

# Harjoitustyö Tavoitetaso

## Sisällysluettelo

Perustason työn toteutus	1
Luettavan tiedoston rakenne	2
Analyysien tallentamat tiedostot	2
Tuloksista Excelillä tehtyjä esimerkkikuvaajia	2
Esimerkki syötetiedostosta	3
Esimerkkiajo 1	4
Esimerkki tulostiedostosta 1	6

Tavoitetason tehtävä perustuu perustason tehtävään eli toteuta ensin toimiva perustason ohjelma ja laajenna ratkaisuaasi sen jälkeen lisäämällä Korkeasaaren kävijätilastojen lukeminen sademäärän ja kävijämäärän yhdistämiseksi, ja kuukausittainen analyysi kävijämäärästä sateisina päivinä. Kävijämäärädatan lukeminen ja yhdistäminen sademääriin tarkoittaa erillistä valintaa kävijämäärädataa sisältävän tiedoston lukemiseen ja sen yhdistämistä valinnassa 1 luettuihin tietoihin. Kuukausianalyysissä jatketaan perustason kategorisointianalyysiä laskemalla keskimääräistä päivittäistä kävijämäärää sademäärän mukaan yhdistämällä taulukkoon sarakkeiksi sademääräkkategoria ja riveiksi kuukausi. Muilta osin tavoitetaso ohjelman toiminnalle on samat vaatimukset kuin perustason ohjelmalle. Datatiedostot ovat samat, joita käytetään minimi- ja perustasoilla. Harjoitustyö palautetaan Moodlen Harjoitustyö-lehdelle, missä on selitetty myös harjoitustyön arviointi. CodeGrade tarkistaa ohjelman toiminnan ja sen kirjoittamien tekstitiedostojen sisällöt, jonka jälkeen assistentti tarkistaa ohjelman rakenteen. Ohjelman rakenteesta saa palautetta myös ASPA-ohjelmalla, ks. Moodlen ASPA-välilehti. Harjoitustyö on henkilökohtainen tehtävä, joten jokainen kirjoittaa itse oman ohjelmansa ja kaikki ohjelmaan vaikuttaneet henkilöt ja lähteet tulee mainita ohjelman alkukommenteissa.

## Perustason työn toteutus

Tavoitetaso työn tulee noudattaa tyyliohjetta ottaen huomioon seuraavat tarkennukset:

- Kävijämäärädatan lukemisessa** kannattaa suoraan yhdistää kävijämäärät olemassa olevaan sademäärään päivämäärän perusteella. Käytännössä tämä tarkoittaa valinnan 2 päivittäisen tulosolion laajentamista lisäämällä siihen eri tulosuunnista (Mustikkamaa, Kauppatori ja Hakaniemi) tulevat kävijämäärät. Tietojen yhdistämisessä voidaan yksinkertaistuksen vuoksi olettaa, että sademäärä- ja kävijämäärätiedostoissa on dataa samoilta päiviltä
  - Moodlessa on tiedosto `kumpula_tavoitetesti.txt`, jossa on tunneittaiset tiedot (360 riviä) samoille 15 päivälle, jotka löytyvät `korkeasaari_15.txt` tiedostosta
- Yhdistetyn datan kirjoittamisessa** tiedostoon tallennetaan päivittäisistä tiedoista päivämäärä, sademäärä ja kunkin tulosuunnan kävijämäärät omiin sarakkeisiinsa
- Kuukausianalyysissä** lasketaan keskimääräiset päivittäiset kävijämäärät eri kuukausina ja eri kategorian sadepäivinä. Eli esimerkiksi jos tammikuussa on 20 runsaan sateen päivää (kategoria 1, > 4,5 mm sadetta) ja näinä päivinä käy yhteensä 1000 ihmistä on ensimmäisen alkion arvo 50,0. Tiedot tallennetaan matriisiin, joten jokainen alkio tarvitsee rivi- ja sarakeindeksit, joista

kuukausi (1-12) saadaan aikaleiman `struct_time`-rakenteesta ja kategoriatieto (1-4) valinnan 2 kategorisointianalyysistä. Matriisina tulee käyttää `numpy`-matriisia L10 mukaisesti ja kaikki arvot tulee pyöristää yhden desimaalin tarkkuuteen

- Matriisin tallentamisessa rivin alkuun tulee kuukauden nimi, joka tulee ottaa `datetime`-kirjastosta lyhennettynä. Tämä voi osalla kotikoneista antaa nimet suomeksi, mutta CodeGradessa nämä tulevat kaikilla englanniksi
- Lisäksi tätä analyysiä varten valinnan 2 kategorisointianalyysissä olion jäsenmuuttujaksi kannattaa lisätä kategoriatieto

