

1. OHJELMOINTI LUOKILLA 7–9

Ohjelmoinnin opiskelu syventää oppilaan algoritmista ajattelua ja kehittää ongelmanratkaisutaitoja. Luokilla 7–9 pyritään siirtymään graafisista ohjelmointiympäristöistä jonkin ohjelmointikielen käyttämiseen. Säde-sarjassa ohjelmointikieliksi on valittu Python ja Scratch. Uuden opetussuunnitelman siirtymävaiheessa oppilailla voi olla hyvin vähän kokemusta ohjelmoinnista, joten Säde-sarjan toisella ohjelmointipolulla harjoitellaan graafista Scratch-kieltä. Opettaja voi valita näistä kahdesta rinnakkaisesta ohjelmointipolusta oppilaidensa taitoihin sopivan.

Säde-sarjan ohjelmointiharjoitukset on pyritty liittämään opiskeltaviin matematiikan sisältöihin. Näin harjoitukset vahvistavat ja syventävät myös matematiikan taitoja. Ohjelmointiharjoituksissa harjoitellaan ensin ohjelmointikielen kirjoittamista ja ohjelmallisia rakenteita. Taitojen kasvaessa siirrytään suunnittelemaan ja toteuttamaan pieniä ohjelmia ja soveltamaan opittuja taitoja itsenäisissä projekteissa.

Tavoitteena on, että ohjelmallisen ajattelun ja ohjelmoinnin opiskeluun käytettäisiin noin 12–16 tuntia lukuvuoden aikana yhden tai kahden tunnin kokonaisuuksina.

2. YLEISET OHJEET

SÄTEEN OHJELMOINTIALUSTA

Ohjelmointiin käytettävä alusta löytyy osoitteesta
<http://sade-oppiminen.herokuapp.com/>

Alusta toimii suoraan selaimessa, mitään ei tarvitse asentaa paikallisesti koneelle.

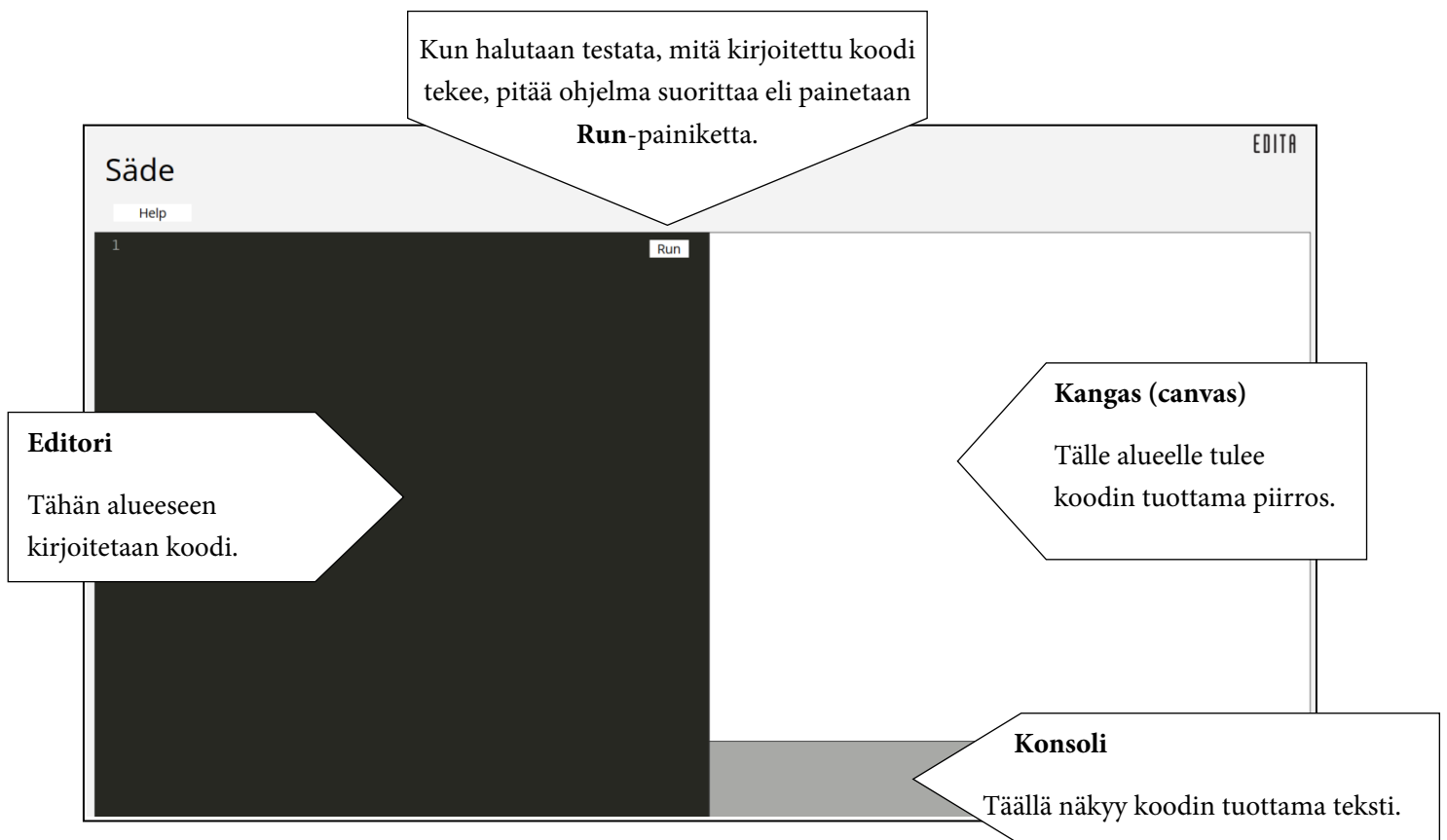
Alustaa on kehitetty yhteistyössä Futuricen kanssa, ja se pohjautuu avoimeen lähdekoodiin.

