

SUORAKULMIOT (kirjan tehtävät 20-25, s. 194)

Tavoite

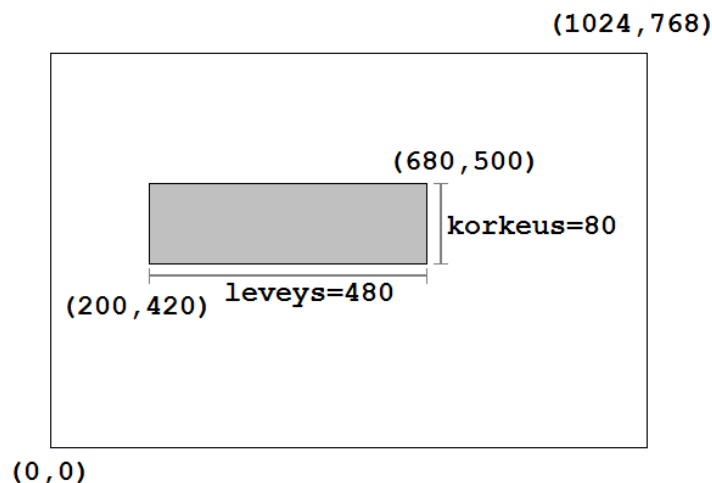
Oppia luokkien välisen periytymissuhteen määrittelyä ja käyttöä. Ymmärtää rajapinnan käsite. Osata tyyppinmuunnos.

Käsitteitä

Periytyminen, aliluokka, ylikuokka, ylikirjoittaminen, this vs. super, this() vs. super(), rajapinta

Alkutoimet: Kopioi tehtäväsarjan koodit (**OlsoTehtSarja3**) kurssin tietovarastosta omaan repositori-
oosi ja luo Eclipse-projekti ensimmäisen tehtäväsarjan ohjeita soveltaen. Projektissa on tehtävässä
mainitut luokat, mutta koodi puuttuu. **tests**-hakemistossa olevat koodit eivät käänny, ennen kuin olet
kirjoittanut tehtävien koodit.

Alla on mallikuva tehtäväsarjan käsitteistä ja koordinaatistosta. Esimerkissä näytölle on piirretty suorakulmio, jonka vasen alanurkka on pisteessä (200,420). Suorakulmion leveys on 480 ja korkeus 80. Kuvan suorakulmion ala olisi edellisten tulo eli 38400. Kuvasta nähdään myös, että suorakulmio mah-
tuisi kokonaan näytölle, jonka koko on 1024 x 768.



1. Ohjelmoi **Suorakulmio**-luokka, jossa suorakulmion ominaisuuksina ovat (private-määreiset, kokonaislukuarvoiset) leveys ja korkeus. Toteuta setterit ja getterit sekä kaksiparametrinen konstruktori, joka asettaa leveyden ja korkeuden (parametrit tässä järjestyksessä). Tee myös tyhjä oletuskonstruktori. Kirjoita metodi `int ala()`, jonka paluuarvona on suorakulmion ala.

Ylikirjoita `toString()`-metodi, joka palauttaa suorakulmion tiedot merkkijonona tämän mallin mukaan: `"leveys=99, korkeus=10"`.

Aja JUnit-testit (`SuorakulmioTest.java`) ensimmäisen tehtäväsarjan ohjeiden mukaisesti. Siirry seuraavaan tehtävään vasta, kun kaikki testit menevät läpi.

2. Ohjelmoi edellisestä luokasta periytyvä aliluokka **NaytonSuorakulmio**, joka kuvaa näytölle piir-
tyvää suorakulmiota. Aliluokan uusina ominaisuuksina ovat suorakulmion vasemman alanurkan
x- ja y-koordinaatit (jälleen private). Kirjoita neljän parametrin konstruktori (parametreina ensin
vasemman alanurkan koordinaatit, sitten leveys ja korkeus), joka kutsuu ylikuokan konstruktori
leveyden ja korkeuden asettamiseksi sekä asettaa alanurkan koordinaatit.

3. Kirjoita `NaytonSuorakulmio`-luokkaan metodi `boolean mahtuu(veys, korkeus)`, joka kertoo, mahtuuko suorakulmio kokonaan näytölle, jonka veys ja korkeus annetaan metodin parametreina (huomaa järjestys). Mahtumisella tarkoitetaan tässä sitä, että suorakulmio on kooltaan ja sijainniltaan sellainen, että se näkyisi näytöllä kokonaisuudessaan.
4. Ylikirjoita `NaytonSuorakulmio`-luokkaan metodi `toString()`, joka palauttaa olion tiedot merkkijonona tämän mallin mukaan: `"x=225, y=120, veys=800, korkeus=30"`. Kirjoita metodi siten, että se hyödyntää ylläluokkansa `toString()`-metodia veyn ja korkeuden osalta.

Aja JUnit-testit (`NaytonSuorakulmioTest.java`) ensimmäisen tehtäväsarjan ohjeiden mukaisesti. Siirry seuraavaan tehtävään vasta, kun kaikki testit menevät läpi.

5. Kirjoita **Testi**-niminen luokka, jossa
 - luot `NaytonSuorakulmio`-olion, jonka vasemman alanurkan (x,y)-koordinaatit ovat (225,120) ja jonka veys on 800 ja korkeus 30.
 - tulostat luomasi olion tiedot.
 - testaat `mahtuu()`-metodilla, mahtuuko luomasi olio näytölle, jonka koko (veys x korkeus) on 1024 x 768.
 - luot `Suorakulmio`-olion, jonka veys on 80 ja korkeus on 40.
 - tulostat luomasi olion tiedot, sekä tulostat sen alan metodilla `ala()`.

Jatkotehtäviä

6. Tutustu JUnit-testien koodeihin sekä logiikkaan, ja katso miten testit on kirjoitettu. Muuta jotain testissä käytettyä numeroarvoa siten, että kaikki testit eivät mene läpi, ja aja testit uudelleen. Mitä JUnit nyt tulostaa? Kopioi joku virheestä raportoiva JUnit-tuloste (screenshot) tehtäväpalautekseen.

Palauta lopuksi testit ennalleen.

7. Lisää projektiin **IKuvio**-niminen rajapinta-määrittely eli **interface IKuvio**. Määrittele rajapintaan yksi abstrakti metodi: **int ala()** (ei mitään muuta). Määrittele `Suorakulmio`-luokka toteuttamaan (implements) `IKuvio`-rajapinta.

Kirjoita **Testi2**-niminen luokka, jossa teet samat temput kuin **Testi**-luokassa (kopioi koodi), mutta luot muuttujat `IKuvio`-rajapinnan toteuttajaksi tyyliin:

```
IKuvio naytonSuorakulmio = new NaytonSuorakulmio(...);  
IKuvio suorakulmio = new Suorakulmio(...);
```

Tee tämän jälkeen kaikki tarvittavat muutokset koodiin, jotta saat `Testi2`:n toimimaan.