

# Harjoitus 1, tehtävä 2

Kirjoitin rutiinien toteutukset uuteen luokkaan "ChessBoard", paketin "task2" sisälle. Luokka määrittelee printBoard() sekä createEmptyBoard() metodit, sekä niitä avustavia metodeja joita käytetään päämääräisesti luokan sisällä kahden päämetodin toteuttamista varten.

## EmptyChessPiece

Vakio, jolla kuvataan tyhjää shakkilaudan ruutua, eli "-" merkki. Voitaisiin lisätä nappula ChessPiece Enumiinkin, mutta ChessPiecen alkuperäisyyden säilyttämiseksi päätin sisällyttää sen vakiona ChessBoardin sisälle.

## printBoard()

Parametrit:

- Board
  - 2-ulotteinen vektorilista
  - Ei voi olla null
  - Tulee sisältää 8 vektoria
    - Jokaisen vektorin pituus tulee olla 8
    - Jokaisen vektorin arvon tulee olla shakkinappulan tekstiesitys, tai tyhjä nappula ( eli " - " merkki)
- usePaddingRows
  - Boolean arvo
  - Määrittää, tuleeko lautaan tulostaa tyhjät "|-----|" rivit alkuun ja loppuun
  - Ei voi olla null

Metodi ei palauta mitään, vaan tulostaa parametrien mukaisen laudan konsolille. Dokumentoinnin helpottamiseksi metodi on eristetty varsinaisesti toteutuksesta. Metodi hakee laudan tekstimuotoisen esityksen getBoardString() metodilla. Metodin parametrit annetaan suoraan getBoardString() metodille.

## getBoardString()

Parametrit: täsmälleen samat kuin printBoard() metodilla

Palauttaa parametrien mukaisen pelilaudan merkkijonoesityksenä:

- Tuloksessa tulee olla yhtä monta riviä kuin syötteessä
- Tuloksen jokaisen rivin tulee vastata jokaista syötteen rivivektoria, kun tuloksesta poistetaan "|" merkit
- Jokaisen tuloksen kirjaimen tulee olla shakkinappulan tekstiesitys, tai tyhjä nappula (eli " - " merkki)
- Jokaisen tuloksen rivin alussa ja lopussa tulee olla "|" merkki

Metodi muodostaa shakkilaudan annettujen parametrien perusteella ja palauttaa sen tekstiesityksenä.

Jos metodille syöttää "board" parametrin jonka pituus ei ole 8, metodi heittää InvalidParameterExceptionin, sillä laudassa tulee olla 8 vaakavektoria.

Jos jonkin syötteen vektorin pituus ei ole 8, metodi heittää InvalidParamsExceptionin, sillä jokaisella rivillä tulisi olla 8 nappulaa. Tyhjiä nappuloiden paikat tulisi täyttää "-" merkeillä, tai luokan EmptyChessPiece vakiolla. Jos vektorin pituus ei ole 8, lauta ei muodostu oikein. Tämän takia käytetään virheen heittämistä.

Jos usePaddingRows parametri saa true arvon, metodi lisää laudan alkuun ja loppuun tyhjän nappularivin ("|-----|").

Tekstiesitys rakennetaan StringBuilderilla käymällä board ja sen sisäiset rivivektorit läpi, ja lisäämällä jokainen kirjain (nappula) StringBuilderiin. Jokaisen rivivektorin alussa ja lopussa lisätään "|" kirjaimet, jolla saadaan lautaan "reunat". Rivivektorin lopussa lisätään myös uusirivi ("\n"), jolla katkaistaan nykyinen rivi.

StringBuilderia käytetään sen tehokkuuden vuoksi.

