

L07 Tehtävät

- Rakenteiset tietotyypit lista ja luokka/olio
- Listan operaatiot ja niiden käyttäminen
- Merkkijonon jako osiin split-jäsenfunktioilla
- Laajeneva valikkopohjainen ohjelma (T3)
- Olioiden lisääminen listan alkioiksi (oliolista)

Lue oppaan tämän viikon asioita käsittelevä luku 7. Lisäksi tehtävien suorittamiseen tarvittavat aiempien lukujen tiedot.

Huomioita viikkotehtävistä

- Käytä paaohjelma()-rakennetta kaikissa tehtävissä L05 mukaisesti
- **HUOM!** Lisää jokaiseen palauttamaasi tehtävään **L07T1 -tehtävästä alkaen** alla oleva alkukommentti ja täytä siihen tehtäväkohtaiset tiedot. Kyseinen otsikkotiedot.py -tiedosto löytyy Moodlesta L07-viikkotehtävien tiedostojen yhteydestä ja asia on selitetty luennolla L07.

```
#####
# CT60A0203 Ohjelmoinnin perusteet
# Tekijä:
# Opiskelijanumero:
# Päivämäärä:
# Kurssin oppimateriaalien lisäksi työhön ovat vaikuttaneet seuraavat
# lähteet ja henkilöt, ja se näkyy tehtävässä seuraavalla tavalla:
#
# Mahdollisen vilppiselvityksen varalta vakuutan, että olen tehnyt itse
# tämän tehtävän ja vain yllä mainitut henkilöt sekä lähteet ovat
# vaikuttaneet siihen yllä mainituilla tavoilla.
#####
# Tehtävä LxTx.py
# eof
```

Sisällysluettelo

Huomioita viikkotehtävistä	1
L07T1: Tehtävälistan käsittely valikkopohjaisella ohjelmalla (M)	1
L07T2: Luokan määrittely ja olion käyttö (M)	5
L07T3: Kehittyvä valikkopohjainen ohjelma (lista, luokka, olio) (P)	6
L07T4: Tiedoston rivin jako tietoalkioihin (M)	8
L07T5: Valikkopohjainen ohjelma tuotetietojen hallintaan luokalla ja oliolistalla (P)	9

L07T1: Tehtävälistan käsittely valikkopohjaisella ohjelmalla (M)

Tehtävän taso: Minimitaso

Tee ohjelma, joka ylläpitää tehtävälistaa. Käyttäjä voi valita valikosta esimerkkiajon mukaisesti tehtävän lisäämisen, poistamisen tai ohjelman lopettamisen. Mikäli käyttäjä haluaa:

1. Lisätä tehtävän. Ohjelma pyytää tehtävän merkkijonona ja lisää sen listalle.
2. Poistaa tehtävän. Ohjelma pyytää poistettavan tehtävän järjestysnumeron ja poistaa sen listalta.
3. Tulostaa listan. Ohjelma tulostaa listan yksi alkio kerrallaan esimerkkiajon mukaisesti.
4. Lopettaa ohjelman. Jos tehtäviä jäi tekemättä, ohjelma kertoo käyttäjälle ennen lopetusta mitkä tehtävät jäivät tekemättä ja tulostaa listan tulostus-aliohjelmalla.

Toteuta tehtävälista Pythonin lista-rakenteella ja muista tyhjentää lista-rakenne ohjelman lopussa `clear`-komennolla. Ohjelmassa tulee olla pääohjelma ja omat aliohjelmat valikolle sekä valikon toiminnoille 1-3 kurssin tyyliohjeen mukaisesti. Mikäli käyttäjä haluaa poistaa tehtävän, jota ei ole listalla, tai valitsee olemattoman toiminnon, ohjelma kertoo käyttäjälle ongelmasta alla olevilla ilmoituksilla. "Tuntematon valinta, yritä uudestaan." "Listalta ei löydy tietoja järjestysnumerolla 99." Tämän virheilmoituksen lisäksi alla oleva ohje, jossa päättymisnumero on listan pituus. "Järjestysnumerot ovat väliltä 1-3."

