake	Kuvaus	Null	Tyyppi	Mahdolliset arvot
<u>Nimike</u>	Nimikkeen id	NOT NULL	NUMBER	FK: Nimike(NimikeID)
ISSN	Lehden tunnistenumero	NOT NULL	VARCHAR2(30)	
Numero	Lehden numero	NOT NULL	VARCHAR2(10)	

### 4.3. Tietokannan elementtien luonti- ja muokkauslauseet

#### 4.3.1. Avainsana

```
CREATE TABLE Avainsana (  
  Nimike      NUMBER NOT NULL,  
  Avainsana  VARCHAR2(20) NOT NULL,  
  CONSTRAINT Avainsana_PK  
    PRIMARY KEY (Nimike,Avainsana),  
  CONSTRAINT Avainsana_FK_Nimike  
    FOREIGN KEY (Nimike)  
      REFERENCES Nimike (NimikeID)  
);
```

#### 4.3.2. Hakuprofiili

```
CREATE TABLE Hakuprofiili (  
  ProfiiliID NUMBER NOT NULL,  
  Nimi       VARCHAR2(20) NOT NULL,  
  Hakuehto  VARCHAR2(500) NOT NULL,  
  Henkilo   NUMBER NOT NULL,  
  CONSTRAINT PK_Hakuprofiili  
    PRIMARY KEY (ProfiiliID),  
  CONSTRAINT Hakuprofiili_FK_Henkilo  
    FOREIGN KEY (Henkilo)  
      REFERENCES Henkilo (Tunnus)  
);
```

#### Pseudosarakkeen luonti

```
CREATE SEQUENCE seq_Profiili;
```

### 4.3.3. Henkilo

```
CREATE TABLE Henkilo (  
  Tunnus      NUMBER NOT NULL,  
  Salasana    VARCHAR2(12) NOT NULL,  
  Hetu        VARCHAR2(11) NOT NULL,  
  Nimi        VARCHAR2(100) NOT NULL,  
  Katuosoite  VARCHAR2(100) NOT NULL,  
  Postinro    VARCHAR2(5) NOT NULL,  
  Huone       VARCHAR2(20) NOT NULL,  
  CONSTRAINT Henkilo_PK  
    PRIMARY KEY (Tunnus),  
  CONSTRAINT Henkilo_FK_Postinnumero  
    FOREIGN KEY (Postinro)  
    REFERENCES Postinnumero (Postinro),  
  CONSTRAINT Henkilo_FK_Huone  
    FOREIGN KEY (Huone)  
    REFERENCES Huone (HuoneenTunnus)  
);
```

### 4.3.4. Huone

```
CREATE TABLE Huone (  
  HuoneenTunnus  VARCHAR2(20) NOT NULL,  
  Toimistorakennus  VARCHAR2(30) NOT NULL,  
  CONSTRAINT Huone_PK  
    PRIMARY KEY (HuoneenTunnus),  
  CONSTRAINT Toimistorakennus_FK_Huone  
    FOREIGN KEY (Toimistorakennus)  
    REFERENCES Toimistorakennus (Nimi)  
);
```

### 4.3.5. Nimike

```
CREATE TABLE Nimike (  
    NimikeID    NUMBER    NOT NULL,  
    Nimi        VARCHAR2(100) NOT NULL,  
    Tekija      VARCHAR2(100) NOT NULL,  
    Aihe        VARCHAR2(20),  
    Kustantaja  VARCHAR2(40) NOT NULL,  
    Julkaisuvuosi NUMBER NOT NULL,  
    Huone       VARCHAR2(20) NOT NULL,  
    Lainauspvm DATE,  
    Lainaja     NUMBER,  
CONSTRAINT Nimike_PK  
    PRIMARY KEY (NimikeID),  
CONSTRAINT Nimike_FK_Huone  
    FOREIGN KEY (Huone)  
    REFERENCES Huone (HuoneenTunnus),  
CONSTRAINT Nimike_FK_Henkilo  
    FOREIGN KEY (Lainaja)  
    REFERENCES Henkilo (Tunnus),  
CONSTRAINT Jvuosi_Check CHECK (Julkaisuvuosi > 0)  
);
```

#### Pseudosarakkeen luonti

```
CREATE SEQUENCE seq_Nimike;
```

### 4.3.6. Postinumero

```
CREATE TABLE Postinumero (  
    Postinro  VARCHAR2(5) NOT NULL,  
    Kaupunki  VARCHAR(50) NOT NULL,  
CONSTRAINT Postinumero_PK  
    PRIMARY KEY (Postinro)  
);
```

