

Verhaal Festival

Je wordt gevraagd om het domain model uit te tekenen met behulp van een UML klassendiagram en te implementeren in Java voor een applicatie met betrekking tot een nieuw festivalconcept. Gebruik in je oplossing zoveel mogelijk OO concepten die in de cursus OO programmeren tot nu toe aan bod gekomen zijn (overerving, polymorfie).

Het ontwerp dat we gaan maken is een vereenvoudigd model. Hou je in je oplossing aan de onderstaande tekst!! Voor de echte applicatie zou het domain model veel uitgebreider zijn.

Een Festival heeft een naam, startdatum en einddatum (LocalDate), en houdt (voorlopig enkel) gegevens bij over de optredens en werknemers. Voor deze opgave werken we enkel het gedeelte met betrekking tot de werknemers uit. Alle velden zijn verplicht, opvraagbaar en niet wijzigbaar. Bovendien mag de einddatum niet voor de begindatum liggen. Je moet een tekstuele vorm van alle werknemers van het festival kunnen weergeven.

Voor een werknemer worden naam, voornaam en rijksregisternummer bijgehouden. Voor het rijksregisternummer checken we of het volgende formaat werd gerespecteerd : xx.xx.xx-xxx.xx. Wanneer een nieuwe werknemer wordt aangemaakt worden alle eigenschappen meegegeven. Alle velden zijn verplicht, opvraagbaar, maar niet wijzigbaar.

Voor elke werknemer moet een verloning kunnen worden berekend. Er zijn werknemers voor wie deze verloning berekend wordt op basis van het aantal gewerkte uren waar hij nog niet voor betaald werd. Het aantal gewerkte uren is voor dit soort werknemers bij aanvang nul en kan worden opgehoogd met een positief aantal. Het aantal uren gewerkt kan ook worden gerest op nul. Er zijn (voorlopig) twee van deze soort werknemers: werknemers die een verloning onder de vorm van geld krijgen (noem deze Loonwerknemer; zij werken aan 10€ per gewerkt uur) en werknemers die een aantal drankbonnetjes krijgen per gewerkt uur (noem deze Vrijwilliger; in dit model 2 bonnetjes per uur). Er zijn ook werknemers voor wie de verloning een vast bedrag is op basis van één specifieke geleverde prestatie (bijvoorbeeld "opbouw lichtinstallatie main stage" aan 12235€). Deze extra eigenschappen worden meegegeven in de constructor (prestatie mag niet leeg zijn, minimaal bedrag is 20 euro). Voor elke werknemer moet een tekstuele vorm kunnen weergegeven worden die de soort (gebruik hiervoor `this.getClass().getSimpleName()`) en alle eigenschappen bevat (bijvoorbeeld

Vrijwilliger: Jan Janssen (96.04.17-989.44)
Verloning: 30
Aantal uren gewerkt (nog niet betaald): 15

Welke functionaliteiten worden verwacht van de nieuwe applicatie:

- Werknemers kunnen toevoegen
- Het totaal aantal bonnetjes berekenen dat je nog moet voorzien voor de nog niet uitbetaalde uren van de vrijwillige werknemers.
- Het gemiddelde bedrag dat aan verloning voor een "prestatie" werd betaald.

- Een lijst van alle prestaties van de laatst genoemde werknemers (werknemers die voor een prestatie worden verloond)
- Tekstuele vorm met naam, start- en einddatum en info over alle werknemers van het festival.

Schrijf een Appeke om de functionaliteiten te testen. Schrijf ook voor minstens 1 soort werknemer een volledig uitgewerkte testklasse.