Alustassa käytetään Python-kielen versiota 2.6.



Säteen ohjelmointialustalle pääsee tällä QR-koodilla.

ALUSTAN OSAT JA KESKEISET TOIMINNOT




Help-painikkeen takaa löytyvät seuraavat linkit:

- Tutorial on Python programming language eli englanninkielinen Python-kielen ohjesivusto.
- Turtle commands eli englanninkielinen ohjesivusto kilpikonnaohjelmointiin. Kilpikonnaohjelmointia käytetään Python-kielessä piirtämiseen.
- Chilocorn Code Club ohjaa Futuricen ohjelmointitehtäväsivuille ja heidän ohjelmointialustansa.

HARJOITUSOHJEET JA NIIDEN RAKENNE

Säde 1 Digitaalinen opettajan aineisto sisältää kuusi Python-harjoitusta. Jokaiseen kurssiin liittyy kaksi harjoitusta, toinen kurssin alkuosaan ja toinen kurssin loppuosaan.

Harjoitusohjeet muodostuvat teoriasta ja tehtävistä. Harjoituksen alussa on kerrottu ohjelmointiin liittyvät tavoitteet ja mitä matematiikan sisältöjä harjoituksessa käsitellään. Väliotsikot kertovat, mitä aihetta harjoitellaan. Teoriaosuuksissa on esimerkkitehtäviä, ja keskeiset käsitteet ja komennot tuodaan esille tietolaatikoissa. Oppilaiden kannattaa kirjoittaa esimerkkien koodit koodieditoriin ja tutkia, mitä koodi tekee.

Jokaisessa harjoituksessa on 10 tehtävää, ja ne alkavat edellisen tai edellisten harjoitusten keskeisiä sisältöjä kertaavilla tehtävillä. Tehtävien edessä on kuvake . Joidenkin harjoitusten lopussa on lisätehtäviä nopeasti eteneville.

