

SiteBezoek

Klasse Tijd

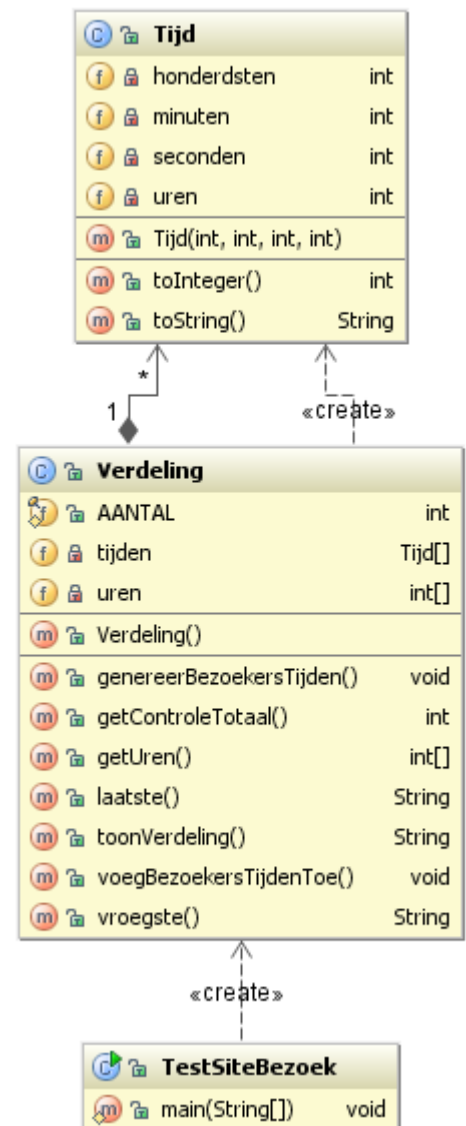
De klasse `Tijd` bevat 4 `int`-attributen met de namen `uren`, `minuten`, `seconden` en `honderdsten`. Verder is er nog een `private static Random` met de naam `random`, waar je meteen een nieuw `Random` object aan toekent.

- ✓ Voorzie een lege *default constructor* en een *constructor* om alle 4 de attributen een waarde te geven.
- ✓ Voorzie *getters* voor alle `int` attributen.
- ✓ Voorzie een methode `toInteger` waarin je de tijd omzet naar een `int`. Bv 2u 13min 54sec 15 honderdsten wordt het getal 2135415.
- ✓ Voorzie een methode `toString` waarin je de tijd in de vorm `x:xx:xx.xx` als een `String` teruggeeft.
- ✓ Voorzie een methode `maakRandomTijd` waarmee je de 4 attributen een willekeurige waarde kunt geven.

Klasse Verdeling

Deze klasse bevat een constante `AANTAL` met de waarde 1000. Verder zijn er de attributen `uren` en `tijden` die respectievelijk van het type `int[]` en `Tijd[]` zijn.

- ✓ Voorzie een *constructor* die aan `uren` een `int`-tabel voor 24 getallen (de 24 uren in een dag) toekent en aan `tijden` een `Tijd`-tabel met een lengte gelijk aan `AANTAL` toekent.
- ✓ Voorzie een *getter* voor de tabel `uren`, de returnwaarde is dus van het type `int[]`.
- ✓ Voorzie een methode `genereerBezoekersTijden` die aan elk element van `tijden` een nieuw `Tijd`-object toekent en daarna voor datzelfde element de methode `maakRandomTijd` van de klasse `Tijd` oproept.
- ✓ Voorzie een methode `voegBezoekersTijdenToe` waarmee het moment dat een bezoeker op de site komt geregistreerd wordt aan de hand van het uur.
- ✓ Voor alle elementen van de tabel `tijden` moet je de waarde van `uren` opvragen en die als index gebruiken om de aantallen in de `int`-tabel `uren` overeenkomstig te verhogen.
- ✓ Voorzie een methode `toonVerdeling` die de gegenereerde bezoekersaantallen voor elk uur in de vorm van een `String` teruggeeft. Zie afdruk bij `TestSiteBezoek`.
- ✓ Voorzie een methode `getControleTotaal` die een `int`-waarde terug met de som van alle aantallen in de tabel `uren`.
- ✓ Voorzie een methode `vroegste` die de tijd van de vroegste bezoeker in de vorm van een `String` teruggeeft.
- ✓ Voorzie een methode `laatste` die de tijd van de laatste bezoeker in de vorm van een `String` teruggeeft.



Klasse TestSiteBezoek

Gebruik deze klasse om je programma te testen. Zie ook een mogelijke uitvoer hieronder.

```
public class TestSiteBezoek {
    public static void main(String[] args) {
        Verdeling verdeling = new Verdeling();
        verdeling.genereerBezoekersTijden();
        verdeling.voegBezoekersTijdenToe();

        System.out.println("Verdeling:");
        System.out.println(verdeling.toonVerdeling());

        System.out.println("Totaal aantal bezoekers: " +
        verdeling.getControleTotaal());
        System.out.printf("Eerste om %-12s\n", verdeling.vroegste());
        System.out.printf("Laatste om %-11s\n ", verdeling.laatste());
    }
}
```

Verwachte uitvoer:

Verdeling:

uur 0: 37	uur 12: 45
uur 1: 47	uur 13: 39
uur 2: 41	uur 14: 42
uur 3: 46	uur 15: 51
uur 4: 48	uur 16: 31
uur 5: 38	uur 17: 44
uur 6: 34	uur 18: 35
uur 7: 34	uur 19: 41
uur 8: 50	uur 20: 49
uur 9: 39	uur 21: 52
uur 10: 36	uur 22: 42
uur 11: 36	uur 23: 43

Totaal aantal bezoekers: 1000

Eerste om 0:00:23.22

Laatste om 23:59:28.89