**KANTAAN VIETÄVIEN RIVIEN LISÄÄMINEN TAULUUN:**

Vaihtoehdossa 1 tuloksia vain lisätään kantaan loputtomasti. On huomattava, että User Stories ”Loppukäyttäjänä haluan, että näen, miten pärjäsin muita pelaajia vastaan (pisteet)” ei voi toimia, jos kaikki tulokset tai ainakin kaikkien parhaat tulokset eivät ole tieto­kannassa. Toki toiminto voidaan nimetä kehityskohteeksi…

SQL-komento on seuraava:

INSERT INTO *table\_name* VALUES (*value1*,*value2*,*value3*, ...);

Esimerkki:

INSERT INTO TilitVALUES (‘Alf’, ‘salasana’, <avatar-linkki>, ...);

Tämä tehdään vain kerran tulosriviä kohden (ja mahdollisesti vain pelaajan pyynnöstä: ”Tee painike, josta käyttäjä voi halutessaan lisätä tuloksensa parhaat pelaajat -listaan”).

**TAULUN RIVIEN PÄIVITTÄMINEN TIETOKANNASSA:**

Vaihtoehdoissa 2 ja 3 tuloksia päivitetään kantaan vain 11 TOP10-riville. Näin tieto­kanta ei kasva rajatta.

SQL-komento on seuraava:

UPDATE table\_name  
SET column1 = value1, column2 = value2, ...  
WHERE condition;

Esimerkki (11 riviä):

UPDATE Tilit SET pnimi = 'Alf', pisteet = '21', … WHERE rivi\_id = 1;

UPDATE Tilit SET pnimi = 'Ben', pisteet = '22', … WHERE rivi\_id = 2;

UPDATE Tilit SET pnimi = 'Carl', pisteet = '22', … WHERE rivi\_id = 3;

UPDATE Tilit SET pnimi = 'Dick', pisteet = '24', … WHERE rivi\_id = 4;

UPDATE Tilit SET pnimi = 'Elf', pisteet = '27', … WHERE rivi\_id = 5;

jne.

**KANNASTA HAETTAESSA VAIHTOEHTOJEN EROT EIVÄT OLE SUURET:**

Palautettavat tiedot voidaan melko helposti muotoilla vaikkapa CSV-taulukoksi (pilkulla erotetuksi luetteloksi) esimerkiksi seuraavasti:

Alf, 21, 60

Ben, 22, 55

Carl, 22, 75

jne.

Työ onkin sitten edessä vaihtoehdoissa 2 ja 3, kun 10 parasta riviä pitää tästä muodosta palauttaa SQL-komennon formaattiin rivien päivittämistä varten. Tätä ei tarvitse tehdä vaihtoehdossa 1, joten **kannatan vahvasti vaihtoehtoa 1!**