Mikäli ohjelmointi tehdään yhtenä kokonaisuutena jossain vaiheessa lukuvuotta, voi harjoituskokoelmien kertaavia tehtäviä hypätä yli. Kertaavat tehtävät tukevat asioiden muistiin palauttamista lukuvuodelle hajautetussa ohjelmoinnin opiskelussa.

TYÖSKENTELYOHJEITA

Oppilaat voivat kirjoittaa joko kaikki harjoitukset allekkain editoriin tai yhden harjoituksen kerrallaan. Jos oppilas haluaa tekemänsä koodit talteen myöhempää tarkastelua tai käyttöä varten, hän voi kopioida kirjoittamansa koodin ja liittää sen esimerkiksi tekstitiedostoon tai valmiiseen vastauspohjatiedostoon.

Jos opettaja haluaa oppilaidensa palauttavan tekemiään koodeja, oppilaat voivat palauttaa tallentamansa koodit yleisesti käytössä olevien palautustapojen mukaisesti esimerkiksi koulun verkkolevyille, oppimisalustalle tai pilvipalveluun. Näin opettaja voi seurata oppilaidensa etenemistä ohjelmointiharjoituksissa.

PYTHON-KOODIN TESTAAMINEN, SEURAAMINEN RIVI RIVILTÄ JA MUUT PYTHON-ALUSTAT

Kirjoitettua koodia voi testata sekä seurata sen etenemistä rivi riviltä sivuston <http://pythontutor.com/> avulla.

Harjoitukset voi tehdä myös jollakin toisella Python-alustalla.

Koneelle asennettavat alustat

Sivulta <https://www.python.org/> saa asennettua koneelle joko versiota 2.x tai 3.x tukevan alustan.

Selainpohjaisia alustoja

<https://www.python.org/> sivustolta aukeaa online-konsoli, joka käyttää versiota 3.6.

<http://www.codeskulptor.org/> käyttää versiota 2.

<https://trinket.io/> sisältää Pythonin lisäksi muita ohjelmointikieliä ja graafisen ohjelmointialustan.

<https://www.brython.info/> käyttää versiota 3.

PYTHON-VERSIOIDEN 2.X JA 3.X KESKEISIÄ EROJA

Ohjeet on tehty Python-kielen versioille 2. Mikäli käytössäsi on jotakin Python-kielen versiota 3 tukeva alusta, seuraavat muutokset on hyvä ottaa huomioon. Alla olevaan taulukkoon on koottu muutamia keskeisiä versioiden eroja.

Toiminto	PYTHON 2.x	PYTHON 3.x
Tulostaminen eli print	Seuraavat komennot tuottavat saman tulosteen: print ("Luku on ") , luku1 tai print "Luku on", luku1	Print-komennon kanssa täytyy käyttää sulkeita. Huomaa jälkimmäisen sulkeen paikka. print("Luku on ", luku1)
Jakolasku /	print 1 / 2 tuottaa vastaukseksi luvun 0. Jos vastaukseksi haluaa 0.5, jaettava tai jakaja täytyy muuttaa desimaalilukumuotoon.	print 1 / 2 tulostaa vastaukseksi 0.5.

3. HARJOITUS 1

HARJOITUKSEN KESTO

Harjoituksen tekemiseen on hyvä varata noin 1 x 45 min.

TAVOITTEET

- Tutustutaan työskentelyalustaan ja Python-kielen versioiden 2 kirjoittamiseen.
- Opitaan tulostuskomento, muuttujien käyttö ja laskutoimitusten kirjoittaminen.
- Tutustutaan muuttujatyyppeihin.

Matematiikan sisällöt

Tehtävissä harjoitellaan kokonais- ja desimaalilukujen peruslaskutoimituksia. Tehtyjä ohjelmia voi testata oppikirjan tehtävillä.

