

Ülesanne

Lua kinopiletite müügi automaat **TicketMachine**, mille kaudu saab kasutaja osta pileteid. Piletil (**Ticket**) on kirjas filmi nimi ning number. Piletite numbrid on unikaalsed, ausal teel pileteid ostes ei ole võimalik saada kahte sama piletiganumbrit, seda ka erinevatest automaatidest ostes. Piletile realiseerida **toString()**-meetod, mis tagastab selle kirjelduse tekstikujul.

Kirjutada meetod **buyTicket()**, millega saab osta pileti etteantud filmile. Meetod tagastab kasutajale pileti. Järjekorranumbrite haldus peaks olema automaadi seespeidus.

Realiseerida automaadis meetod **isValid()**, mis otsustab etteantud pileti korral, kas selle piletiga on kinos juba käidud.

Lua meetod **useTicket()**, mis käivitatakse saali sisenemisel ning märgib etteantud pileti kasutatuks. Kui piletiga on üks kord juba sisenetud, siis teist korda samapiletiga enam saali ei saa.

NB! Pileti „ärakasutamise“ info tuleb talletada selliselt, et kasutaja (väljaspoolklasse **Ticket/TicketMachine**) ei saaks pruugitud piletit teistkordselt kasutada.

Realiseerimisel mõelge läbi ligipääsud, et kasutaja ei saaks piletiautomaadiga manipuleerida. Paigutage klassid pakettidesse.

Nuputamist kogenud programmeerijatele: kuidas kõige efektiivsemalt vältida „isejoonistatud“ piletitega kinno pääsemine. Realiseerige vastav loogika.

Testimiseks koostada testklass, mis tekitab 2 automaati, väljastab neist mõned piletid ning kontrollib sisenemist ja piletite kehtivust.

Esmaseks testimiseks sobib järgmisel leheküljel toodud kood:

```
// Loo me mõned masinad:
TicketMachine m = new TicketMachine();
TicketMachine m2 = new TicketMachine();

// ja ostame portsu pileteid:
Ticket t1 = m.buyTicket("SherlockHolmes");
Ticket t2 = m2.buyTicket("Mission Impossible");
Ticket t3 = m.buyTicket("SherlockHolmes");
Ticket t4 = m.buyTicket("MissionImpossible");

// Testime, kas toString() toimib:
System.out.println("t1.toString() - " + t1);

// Kontrollime, et masinast ostetud piletid on kehtivad.
// Märgime, et teisest masinast ostetud piletid kehtivad ka esimeses masinas
System.out.println("isValid(t1) - " + m.isValid(t1)); // ->true
System.out.println("isValid(t2) - " + m.isValid(t2)); // ->true

// Kasutame pileti ära:
m.useTicket(t1); // -> teade: "Häid filmielamusi"
// pärast mida ei tohiks see kehtiv olla:
```

```
System.out.println("Pärast sisenemist: t1.isValid " + m.isValid(t1)); // ->false

// ning ükski automaat ei tohi seda enam aktsepteerida:
m2.useTicket(t1); // -> teade: "ALARM! Pilet ei ole kehtiv"

// Ise joonistatud piletiga ei tohiks olla kehtiv
Ticketfake = newTicket("Sherlock");
fake.number= 4;
System.out.println("fake.isValid() - " + m.isValid(fake) ); // ->false

// ning sellega ei pääse ka saali:
m.useTicket(fake); // -> teade: "ALARM! Pilet ei ole kehtiv"
```