

TEHTÄVÄNANTO:

Syötä jokaiseen kirjastotietokannan tauluun muutama tietue:

- 1. Ainakin kaksi samaa kirjaa (pitää tallentua eri pääavaimille)
- 2. Kirja, jossa on useampi kuin yksi tekijä.
- 3. Lainaustapahtuma, jossa asiakas on lainannut kirjan ja se on vielä lainassa.

1 Tässä on kolme kpl samaa kirjaa kirjastossa (idTeos = 1 eli ”Insinöörin Matematiikka 2”):

```
mysql> select * from kirja;
```

idKirja	idTeos	hyllysijainti	hyllysaldo
1	1	MAT6	1
2	1	MAT6	1
3	1	MAT6	1
4	3	MAT6	1

4 rows in set (0.00 sec)

```
mysql> select * from teos;
```

idTeos	teos_nimi	julkaisuvuosi	painopaikka	versio
1	Insinöörin Matematiikka 2	2014-04-01	Porvoo	2.Painos
3	Tekniikan Kaavasto	2005-09-01	Tampere	25.Painos

2 rows in set (0.00 sec)

2 Kirjassa idTeos 1 eli ”Insinöörin Matematiikka 2” on yhteensä viisi eri tekijää:

```
mysql> select tekija.etunimi as Etunimi, tekija.sukunimi as Sukunimi, teos.teos_nimi as Teos
-> from tekija
-> INNER JOIN teoksen_tekija ON tekija.idTekija = teoksen_tekija.idTekija
-> INNER JOIN teos ON teos.idTeos = teoksen_tekija.idTeos;
```

Etunimi	Sukunimi	Teos
Jarkko	Surakka	Insinöörin Matematiikka 2
Heikki	Makkonen	Insinöörin Matematiikka 2
Maija	Huuskonen	Insinöörin Matematiikka 2
Eero	Holmlund	Insinöörin Matematiikka 2
Ari	Tuomenlehto	Insinöörin Matematiikka 2
Marko	Mäkelä	Tekniikan Kaavasto

6 rows in set (0.00 sec)

Viiden eri tekijän jakautuminen teoksiin. Yhdellä kirjalla on viisi tekijää ja toisella yksi:

```
mysql> select * from teoksen_tekija;
```

idTeos	idTekija
1	1
1	2
1	3
1	4
1	5
3	6

3 Lainaustapahtuma: Asiakas ”Christian Horner” on varannut ”Tekniikan kaavasto” kirjan kahdeksi viikoksi. Asiakas ”Mattia Binotto” haluaa varata kyseisen kirjan heti itselleen Christian Hornerin jälkeen.

Tulostetaan kaikki asiakkaat:

```
mysql> select * from asiakas;
```

idAsiakas	etunimi	sukunimi
1	Christian	Horner
2	Mattia	Binotto
3	Toto	Wolff

```
3 rows in set (0.00 sec)
```

Christianin lainaus (huom. *lainaus_status* = 1 tarkoittaa aktiivista lainausta):

```
mysql> INSERT INTO lainaus VALUES(NULL,1,4,1,CURDATE(),DATE_ADD(CURDATE(),INTERVAL 14 DAY));
Query OK, 1 row affected (0.01 sec)
```

```
mysql> select * from lainaus;
```

idLainaus	idAsiakas	idKirja	lainaus_status	lainaus_pvm	palautus_pvm
1	1	4	1	2025-02-03	2025-02-17

```
1 row in set (0.00 sec)
```

Lisätään Mattian varaus:

```
mysql> INSERT INTO lainaus VALUES(NULL,2,4,0,DATE_ADD(CURDATE(),INTERVAL 14 DAY),DATE_ADD(CURDATE(),INTERVAL 28 DAY));
Query OK, 1 row affected (0.01 sec)
```

```
mysql> select * from lainaus;
```

idLainaus	idAsiakas	idKirja	lainaus_status	lainaus_pvm	palautus_pvm
1	1	4	1	2025-02-03	2025-02-17
2	2	4	0	2025-02-17	2025-03-03

```
2 rows in set (0.00 sec)
```

Niin kuin ma 3.2.2025 luennolla totesimme, tietokannastamme puuttuu vielä kokonaan esim. lainaustapahtuman aliohjelma, joka tarkistaa lainauksen statuksen ja päivittää uuden lainausajan edellisen lainaustapahtuman jälkeen automaattisesti.

Tässä tehtävässä tämä piti päivittää manuaalisesti, mutta se opetti enemmän tietokannan käytöstä. Sain heti ideoita siitä, miten voisin kehittää tietokantani taulukoita joustavampaan muotoon.