### 4.3.7. Puhelinnumero

```
CREATE TABLE Puhelinnumero (  
    Henkilo    NUMBER NOT NULL,  
    Numero     VARCHAR2(20) NOT NULL,  
CONSTRAINT Puhelinnumero_PK  
    PRIMARY KEY (Henkilo,Numero),  
CONSTRAINT Puhelinnumero_FK_Henkilo  
    FOREIGN KEY (Henkilo)  
    REFERENCES Henkilo (Tunnus)  
);
```

#### 4.3.8. Toimistorakennus

```
CREATE TABLE Toimistorakennus (  
  Nimi          VARCHAR2(30) NOT NULL,  
  Katuosoite    VARCHAR2(100) NOT NULL,  
  Postinro      VARCHAR2(5) NOT NULL,  
  CONSTRAINT Toimistorakennus_PK  
    PRIMARY KEY (Nimi),  
  CONSTRAINT Toimistorakennus_FK_Postinro  
    FOREIGN KEY (Postinro)  
      REFERENCES Postinnumero(Postinro)  
);
```

#### 4.3.9. Kirja

```
CREATE TABLE Kirja (  
  Nimike  NUMBER NOT NULL,  
  ISBN    VARCHAR2(30) NOT NULL,  
  CONSTRAINT Kirja_PK  
    PRIMARY KEY (Nimike),  
  CONSTRAINT Kirja_FK_Nimike  
    FOREIGN KEY (Nimike)  
      REFERENCES Nimike (NimikeID)  
);
```

#### 4.3.10. Lehti

```
CREATE TABLE Lehti (  
  Nimike  NUMBER NOT NULL,  
  ISSN    VARCHAR2(30) NOT NULL,  
  Numero  VARCHAR(10) NOT NULL,  
  CONSTRAINT Lehti_PK  
    PRIMARY KEY (Nimike),  
  CONSTRAINT Lehti_FK_Nimike  
    FOREIGN KEY (Nimike)  
      REFERENCES Nimike (NimikeID)  
);
```

#### 4.3.11. Indeksimalleja

```
CREATE INDEX Henkilo_Nimi_Idx on Henkilo(Nimi);  
CREATE INDEX Nimike_Tekija_Idx on Nimike(Tekija, Nimi);  
CREATE INDEX Nimike_Lainaus_Idx on Nimike(Lainauspvm, Tekija, Nimi);
```

### 4.3.12. Testitiedon syöttölauseet

```
INSERT INTO Henkilo (Tunnus, Salasana, Hetu, Nimi, Katuosoite,  
Postinro, Huone)  
VALUES (1122,'IsoKissa', '010165-199L', 'Aku Ankka', 'Hanhitie 5',  
'10100', 'AR200');
```

```
INSERT INTO Henkilo (Tunnus, Salasana, Hetu, Nimi, Katuosoite,  
Postinro, Huone)  
VALUES (0,'AdMi', '050567-199L', 'Kirjastonhoitaja', 'Hanhitie 7',  
'10100', 'HH12.5');
```

```
INSERT INTO Puhelinnumero (Henkilo, Numero)  
VALUES (1122,'02-223344');
```

#### Seuraavat kolme lausetta liittyvät toisiinsa:

```
INSERT INTO Nimike (NimikeID, Nimi, Tekija, Aihe, Kustantaja,  
Julkaisuvuosi, Huone)  
VALUES (seq_Nimike.nextval,'Kissan seikkailut', 'Herra Huu',  
'kissatarina', 'WSOY', 1997, 'HH12.5');
```

```
INSERT INTO Kirja (Nimike, ISBN)  
VALUES (seq_Nimike.CurrVal,'00-1-947502');
```

```
INSERT INTO Avainsana (Nimike, Avainsana)  
VALUES (seq_Nimike.CurrVal,'kissa');
```

```
INSERT INTO Postinumero (Postinro, Kaupunki)  
VALUES (10100,'Helsinki');
```

```
INSERT INTO Toimistorakennus (Nimi ,Katuosoite, Postinro)  
VALUES ('Rakennus 1','Mannerheimintie 1',10100);
```

```
INSERT INTO Huone (HuoneenTunnus ,Toimistorakennus)  
VALUES ('AR200', 'Rakennus 1');
```

```
INSERT INTO Hakuprofiili (ProfiiliId ,Nimi,Hakuehto,Henkilo)  
VALUES (seq_Profiili.NextVal,'Profiili 1','Select * from Henkilo',0);
```