## createBoardRow()

Parametrit:

- chessPieces
  - Yksiulotteinen merkkijonovektori
  - Joko null, tai sisältää 8 merkkijonoa
  - Jos ei null, jokaisen merkkijonon tulee olla shakkinappulan tekstiesitys

Palauttaa rivin shakkinappuloita merkkijonovektorina:

- Vektorin pituus tulee olla 8
- Jos annettu vektori ei ollut null, palautetaan syötevektori
  - Annetun vektorin tulee täsmätä tulosvektoria
- Jos annettu vektori oli null, palautetaan uusi vektori, joka täytetään tyhjän nappulan arvoilla ("-")
  - Jokaisen tulosvektorin arvo tulee olla "-"

Metodi palauttaa annetun syötteen mukaisen rivin nappuloita vektoriesityksenä.

Jos syötevektori ei ole null, varmistetaan että sen pituus on 8 ja että jokainen nappula on tekstiesitys shakkinappulasta. Jos nämä ehdot eivät toteudu, heitetään `InvalidParameterException`. Tyhjien nappuloiden paikat tulisi täyttää "-" merkeillä, tai luokan `EmptyChessPiece` vakiolla. Jos vektorin pituus ei ole 8, lauta ei muodostu oikein. Tämän takia käytetään virheen heittämistä.

Jos ehdot täyttyvät, palautetaan syötevektori.

Jos syötevektori on null, luodaan uusi vektori käyttämällä `createEmptyBoardRow()` metodia, ja palautetaan se.

## createEmptyBoard()

Palauttaa shakkilaudan aloitustilanteessa 2-ulotteisena vektorilistana:

- Lista ei voi olla null
- Listan pituus tulee olla 8
- Jokainen listan vektori:
  - Ei voi olla null
  - Pituuden tulee olla 8
  - Rivien 1 & 8 tulee olla vektori: ("R", "N", "B", "Q", "K", "B", "N", "R")
  - Rivien 2 & 7 tulee olla vektori: ("P", "P", "P", "P", "P", "P", "P", "P")
  - Rivien 3,4,5,6 tulee olla vektori: (" ", " ", " ", " ", " ", " ", " ", " ")

Metodi luo uuden shakkilaudan pelin aloitustilassa ja palauttaa sen 2-ulotteisena vektorilistana. Jokainen rivi kuvaa yhtä riviä shakkiruudulla, ja jokaisen rivin alkio kuvaa yksittäistä shakkinappulaa rivillä.

Metodin @.post totuuslauseke tarkistaa, että palautettu lauta on oikean näköinen.

Dokumentoinnin helpottamiseksi käytin vektoreiden indeksivertailussa if-else shorthand -operaattoreita. Tehokkaampi ja mielekkäämpi tapa olisi käyttää Switch -rakennetta. Ehdon viimeinen "false" kuvaa tilannetta jossa loopin indeksi ei ole mikään annetuista riveistä. Ehto ei silti koskaan toteudu, sillä looppia ei käydä läpi jos vektorilistan pituus ei ole 8.

Metodi luo uuden kaksiulotteisen vektorilistan ja täyttää sen käyttäen apumetodeita createDefaultRoyalsBoardRow(), createDefaultPawnBoardRow() sekä createEmptyBoardRow(), ja lopuksi palauttaa muodostetun listan.

## createDefaultRoyalsBoardRow()

Palauttaa shakkilaudan kuninkaallisten rivin vektoriesityksenä:

- Vektori ei voi olla null
- Vektorin pituuden tulee olla 8
- Vektorin tulee täsmätä vektoria ("R", "N", "B", "Q", "K", "B", "N", "R")

## createDefaultPawnBoardRow()

Palauttaa shakkilaudan sotilaiden rivin vektoriesityksenä:

- Vektori ei voi olla null
- Vektorin pituuden tulee olla 8
- Vektorin tulee täsmätä vektoria ("P", "P", "P", "P", "P", "P", "P", "P")

## createEmptyBoardRow()

Palauttaa tyhjän shakkilaudan rivin vektoriesityksenä:

- Vektori ei voi olla null
- Vektorin pituuden tulee olla 8
- Vektorin tulee täsmätä vektoria (" ", " ", " ", " ", " ", " ", " ", " ")