

OPDRACHT: GOTTA CARD THEM ALL!

Er is een nieuwe rage op Intec!

Postkaarten van over heel de wereld, iedereen probeert minstens 1 postkaart per land te verzamelen.

Uiteraard ben jij één van de beste verzamelaars, want niet alleen heb je veel kaarten; jij beschikt ook over de nodige Java codering capaciteiten om je kaarten optimaal te verzamelen!

Vereisten:

Maak het Object 'PostCard' met variabele *country* (String), en variabele *continent* (String). Getters en setters niet vergeten, en optioneel voeg je een toString methode erbij.

Je maakt nog een Object 'Friend', met als variabelen: *name*(String), *family*(boolean), *yearsKnown*(int) en *friendShipLevel* (int). Getters en setters inclusief. Vergeet de equals/hashcode niet voor beide klassen!

Je plaats dit in je main methode

//your cards

```
PostCard belgium = new PostCard("Belgium", "Europe");
PostCard france = new PostCard("France", "Europe");
PostCard japan = new PostCard("Japan", "Asia");
PostCard rdc = new PostCard("Democratic Republic of Congo",
"Africa");
PostCard southKorea = new PostCard("South Korea", "Asia");
PostCard belgium2 = new PostCard("Belgium", "Europe");
PostCard southAfrica = new PostCard("South Africa", "Africa");
PostCard france2 = new PostCard("France", "Europe");
PostCard belgium3 = new PostCard("Belgium", "Europe");
PostCard usa = new PostCard("United States of America", "North
America");
PostCard canada = new PostCard("Canada", "North America");
PostCard peru = new PostCard("Peru", "South America");
PostCard samoa = new PostCard("Samoa", "Oceania");
PostCard belgium4 = new PostCard("Belgium", "Europe");
PostCard france3 = new PostCard("France", "Europe");
```

```
List<PostCard> yourPostCardList = new ArrayList();
yourPostCardList.add(belgium);
yourPostCardList.add(france);
yourPostCardList.add(japan);
yourPostCardList.add(rdc);
yourPostCardList.add(southKorea);
yourPostCardList.add(belgium2);
yourPostCardList.add(southAfrica);
yourPostCardList.add(france2);
yourPostCardList.add(belgium3);
yourPostCardList.add(usa);
yourPostCardList.add(canada);
yourPostCardList.add(peru);
yourPostCardList.add(samoa);
yourPostCardList.add(belgium4);
yourPostCardList.add(france3);
```

```
//your friend's cards
```

```
PostCard northKorea = new PostCard("North Korea", "Asia");
PostCard usa2 = new PostCard("United States of America", "North
America");
PostCard botswana = new PostCard("Botswana", "Africa");
PostCard northKorea2 = new PostCard("North Korea", "Asia");
```

```
List<PostCard> yourFriendsPostCardList = new ArrayList();
yourFriendsPostCardList.add(northKorea);
yourFriendsPostCardList.add(usa2);
yourFriendsPostCardList.add(botswana);
yourFriendsPostCardList.add(northKorea2);
```

```
//your friends
```

```
Friend bobby = new Friend("Bobby", false, 3, 5);
Friend melissa = new Friend("Melissa", false, 1, 6);
Friend darla = new Friend("Darla", true, 14, 2);
Friend bert = new Friend("Bert", false, 10, 4);
Friend grandma = new Friend("Nana", true, 12, 7);
Friend fester = new Friend("Fester", false, 1, 2);
Friend anna = new Friend("Anna", false, 8, 4);
```

Opdracht 1 :

Je beste vriend heeft een paar kaarten die je graag met hem zou willen ruilen. Hij vindt het goed, zolang je alleen maar dubbeltjes van hem vraagt.

Jij hebt een systeem voor alles dat je ruilt: graag plaats jij de nieuwe kaart op dezelfde plaats als de kaart dat je net weg ruilde. (INDEX!)

Jouw vriend kan het niet echt schelen, zolang hij maar zijn nieuwe kaart krijgt.

Zorg ervoor dat deze ruil gebeurt.

Opdracht 2 :

Je beslist toch je systeem aan te passen en wilt al je kaarten sorteren per land. Gebruik hiervoor een Comparator!

Print het resultaat uit.

Opdracht 3 :

Je systeem is nog niet optimaal genoeg, en je beslist je dubbeltjes uit je hoop kaarten te halen.

Maak een methode die kijkt hoeveel kaarten je van een bepaald land hebt, en de extras dan toevoegt aan een nieuwe List. Zorg ervoor dat de dubbele kaarten niet meer in de oude lijst zijn. Print dan uit hoeveel dubbele kaarten je hebt in de nieuwe lijst.

HINT: gebruik Collections.frequency();

HINT2: Een Set kan ook helpen;

Print dan het resultaat uit.

Opdracht 4 :

Je komt erachter dat kaarten verzamelen eigenlijk niet meer in de mode is en je beslist ze weg te geven aan al je vrienden.

Je hebt hier ook een systeem voor. Je wilt hier dit keer niet te veel code voor gebruiken, het gaat toch maar om kaarten die je niet meer gaat verzamelen.

Maak een PriorityQueue die 'sortedFriends' heet aan die aan de hand van lambdas eerst prioriteit geeft aan je familieleden, dan aan je beste vriend en dan pas aan de mensen die je het langste kent. Vul de queue dan aan met het volgende:

```
sortedFriends.offer(bobby);
sortedFriends.offer(melissa);
sortedFriends.offer(darla);
sortedFriends.offer(bert);
sortedFriends.offer(grandma);
sortedFriends.offer(fester);
sortedFriends.offer(anna);
```

Success!