### 4.3.13. Testitiedon päivittäminen ja poistaminen

#### Lainataan nimike.

```
UPDATE Nimike
SET Lainauspvm = to_date(sysdate, 'DD.MM.YYYY'), Lainaaja = 1122,
Huone = (
    SELECT Huone
    FROM Henkilo
    WHERE Tunnus = 1122 )
WHERE NimikeID = &Nimike;
```

#### Seuraavat kolme lausetta liittyvät toisiinsa:

```
DELETE FROM Kirja
WHERE Nimike = &Nimike;
```

```
DELETE FROM Avainsana
WHERE Nimike = &&Nimike;
```

```
DELETE FROM Nimike
WHERE NimikeID = &&Nimike;
```

### 4.3.14. Näkymiä

#### Näkymä kertoo mitä lainoja kyseisellä käyttäjällä on.

```
CREATE OR REPLACE VIEW HenkilonLainat AS
SELECT Nimi
FROM Nimike
WHERE Lainaaja = 1122;
```

#### Näkymä kertoo kaikkien nimikkeiden lainaustietoja.

```
CREATE OR REPLACE VIEW NimikkeenTiedot
(NimikkeenNimi, Linauspvm, HenkiloNimi, HuoneNimi, Rakennus) AS
SELECT N.Nimi, N.Linauspvm, H.Nimi, N.Huone, Hu.Toimistorakennus
FROM Nimike N, Henkilo H, Huone Hu
WHERE N.Lainaaja = H.Tunnus AND N.Huone = Hu.HuoneenTunnus;
```

### 4.3.15. Käyttäjät

```
CREATE USER '1122' IDENTIFIED BY 'IsoKissa';
CREATE USER '0' IDENTIFIED BY 'AdMi';
```



### 4.3.16. Oikeudet

Tavallisen käyttäjän rooli:

```
CREATE ROLE kayttaja;
```

Esimerkki tavallinen käyttäjä liitetään rooliin seuraavasti:

```
GRANT kayttaja TO 1122;
```

Kirjastonhoitajan rooli:

```
CREATE ROLE kirjastonhoitaja;  
GRANT kirjastonhoitaja TO 0;
```

Oikeuksien jakaminen:

```
GRANT SELECT on USER_TABLES to public;  
REVOKE SELECT Salasana on Henkilo to public;  
GRANT EXECUTE on USER_TABLES to public;
```

```
GRANT INSERT on USER_TABLES to kirjastonhoitaja;  
GRANT UPDATE on USER_TABLES to kirjastonhoitaja;  
GRANT DELETE on USER_TABLES to kirjastonhoitaja;
```

(Edellä USER\_TABLES tarkoittaa kaikkia kirjastojärjestelmän tauluja.)

## 4.4. PL/SQL proseduurit ja funktiot

### 4.4.1. Proseduurit

```
-- *****  
-- Kirjastojärjestelmä  
-- Henkilötietojen haku henkilön tunnuksen  
-- perusteella.  
-- *****  
  
CREATE OR REPLACE PROCEDURE HaeHenkilo(  
    nTunnus IN Henkilo.Tunnus%TYPE,  
    vNimi OUT Henkilo.Nimi%TYPE,  
    vHuone OUT Henkilo.Huone%TYPE)  
  
IS  
    CURSOR CsrHenkTie IS  
        SELECT Nimi, Huone  
        FROM Henkilo  
        WHERE Tunnus = nTunnus;  
  
BEGIN  
    OPEN CsrHenkTie;  
    FETCH CsrHenkTie INTO vNimi, vHuone;  
    CLOSE CsrHenkTie;  
  
EXCEPTION  
    WHEN NO_DATA_FOUND THEN  
        DBMS_OUTPUT.PUT_LINE('Henkilötietoja ei löydy!');  
  
END;
```