4. Tavoitetason valinnat 6 ja 7 vaativat valintojen 1, 2 ja 5 suorittamista (kts. esimerkkiajo)

5. Analyysien tulokset tallennetaan tietorakenteisiin, jotka tulee luoda pääohjelmassa ja välittää parametrina aliohjelmiin. Uudelleen analysoinnin on oltava mahdollista, joten tietorakenteet tulee tyhjentää, ja matriisin arvot nollata, analyysialiohjelmien alussa

6. Ohjelman lopuksi matriisien alkiot tulee poistaa ja kaikki pääohjelmassa luodut listat tyhjentää.

Ohjelman tavoitetason toiminnot näkyvät parhaiten esimerkin valikosta, mutta uusina valintoina käyttäjä voi valita kävijämäärien lukemisen, yhdistettyjen tietojen tallentamisen ja kuukausianalyysin. Tiedoston lukeminen on käsitelty Kohdassa 2 ja kirjoittaminen Kohdassa 3.

## Luettavan tiedoston rakenne

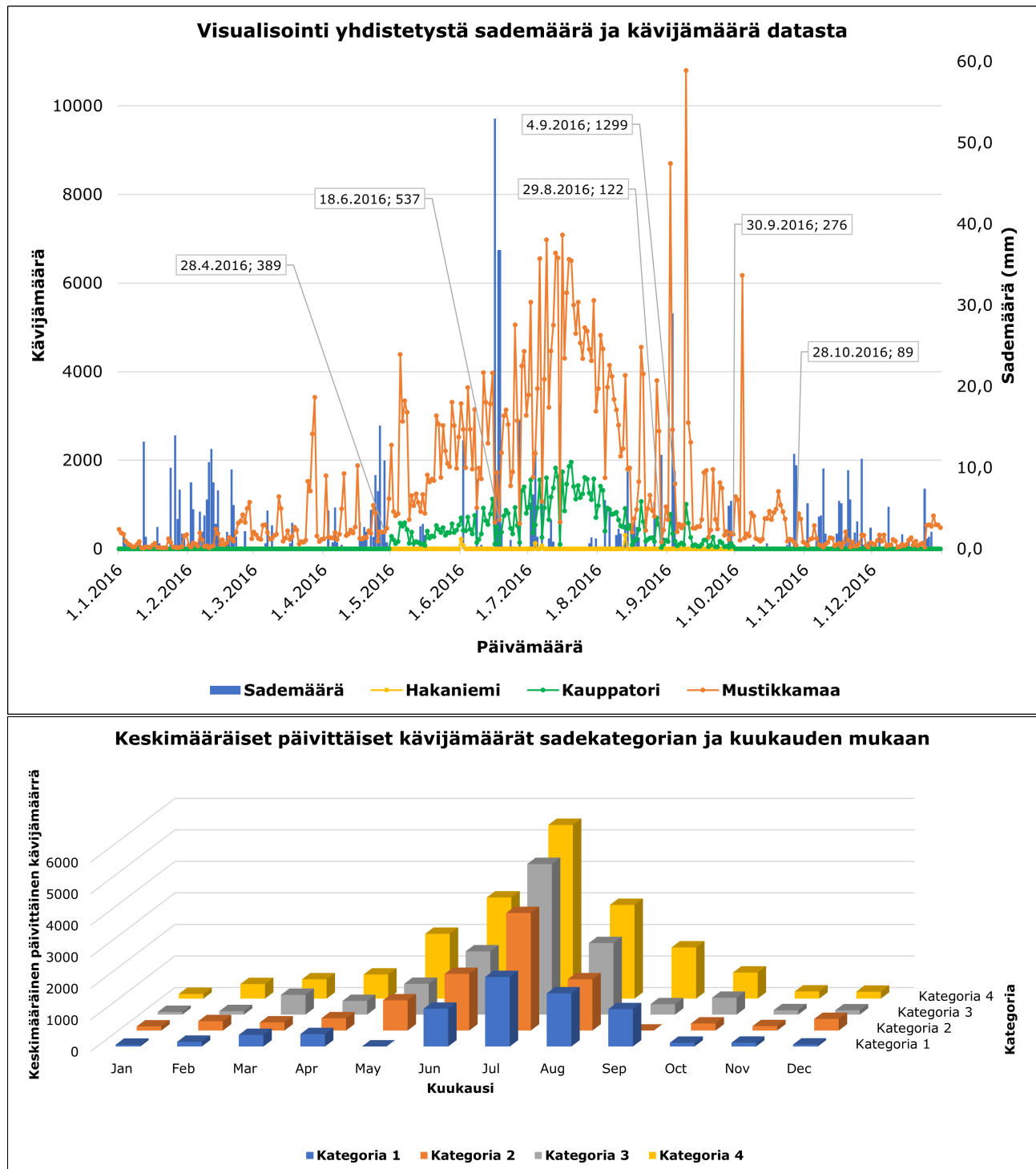
Luettavan Korkeasaaren kävijämäärietiedoston rakenne näkyy alla. Yksi rivi vastaa saapuneiden kävijöiden määrää yhden päivän aikana. "Pvm"-sarake, joka sisältää päivämäärietiedot dd.mm.YYYY formaatissa, noudattaa L08 läpikäytyjä periaatteita, Mustikkamaa, Kauppatori ja Hakaniemi -sarakkeet sisältävät nimeä vastaavan sisäänkäynnin kautta saapuneiden kävijöiden määrän kyseisenä päivänä. Alla on esitetty luettavan tiedoston rakenne käyttäen Moodlesta ja CodeGradesta nimellä `korkeasaari_15.txt` löytyvän tiedoston alkua ja laajempi kaikki tiedot sisältävä tiedosto on `korkeasaari_2016.txt`. Tiedosto on järjestetty "Pvm"-sarakkeen mukaan kronologiseen järjestykseen. Kävijöitä ei ole vuoden jokaiselle päivälle eikä välttämättä edes jokaiselle kuukaudelle.