Esimerkkiajo 1

Syötteet:

```
1
Käy kaupassa
1
Katso elokuva
3
2
1
2
1
0
```

Tuloste:

```
Valitse haluamasi toiminto:
1) Lisää tehtävä listaan
2) Poista tehtävä listalta
3) Tulosta tehtävälista
0) Lopeta
Valintasi: 1
Anna lisättävä tehtävä: Käy kaupassa

Valitse haluamasi toiminto:
1) Lisää tehtävä listaan
2) Poista tehtävä listalta
3) Tulosta tehtävälista
0) Lopeta
Valintasi: 1
Anna lisättävä tehtävä: Katso elokuva

Valitse haluamasi toiminto:
```

```
1) Lisää tehtävä listaan
2) Poista tehtävä listalta
3) Tulosta tehtävälista
0) Lopeta
Valintasi: 3
Tehtävälistassasi on seuraavat tehtävät:
1. Käy kaupassa
2. Katso elokuva

Valitse haluamasi toiminto:
1) Lisää tehtävä listaan
2) Poista tehtävä listalta
3) Tulosta tehtävälista
0) Lopeta
Valintasi: 2
Tehtävälistassasi on 2 tehtävää.
Anna poistettavan tehtävän järjestysnumero: 1

Valitse haluamasi toiminto:
1) Lisää tehtävä listaan
2) Poista tehtävä listalta
3) Tulosta tehtävälista
0) Lopeta
Valintasi: 2
Tehtävälistassasi on 1 tehtävää.
Anna poistettavan tehtävän järjestysnumero: 1

Valitse haluamasi toiminto:
1) Lisää tehtävä listaan
2) Poista tehtävä listalta
3) Tulosta tehtävälista
0) Lopeta
Valintasi: 0
Lopetetaan.

Kiitos ohjelman käytöstä.
```

Esimerkkiajo 2

Syötteet:

```
1
Lue ohjelmointioppaan luku 7
1
Lue tehtävänanto
1
Tee ohjelmointitehtävä
1
Tarkista ohjelma ASPA:lla
1
Palauta tehtävä
3
2
1
2
1
```

2
1
0

Tuloste:

Valitse haluamasi toiminto:

- 1) Lisää tehtävä listaan
- 2) Poista tehtävä listalta
- 3) Tulosta tehtävälista
- 0) Lopeta

Valintasi: 1

Anna lisättävä tehtävä: Lue ohjelmointioppaan luku 7

Valitse haluamasi toiminto:

- 1) Lisää tehtävä listaan
- 2) Poista tehtävä listalta
- 3) Tulosta tehtävälista
- 0) Lopeta

Valintasi: 1

Anna lisättävä tehtävä: Lue tehtävänanto

Valitse haluamasi toiminto:

- 1) Lisää tehtävä listaan
- 2) Poista tehtävä listalta
- 3) Tulosta tehtävälista
- 0) Lopeta

Valintasi: 1

Anna lisättävä tehtävä: Tee ohjelmointitehtävä

Valitse haluamasi toiminto:

- 1) Lisää tehtävä listaan
- 2) Poista tehtävä listalta
- 3) Tulosta tehtävälista
- 0) Lopeta

Valintasi: 1

Anna lisättävä tehtävä: Tarkista ohjelma ASPA:lla

Valitse haluamasi toiminto:

- 1) Lisää tehtävä listaan
- 2) Poista tehtävä listalta
- 3) Tulosta tehtävälista
- 0) Lopeta

Valintasi: 1

Anna lisättävä tehtävä: Palauta tehtävä

Valitse haluamasi toiminto:

- 1) Lisää tehtävä listaan
- 2) Poista tehtävä listalta
- 3) Tulosta tehtävälista
- 0) Lopeta

Valintasi: 3

Tehtävälistassasi on seuraavat tehtävät:

1. Lue ohjelmointioppaan luku 7
2. Lue tehtävänanto
3. Tee ohjelmointitehtävä
4. Tarkista ohjelma ASPA:lla
5. Palauta tehtävä

```
Valitse haluamasi toiminto:
1) Lisää tehtävä listaan
2) Poista tehtävä listalta
3) Tulosta tehtävälista
0) Lopeta
Valintasi: 2
Tehtävälistassasi on 5 tehtävää.
Anna poistettavan tehtävän järjestysnumero: 1
```

```
Valitse haluamasi toiminto:
1) Lisää tehtävä listaan
2) Poista tehtävä listalta
3) Tulosta tehtävälista
0) Lopeta
Valintasi: 2
Tehtävälistassasi on 4 tehtävää.
Anna poistettavan tehtävän järjestysnumero: 1
```

```
Valitse haluamasi toiminto:
1) Lisää tehtävä listaan
2) Poista tehtävä listalta
3) Tulosta tehtävälista
0) Lopeta
Valintasi: 2
Tehtävälistassasi on 3 tehtävää.
Anna poistettavan tehtävän järjestysnumero: 1
```

```
Valitse haluamasi toiminto:
1) Lisää tehtävä listaan
2) Poista tehtävä listalta
3) Tulosta tehtävälista
0) Lopeta
Valintasi: 0
Sinulta jäi tekemättä seuraavat tehtävät:
1. Tarkista ohjelma ASPA:lla
2. Palauta tehtävä
Lopetetaan.
```

```
Kiitos ohjelman käytöstä.
```

L07T2: Luokan määrittely ja olion käyttö (M)

Tehtävän taso: Minimitaso

Tee ohjelma, jonka alussa määritellään luokka ja demonstroi luokan käyttöä kysymällä tietoja sekä tulostamalla ne eri aliohjelmissa. Ohjelman perusrakenne on seuraava:

1. Määrittele luokka ELOKUVA, jossa on kolme jäsenmuuttujaa, Nimi, Pituus ja Ohjaaja.
2. Luo pääohjelmassa luokasta olio ja kutsu aliohjelmaa välittäen aliohjelmalle parametrina tekemäsi olio. Aliohjelma kysyy tietoja käyttäjältä, sijoittaa ne olion jäsenmuuttujiksi ja palauttaa olion paluuarvona pääohjelmaan.
3. Pääohjelmassa paluuarvo tulee sijoittaa muuttujaan, jotta sen voi välittää tulostus-aliohjelmaan. Tämä aliohjelma tulostaa olion sisältämien jäsenmuuttujien arvot.

Esimerkkiajo 1

Syötteet:

```
The Imitation Game
114
Morten Tyldum
```

Tuloste:

```
Anna elokuvan nimi: The Imitation Game
Anna elokuvan pituus kokonaisina minuutteina: 114
Anna elokuvan ohjaaja: Morten Tyldum
Elokuvan 'The Imitation Game' on ohjannut Morten Tyldum ja sen kesto on 114 minuuttia.
Kiitos ohjelman käytöstä.
```

L07T3: Kehittyvä valikkopohjainen ohjelma (lista, luokka, olio) (P)

Tehtävän taso: Perustaso

Tällä kertaa laajenevaan valikkopohjaiseen ohjelmaan lisätään lista, luokka ja olio. Aiemmin jouduimme lukemaan tiedoston samassa aliohjelmassa analysoinnin kanssa, mutta nyt lista-rakenteen avulla voimme lukea kaikki datatiedostossa olevat tiedot ensin listaan ja analysoida sitä toisessa aliohjelmassa. Siksi muutamme ensin ohjelman toimintaperiaatetta ja erotamme tiedoston lukemisen ja analysoinnin omiksi kohdiksi valikkoon. Tämä tekee ohjelmasta kokonaisuuden kannalta järkevämmän ja ylläpidettävämmän.

1. Muuta valikon ensimmäistä valintaa siten, että valinnassa kysytään setelin arvo ja tiedoston nimi. Tämän jälkeen luetaan datatiedosto. Toteuta jokainen toiminto omassa aliohjelmassaan. Tiedoston lukeva aliohjelma lukee tiedoston ja lisää kaikki numerot ja päivämäärät olioihin, siten että yhdellä rivillä oleva numero ja päivämäärä ovat yhdessä oliossa kumpikin erillisessä jäsenmuuttujassa. Lisää kunkin rivin olio listaan, eli tee oliolista. Lukemisen jälkeen lista palautetaan pääohjelmaan. Huomaa, että lista-rakenne tulee tyhjentää `clear()`-komennolla ennen tietojen lukemista sekä ohjelman päättyessä.
2. Analysoi-valinta saa parametrinä luetut tiedot sisältävän listan ja tulos-olion. Analyysi-aliohjelma käy listan läpi yhden kerran ja etsii samassa toistorakenteessa pienimmän ja suurimman arvon sekä laskee tarvittavat lukumäärä- ja summatiedot lukujen keskiarvon selvittämistä varten. Analyysin tulokset kirjoitetaan parametrinä saadun olion jäsenmuuttujiin ja palautetaan pääohjelmaan.
3. Tulostus-aliohjelma saa parametrina tulos-olion, setelin arvon ja tiedoston nimen. Tulokset kirjoitetaan tulostiolista tulostiedostoon.
4. Lisää luvun kysyvään aliohjelmaan tarkistus setelin oikeellisuuteen liittyen. Setelin arvon kysymisessä tulee tarkistaa onko annettu syöte luku ennen sen muuttamista kokonaisluvuksi. Tämä onnistuu esimerkiksi `isdigit()`-komennolla, josta on lisätietoa ohjelmointioppaassa. Tässä kohtaa emme vielä tarkista onko annettu arvo oikea seteli. Mikäli käyttäjä ei anna kokonaislukua, kysy sitä käyttäjältä uudestaan niin kauan, kunnes käyttäjä antaa hyväksyttävän arvon, eli kokonaisluvun.