Proseduuria voi testata seuraavalla PL/SQL-lausekkeella:

```
declare  
    vHuone Henkilo.Huone%TYPE;  
    vNimi Henkilo.Nimi%TYPE;  
begin  
    HaeHenkilo(1122, vNimi, vHuone);  
    DBMS_OUTPUT.PUT_LINE(vNimi || vHuone);  
end;  
/
```

```
-- *****  
-- Kirjastojärjestelmä  
-- Nimikkeen poistaminen.  
--*****
```

```
CREATE OR REPLACE PROCEDURE PoistaNimike(  
    nNimikeID IN Nimike.NimikeID%TYPE)
```

```
IS
```

```
    CURSOR CsrNimTie IS  
        SELECT *  
        FROM Nimike  
        WHERE NimikeID = nNimikeID;  
    NimRek CsrNimTie%ROWTYPE;
```

```
BEGIN
```

```
    OPEN CsrNimTie;  
    FETCH CsrNimTie INTO NimRek;  
    CLOSE CsrNimTie;
```

```
    IF Nimrek.Lainaaaja > 0 THEN
```

```
        UPDATE Nimike SET Lainauspvm = NULL, Lainaaaja = NULL
```

```
WHERE NimikeID = nNimikeID;
```

```
    END IF;
```

```
        DELETE FROM Avainsana WHERE Nimike = nNimikeID;
```

```
        DELETE FROM Kirja WHERE Nimike = nNimikeID;
```

```
        DELETE FROM Lehti WHERE Nimike = nNimikeID;
```

```
        DELETE FROM Nimike WHERE NimikeID = nNimikeID;
```

```
EXCEPTION
```

```
    WHEN NO_DATA_FOUND THEN
```

```
        DBMS_OUTPUT.PUT_LINE('Nimiketietoja ei löydy!');
```

```
END;
```

**Proseduuria voi testata seuraavalla PL/SQL-lausekkeella:**

```
declare  
begin  
    PoistaNimike(2);  
end;  
/
```

## 4.4.2. Funktiot

```
-- *****  
-- Kirjastojärjestelmä  
-- Palautetaan tieto, onko parametrinä tullut  
-- nimike saatavilla.  
-- *****  
  
CREATE OR REPLACE FUNCTION nSaatavilla(  
    nNimikeID IN OUT Nimike.NimikeID%TYPE) RETURN NUMBER  
  
IS  
    nTunnus NUMBER;  
    CURSOR CsrNimLai IS  
        SELECT Lainaaja, NimikeID  
        FROM Nimike  
        WHERE NimikeID = nNimikeID;  
  
BEGIN  
    OPEN CsrNimLai;  
    FETCH CsrNimLai INTO nTunnus, nNimikeID;  
  
    IF nTunnus IS NULL AND nNimikeID IS NULL THEN  
        nTunnus := -2;  
        -- Nimikettä ei löydy.  
    ELSIF nTunnus IS NULL THEN  
        nTunnus := -1;  
        -- Nimike ei ole lainassa.  
    END IF;  
    CLOSE CsrNimLai;  
  
    RETURN nTunnus;  
END;
```

Funktiota voi testata seuraavalla PL/SQL-lausekkeella:

```
declare  
tunnus Nimike.NimikeId%TYPE := &id;  
begin  
    DBMS_OUTPUT.PUT_LINE(nSaatavilla(tunnus));  
end;  
/
```

```
-- *****  
-- Kirjastojärjestelmä  
-- Palautetaan hakuehdot.  
-- *****  
  
CREATE OR REPLACE FUNCTION vHaeHakuehdot (  
    vNimi IN Hakuprofiili.Nimi%TYPE,  
    nTunnus IN Henkilo.Tunnus%TYPE) RETURN VARCHAR2  
  
IS  
    vHakuehdot Hakuprofiili.Hakuehto%TYPE;  
    CURSOR CsrHakuE IS  
        SELECT Hakuehto  
        FROM Hakuprofiili  
        WHERE Nimi = vNimi AND Henkilo = nTunnus;  
  
BEGIN  
    OPEN CsrHakuE;  
    FETCH CsrHakuE INTO vHakuehdot;  
    CLOSE CsrHakuE;  
  
    RETURN vHakuehdot;  
  
EXCEPTION  
    WHEN NO_DATA_FOUND THEN  
        DBMS_OUTPUT.PUT_LINE('Antamillasi ehdoilla ei löydy  
hakuehtoa!');  
END;
```

**Funktiota voi testata seuraavalla PL/SQL-lausekkeella:**

```
declare  
vNimi Hakuprofiili.Nimi%TYPE := '&hakuehto';  
nTunnus Henkilo.Tunnus%TYPE := &tunnus;  
begin  
    DBMS_OUTPUT.PUT_LINE(vHaeHakuehdot (vNimi, nTunnus));  
end;  
/
```