## Analyysien tallentamat tiedostot

Uusissa valinnoissa kirjoitettavien tiedostojen rakenne näkyy alla. Kummassakin valinnassa on käytetty yhdistettyä dataa.

## Tuloksista Excelillä tehty esimerkkikuvaaja

Alla Kuvissa 1 ja 2 on vuoden 2016 tiedoilla tehtyjen analyysien tuloksista piirretyt Excel-kaaviot. CodeGrade tarkistaa vain tehdyn tiedoston sisällön, joten kuvaajien tekeminen jää oman mielenkiinnon ja harrastuksen varaan. Kuvassa 1 on yhdistetty data, johon on korostettu yksittäisten sateisten päivien kävijämääriä Mustikkamaan tulosuunnasta. Tästä voidaan nähdä että vaikuttaisi siltä että sade vähentää ihmisten halua mennä Korkeasaareen varsinkin kesäkuukausina. Kuvassa 2 on kuukausianalyysin tulos vuoden 2016 datalle. Tuloksista nähdään selvästi että keskimääräinen päivittäinen kävijämäärä on pääsääntöisesti matalin suurimman sateen (kategoria 1) aikana ja korkein vähäisimmän sateen (kategoria 4) aikana.



Esimerkki syötetiedostosta 'kumpula\_tavoitetesti.txt':

```
Aikaleima (YYYY.mm.dd HH:MM);Aikavyöhyke;Sademäärä (mm)
2016.01.01 00:00;UTC+2;0
2016.01.01 01:00;UTC+2;0
2016.01.01 02:00;UTC+2;0
2016.01.01 03:00;UTC+2;0
...
2016.06.02 14:00;UTC+3;0
2016.06.02 15:00;UTC+3;3.1
```

```
2016.06.02 16:00;UTC+3;5.6
2016.06.02 17:00;UTC+3;4.3
2016.06.02 18:00;UTC+3;0.4
...
2016.10.20 22:00;UTC+3;0
2016.10.20 23:00;UTC+3;0
```

### Esimerkki syötetiedostosta 'korkeasaari\_15.txt':

```
Pvm (dd.mm.YYYY);Mustikkamaa;Kauppatori;Hakaniemi
01.01.2016;443;0;0
02.01.2016;363;0;0
03.01.2016;327;0;0
02.05.2016;638;204;0
03.05.2016;634;117;0
04.05.2016;627;165;0
01.06.2016;2583;479;219
02.06.2016;2279;342;76
04.07.2016;1610;418;125
05.07.2016;2691;927;0
06.07.2016;5001;1554;0
07.07.2016;811;191;68
08.07.2016;2902;926;0
09.07.2016;5373;1602;0
20.10.2016;1299;0;0
```

### Esimerkkiajo 1

#### *Syötteet:*

```
1
kumpula_tavoitetesti.txt
2
5
korkeasaari_15.txt
6
tavoite_tulos_yhdistetty_15.txt
7
tavoite_tulos_matriisi_15.txt
0
```

#### *Tuloste:*

```
Valitse haluamasi toiminto:
1) Lue tiedosto
2) Analysoi
3) Kirjoita tiedosto
4) Analysoi viikontäivittäiset sademäärät
5) Lue ja yhdistä Korkeasaari tiedosto
6) Kirjoita yhdistetty data tiedostoon
7) Analysoi viikoittaiset kävijämäärät
0) Lopeta
Anna valintasi: 1
Anna luettavan tiedoston nimi: kumpula_tavoitetesti.txt
Tiedosto 'kumpula_tavoitetesti.txt' luettu.
Tiedostosta lisättiin 360 datariviä listaan.
```

Valitse haluamasi toiminto:

- 1) Lue tiedosto
- 2) Analysoi
- 3) Kirjoita tiedosto
- 4) Analysoi viikontäivittäiset sademäärät
- 5) Lue ja yhdistä Korkeasaari tiedosto
- 6) Kirjoita yhdistetty data tiedostoon
- 7) Analysoi viikoittaiset kävijämäärät
- 0) Lopeta

Anna valintasi: 2

Päivittäiset summat laskettu 15 päivälle.