Muuttunut ohjelmarakenne hyödyntää TULOS-luokkaa ja siitä pääohjelmassa luotavaa tulos-oliota, joka sisältää kaikki kiinnostavat tiedot, kuten keskiarvo, pienin ja suurin arvo. Kun analyysissä nämä

tiedot on sijoitettu tulos-olioon, voidaan aliohjelmasta palauttaa kaikki halutut tiedot tämän olion avulla yhtenä paluuarvona.

Moodlessa on datatiedostot L07T3D1.txt ja L07T3D2.txt sekä tulostiedostot L07T3T1.txt ja L07T3T2.txt, joilla ohjelmaa voi testata.

Esimerkki syötetiedostosta 'L07T3D1.txt':

```
Aikaleima;20€ setelien määrä
31.1.2002;1961761
28.2.2002;1925594
31.3.2002;1909862
30.4.2002;1827840
31.5.2002;1801616
```

Esimerkkiajo 1

Syötteet:

```
1
20
L07T3D1.txt
2
3
L07T3T1.txt
0
```

Tuloste:

Tällä ohjelmalla voit analysoida eurosetelien tietoja.

Valitse haluamasi toiminto:

- 1) Lue tiedosto
- 2) Analysoi
- 3) Kirjoita tiedosto
- 0) Lopeta

Anna valintasi: 1

Anna analysoitavan setelin arvo: 20

Anna luettavan tiedoston nimi: L07T3D1.txt

Tiedosto L07T3D1.txt luettu.

Valitse haluamasi toiminto:

- 1) Lue tiedosto
- 2) Analysoi
- 3) Kirjoita tiedosto
- 0) Lopeta

Anna valintasi: 2

Analyysi suoritettu, 5 alkiota analysoitu.

Valitse haluamasi toiminto:

- 1) Lue tiedosto
- 2) Analysoi
- 3) Kirjoita tiedosto
- 0) Lopeta

Anna valintasi: 3

Anna kirjoitettavan tiedoston nimi: L07T3T1.txt

```
Tulosteet kirjoitettu tiedostoon 'L07T3T1.txt'
```

```
Valitse haluamasi toiminto:
```

- ```
1) Lue tiedosto
2) Analysoi
3) Kirjoita tiedosto
0) Lopeta
```

```
Anna valintasi: 0
```

```
Lopetetaan.
```

```
Kiitos ohjelman käytöstä.
```

### Esimerkki tulostiedostosta 'L07T3T1.txt':

```
Analysoitiin 20 € seteleiden lukumääriä.
```

```
Keskimäärin seteleitä oli kuukausittain 1885334600 kpl.
```

```
Vähimmillään seteleitä oli 1801616000 kpl, jolloin niiden arvo oli 36032320000 €.
```

```
Enimmillään seteleitä oli 1961761000 kpl, jolloin niiden arvo oli 39235220000 €.
```

## L07T4: Tiedoston rivin jako tietoalkioihin (M)

*Tehtävän taso: Minimitaso*

Data-tiedostoissa L07T4D1.txt ja L07T4D2.txt on useita tietoalkioita samalla rivillä erotettuna erotin-merkillä. Tee ohjelma, joka

1. Lukee tiedoston rivi kerrallaan, tiedoston muoto näkyy alla.
2. Jakaa rivin tietoalkioihin erotinmerkkien (;) kohdista ohjelmointioppaan luvun 6 mukaisesti `split`-komennolla.
3. Tulostaa luetussa tiedostossa olevat tiedot käyttäjälle esimerkkiajon mukaisesti. Huomaa, että tämä ohjelma vain tulostaa tiedot, jolloin tietojen tietotyyppejä ei tarvitse vaihtaa.