HUOMIOITA TEHTÄVISTÄ

Tulostaminen tarkoittaa halutun lopputuloksen tulostamista näytölle. Tulostuskomento on **print**. Tulostettavaksi haluttu teksti eli merkkijono laitetaan lainausmerkkeihin, esim. `print "kissa"` tulostaa näytölle sanan kissa. Tulostettavaksi halutut muuttujan arvot tai luvut kirjoitetaan sellaisenaan `print`-komennon perään. Kun halutaan tulostettaessa yhdistää tekstiä ja lukuja, pitää luvut muuttaa merkkijonoiksi. Tämän harjoituksen tehtävissä muuttujatyypin muunnokset tehdään tulostuskomennon sisällä.

Esimerkeissä näytetään mallia, kuinka Python-kieltä kirjoitetaan. Python-kieltä kirjoittaessa kirjainkoolla on merkitystä, joten isojen ja pienten kirjaimien kanssa pitää olla tarkkana. Desimaalilukuja kirjoittaessa desimaalierottimena on piste.

Tehtävässä 5 tuloste $8 + 4 = 12$ saadaan koodilla `print " luku1, "+", luku2, "=", luku1 + luku2`.

Tehtävän 6 viimeinen kohta voi olla osalle oppilaista haastava.

Tehtävässä 7 pyydetään kirjoittamaan kokonaisluku desimaalilukuna. Esimerkiksi luku 1 on desimaalilukuna kirjoitettuna 1,0.

Tehtävän 10 viimeisessä kohdassa olisi tarkoitus käyttää **str()**-komentoa.

Tehtävät 6 ja 10 esittelevät erilaiset tavat samantyylisten lauseiden tulostamiseen.

Nopeasti eteneville on monisteen lopussa kolme lisätehtävää. Lisätehtävissä tutkitaan vastaluvun kirjoittamista, useamman muuttujan käyttämistä peruslaskutoimituksissa ja harjoitellaan laskujärjestystä.

MALLIRATKAISUT

```
# Tehtävä 1
print 'Etunimi'
print 'Etunimi Sukunimi'
print u'Tänään on hyvä päivä.'

# Tehtävä 2
print 27 + 32
print 14 - 19
print 5 * 12
print 72 / 8

# Tehtävä 3
# Rivi print "luku1" tulostaa tekstin luku1.
# Rivi print luku1 tulostaa muuttujalle luku1 annetun arvon 4.

# Tehtävä 4
luku1 = 12
luku2 = 4
print luku1 + luku2
print luku1 - luku2
print luku1 * luku2
print luku1 / luku2
```

```
# Tehtävä 5
# Ensimmäisestä tulostusrivistä tulostuu lause: Laskutoimitus on 8 + 4
# Toinen tulostuslause tulostaa lauseen: Vastaus on 12

# Tehtävä 6
luku1 = 21
luku2 = -7
summa = luku1 + luku2
erotus = luku1 - luku2
tulo = luku1 * luku2
osamaara = luku1 / luku2 # muuttujan nimessä ei voi olla ä-kirjainta
print summa
print erotus
print tulo
print osamaara
print 'Lukujen ', luku1, ' ja ', luku2, ' summa on ', summa, ', erotus on ',
erotus, ', tulo on ', tulo, u' ja osamäärä on ', osamaara, '.'
```

```
# Tehtävä 7
luku1 = 1
luku2 = 2
# Molemmat muuttujat ovat kokonaislukuja eli tyyppiä int.
print luku1 / luku2
# Vastaukseksi tulee 0. Koska molemmat muuttujat ovat int-tyyppiä, antaa Python
2.6 vastaukseksi int-tyypin.
luku1 = 1.0 # Muutettu desimaaliluvuksi eli float-tyypiksi.
print luku1 / luku2
# Vastaukseksi saadaan 0.5. Koska toinen lähtöarvoista on float-tyyppiä, on
vastaus myös float-tyyppiä. Muuttuja luku2 on edelleen int-tyyppiä.

# Tehtävä 8
muuttuja1 = 'oppi'
muuttuja2 = 'kirja'
print muuttuja1 + muuttuja2
# Tulostaa oppikirja.
muuttuja2 = 1
print muuttuja1 + muuttuja2
# Saadaan virhe, koska yritetään yhdistää merkkijonoa ja lukua eli str-tyyppiä ja
int-tyyppiä, joita ei voi sellaisenaan yhdistää.
```

```
# Tehtävä 9
muuttuja1 = 'Etunimi'
muuttuja2 = 17
print muuttuja1 + str(muuttuja2)

# Tehtävä 10
luku1 = -5
luku2 = 1
luku3 = 7
summa = luku1 + luku2 + luku3
tulo = luku1 * luku2 * luku3
print 'Lukujen ' + str(luku1) + ', ' + str(luku2) + ' ja ' + str(luku3) +
' summa on ' + str(summa) + ' ja tulo on ' + str(tulo) + ' . '
```