Päivät kategorisoitu 4 kategoriaan.

Valitse haluamasi toiminto:

- 1) Lue tiedosto
- 2) Analysoi
- 3) Kirjoita tiedosto
- 4) Analysoi viikontäivittäiset sademäärät
- 5) Lue ja yhdistä Korkeasaari tiedosto
- 6) Kirjoita yhdistetty data tiedostoon
- 7) Analysoi viikoittaiset kävijämäärät
- 0) Lopeta

Anna valintasi: 5

Anna luettavan tiedoston nimi: korkeasaari\_15.txt

Tiedosto 'korkeasaari\_15.txt' luettu.

Tiedot yhdistetty, kävijämäärä on yhteensä 34994.

Valitse haluamasi toiminto:

- 1) Lue tiedosto
- 2) Analysoi
- 3) Kirjoita tiedosto
- 4) Analysoi viikontäivittäiset sademäärät
- 5) Lue ja yhdistä Korkeasaari tiedosto
- 6) Kirjoita yhdistetty data tiedostoon
- 7) Analysoi viikoittaiset kävijämäärät
- 0) Lopeta

Anna valintasi: 6

Anna kirjoitettavan tiedoston nimi: tavoite\_tulos\_yhdistetty\_15.txt

Tiedosto 'tavoite\_tulos\_yhdistetty\_15.txt' kirjoitettu.

Valitse haluamasi toiminto:

- 1) Lue tiedosto
- 2) Analysoi
- 3) Kirjoita tiedosto
- 4) Analysoi viikontäivittäiset sademäärät
- 5) Lue ja yhdistä Korkeasaari tiedosto
- 6) Kirjoita yhdistetty data tiedostoon
- 7) Analysoi viikoittaiset kävijämäärät
- 0) Lopeta

Anna valintasi: 7

Kuukausikohtaiset sademäärät analysoitu.

Anna kirjoitettavan tiedoston nimi: tavoite\_tulos\_matriisi\_15.txt

Tiedosto 'tavoite\_tulos\_matriisi\_15.txt' kirjoitettu.

Valitse haluamasi toiminto:

- 1) Lue tiedosto
- 2) Analysoi
- 3) Kirjoita tiedosto
- 4) Analysoi viikontäivittäiset sademäärät

```
5) Lue ja yhdistä Korkeasaari tiedosto
6) Kirjoita yhdistetty data tiedostoon
7) Analysoi viikoittaiset kävijämäärät
0) Lopeta
Anna valintasi: 0
Lopetetaan.
```

Kiitos ohjelman käytöstä.

#### Esimerkki tulostiedostosta 'tavoite\_tulos\_yhdistetty\_15.txt':

```
Pvm;Sademäärä;Mustikkamaa;Kauppatori;Hakaniemi
01.01.2016;0.0;443;0;0
02.01.2016;0.0;363;0;0
03.01.2016;1.9;327;0;0
02.05.2016;0.0;638;204;0
03.05.2016;0.0;634;117;0
04.05.2016;0.0;627;165;0
01.06.2016;0.0;2583;479;219
02.06.2016;13.4;2279;342;76
04.07.2016;12.3;1610;418;125
05.07.2016;1.5;2691;927;0
06.07.2016;0.0;5001;1554;0
07.07.2016;6.2;811;191;68
08.07.2016;0.0;2902;926;0
09.07.2016;0.0;5373;1602;0
20.10.2016;0.0;1299;0;0
```

#### Esimerkki tulostiedostosta 'tavoite\_tulos\_matriisi\_15.txt':

```
Kuukausi;Kategoria 1;Kategoria 2;Kategoria 3;Kategoria 4
Jan;0.0;327.0;0.0;403.0
Feb;0.0;0.0;0.0;0.0
Mar;0.0;0.0;0.0;0.0
Apr;0.0;0.0;0.0;0.0
May;0.0;0.0;0.0;795.0
Jun;2697.0;0.0;0.0;3281.0
Jul;1611.5;3618.0;0.0;5786.0
Aug;0.0;0.0;0.0;0.0
Sep;0.0;0.0;0.0;0.0
Oct;0.0;0.0;0.0;1299.0
Nov;0.0;0.0;0.0;0.0
Dec;0.0;0.0;0.0;0.0
```