Tiedoston alussa on otsikkorivi, joka tulee jättää huomiotta datan käsittelyssä, jolloin kaikki data-rivit voidaan käydä läpi samalla tavalla. Ole tarkkana datassa olevien rivinvaihtojen kanssa. Tehtävä on hyvin samantyyppinen kuin L06T5, mutta nyt pystymme pilkkomaan rivin ja käsittelemään pilkottua dataa merkkijonoleikkauksen sijaan listalla ja yhdessä sarakkeessa olevat arvot voivat olla keskenään eri levyisiä. Tämä ohjelma on niin lyhyt, että se kannattaa toteuttaa kokonaan pääohjelmassa ja kaikki tulosteet voivat olla merkkijonoja.

### Esimerkki syötetiedostosta 'L07T4D1.txt':

```
Sessio;Hahmo;Noppien määrä;Noppa;Noppien silmäluvut
2.1;Jaska;1;d6;4
2.1;Jaska;5;d6;1,1,2,2,3
2.1;Julia;1;d20;2
2.3;Jaska;2;d20;5,14
2.4;Olli-Pekka;8;d6;5,5,5,4,2,3,3,3
2.4;Olli-Pekka;2;d6;6,5
```



## Esimerkkiajo 1

### *Syötteet:*

```
L07T4D1.txt
```

### *Tuloste:*

```
Anna tiedoston nimi: L07T4D1.txt
Nopanheiton tulokset ovat seuraavat:
Pelaaja 'Jaska' heitti 1 kertaa noppaa d6 ja sai silmäluvut '4'.
Pelaaja 'Jaska' heitti 5 kertaa noppaa d6 ja sai silmäluvut '1,1,2,2,3'.
Pelaaja 'Julia' heitti 1 kertaa noppaa d20 ja sai silmäluvut '2'.
Pelaaja 'Jaska' heitti 2 kertaa noppaa d20 ja sai silmäluvut '5,14'.
Pelaaja 'Olli-Pekka' heitti 8 kertaa noppaa d6 ja sai silmäluvut '5,5,5,4,2,3,3,3'.
Pelaaja 'Olli-Pekka' heitti 2 kertaa noppaa d6 ja sai silmäluvut '6,5'.
Kiitos ohjelman käytöstä.
```

## L07T5: Valikkopohjainen ohjelma tuotetietojen hallintaan luokalla ja oliolistalla (P)

### *Tehtävän taso: Perustaso*

Tee valikkopohjainen ohjelma, joka pystyy kysymään käyttäjältä uusien tuotteiden tietoja, lukemaan aiemmat tiedot tiedostosta, tulostamaan kaikki annetut tiedot näytölle, tallettamaan kaikki tiedot tiedostoon ja lopettamaan ohjelman suorittamisen esimerkkiajon mukaisesti. Määrittele ohjelmassa luokka, jossa on kaksi jäsenmuuttujaa yksi tuotteen nimeä (merkkijono) ja yksi hintaa (desimaaliluku) varten. Toteuta ohjelmaan seuraavat toiminnot:

1. Käyttäjän valitessa uusien tietojen antamisen, kutsu sopivaa aliohjelmaa ja välitä sille parametrina lista, luo aliohjelmassa uusi olio, sijoita käyttäjän antamat tiedot olioon, lisää olio listaan ja palauta lista paluuarvona kutsuvaan ohjelmaan. Näin saat lisättyä uuden tuotteen tiedot oliolistaan.
2. Käyttäjän valitessa tietojen lukemisen, välitä sopivalle aliohjelmalle tiedoston nimi ja lista. Lue tiedoston rivit, luo jokaista riviä varten olio ja sijoita kunkin sarakkeen arvo sopivaan olion jäsenmuuttujaan. Palauta lista paluuarvona kutsuvaan ohjelmaan. Näin saat lisättyä useita tuotteita oliolistaan.
3. Käyttäjän valitessa tietojen tulostuksen, välitä sopivalle aliohjelmalle lista ja tulosta tiedot siellä.
4. Käyttäjän valitessa tietojen tallennuksen, välitä sopivalle aliohjelmalle tallennettavan tiedoston nimi sekä lista ja tallenna tiedot tiedostoon aliohjelmassa.
5. Käyttäjän valitessa lopeta-toiminnon, ohjelma suorittaa lopetusrutiinit ja lopettaa ohjelman. Nämä sisältävät tässä vaiheessa oliolistan tyhjennyksen ja lopetustulosteiden näyttämisen. Muista tyhjentää tuotelista ohjelman lopussa.