LISÄTEHTÄVIEN MALLIRATKAISUT

```
# Lisätehtävä 1
# teht 105 a)
luku = -19
kohta_a = -luku
print luku
print kohta_a

# teht 105 b)
luku = -4
kohta_b = +luku
print luku
print kohta_b

# teht 105 c)
luku = +10
kohta_c = +(-luku)
print luku
print kohta_c

# teht 105 d) Tällä saat tulostettua luvun ja sen vastaluvun.
luku = 15
vastaluku = -luku
print luku
print vastaluku

# Ohjelma, joka tulostaa luvun ja sen vastaluvun vastaluvun.
luku = 15
vastaluvunVastaluku = -(-luku)
print luku
print vastaluvunVastaluku
```

```
# Lisätehtävä 2
# Määritetään muuttujat luku1, luku2 ja luku3 ja niille arvot.
luku1 = 4
luku2 = -5
luku3 = -2
# Määritetään apumuuttujat ja niille laskutoimitukset. Erotuksessa
luku2 ja luku3 vähennetään luku1:stä.
summa = luku1 + luku2 + luku3
erotus = luku1 - luku2 - luku3
tulo = luku1 * luku2 * luku3
print 'Summa', summa
print 'Erotus', erotus
print 'Tulo', tulo
```



```
# Lisätehtävä 3
# Tehtävä 240
# Tavoitteena on yhdellä ohjelmalla saada laskettua kaikki
# tehtävän kohdat. Vain muuttujien arvoja vaihdetaan kohdasta toiseen
# siirryttäessä.
luku1 = 15
luku2 = 3
luku3 = 4
# Kaikissa kohdissa yläosassa on erotus. Tehdään apumuuttuja
# erotus.
erotus = luku1 - luku2
# Lopullinen vastaus saadaan jakamalla erotus luku3:lla.
osamaara = erotus / luku3
# Tulostetaan vastaus.
print osamaara
# Vaihda arvot muuttujille luku1, luku2 ja luku3 ja suorita ohjelma
# uudestaan.
```

```
# Tehtävä 251
# Tavoitteena on tehdä ohjelma, jolla voidaan tutkia samojen
# lukujen erilaisia laskujärjestyksiä ja sen vaikutusta vastaukseen.
luku1 = 7
luku2 = 5
luku3 = 6
# Kirjoitetaan laskut laskutoimitusmerkkejä käyttäen.
kohta_a = luku1 + luku2 * luku3
kohta_b = (luku1 + luku2) * luku3
kohta_c = (luku1 + luku2) / luku3
kohta_d = luku1 * (luku2 + luku3)
print kohta_a
print kohta_b
print kohta_c
print kohta_d
```

```
# Tehtävä 255
# Nyt tarvitaan neljä muuttujaa. Tavoite on sama kuin tehtävän 251
# ohjelmassa.
luku1 = -3
luku2 = 4
luku3 = 8
luku4 = 2
# Kirjoitetaan laskut laskutoimitusmerkkejä käyttäen.
kohta_a = luku1 * luku2 + luku3 / luku4
kohta_b = luku1 * (luku2 + luku3) / luku4
kohta_c = luku1 * (luku2 + luku3 / luku4)
kohta_d = (luku1 * luku2 + luku3) / luku4
print kohta_a
print kohta_b
print kohta_c
print kohta_d
```