Ohjelman alussa ohjetekstin tulostamisen jälkeen kysy käyttäjältä tiedoston nimi. Samaa tiedostoa käytetään tietojen lukemisessa ja kirjoittamisessa, joten huomaa että testausvaiheessa kannattaa käyttää omaa tiedostoa, sillä tiedosto muuttuu kirjoitettaessa.

Tulostettavien ja tallennettavien merkkijonojen muotoiluun voi käyttää format-käskyä tai aiemmin läpikäytyjä merkkijonojen muodostuskäskyjä. Tuotteiden hinnat tallennetaan tiedostoon kahden desimaalin tarkkuudella. Mikäli ohjelma ei tunnista käyttäjän antamaa valintaa, ilmoittaa se käyttäjälle tästä ilmoituksella "Tuntematon valinta, yritä uudestaan."

### Esimerkki syötetiedostosta 'L07T5D1.txt':

```
Tuotteen nimi;Tuotteen hinta
puhelimen laturi;33.95
USB-C-kaapeli;15.99
```

### Esimerkkiajo 1

#### *Syötteet:*

```
2
L07T5D1.txt
3
1
näytön suojakalvo
9.99
3
4
L07T5T1.txt
0
```

#### *Tuloste:*

```
Tämä ohjelma hallitsee tuotteiden tietoja listalla.
Valitse haluamasi toiminto:
1) Anna uuden tuotteen tiedot
2) Lue tiedot tiedostosta
3) Tulosta tiedot listalta
4) Tallenna listan tiedot tiedostoon
0) Lopeta
Anna valintasi: 2
Anna luettavan tiedoston nimi: L07T5D1.txt
Tiedosto L07T5D1.txt luettu.

Valitse haluamasi toiminto:
1) Anna uuden tuotteen tiedot
2) Lue tiedot tiedostosta
3) Tulosta tiedot listalta
4) Tallenna listan tiedot tiedostoon
0) Lopeta
Anna valintasi: 3
Listalta löytyy seuraavat tuotteet ja hinnat:
puhelimen laturi 33.95
USB-C-kaapeli 15.99

Valitse haluamasi toiminto:
1) Anna uuden tuotteen tiedot
2) Lue tiedot tiedostosta
3) Tulosta tiedot listalta
```

```
4) Tallenna listan tiedot tiedostoon
0) Lopeta
Anna valintasi: 1
Anna tuotteen nimi: näytön suojakalvo
Anna tuotteen hinta (e): 9.99

Valitse haluamasi toiminto:
1) Anna uuden tuotteen tiedot
2) Lue tiedot tiedostosta
3) Tulosta tiedot listalta
4) Tallenna listan tiedot tiedostoon
0) Lopeta
Anna valintasi: 3
Listalta löytyy seuraavat tuotteet ja hinnat:
puhelimien laturi 33.95
USB-C-kaapeli 15.99
näytön suojakalvo 9.99

Valitse haluamasi toiminto:
1) Anna uuden tuotteen tiedot
2) Lue tiedot tiedostosta
3) Tulosta tiedot listalta
4) Tallenna listan tiedot tiedostoon
0) Lopeta
Anna valintasi: 4
Anna kirjoitettavan tiedoston nimi: L07T5T1.txt
Tiedot kirjoitettu tiedostoon 'L07T5T1.txt'.

Valitse haluamasi toiminto:
1) Anna uuden tuotteen tiedot
2) Lue tiedot tiedostosta
3) Tulosta tiedot listalta
4) Tallenna listan tiedot tiedostoon
0) Lopeta
Anna valintasi: 0
Lopetetaan.

Kiitos ohjelman käytöstä.
```

### Esimerkki tulostiedostosta 'L07T5T1.txt':

```
Tuotteen nimi;Tuotteen hinta
puhelimien laturi;33.95
USB-C-kaapeli;15.99
näytön suojakalvo;9.99
```