

Overzichtsoefening

Course.js

Maak de Javascript klasse Course in het bestand src/school/Course.js.

Course heeft eigenschappen `_id`, `_grade` en `_completed`.

De constructor heeft als argument een waarde voor de eigenschap `_id`. `_grade` wordt gelijk gesteld aan 0 en `_completed` wordt gelijk gesteld aan false.

Voorzie getters en setter voor alle eigenschappen.

Je mag er vanuit gaan dat `_grade` altijd een geheel getal bevat tussen 0 en 20.

Student.js

Maak de Javascript klasse Student in het bestand src/school/Student.js en

Student heeft eigenschappen `_id` en `_courses`.

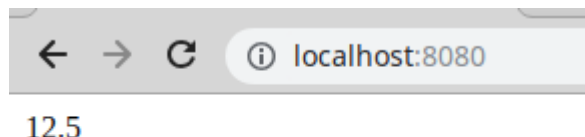
De constructor heeft als argument een waarde voor de eigenschap `id`. `_courses` wordt gelijk gesteld aan een lege array.

Voorzie de function `addCourse`. Deze functie verwacht een Course-object als argument. Als het argument geen Course is wordt een Error opgeworpen (je kan dit controleren via `instanceof`). Indien het argument wel een Course is, wordt de course toegevoegd aan de array `_courses` (een element toevoegen aan een array kan via `push`).

Maak de function `calculateGrade`. In deze functie worden alle elementen uit `_courses` doorlopen. Per completed course wordt de waarde van de grade opgeteld bij de som. De totale eindscore (grade) die teruggekeerd wordt, is de som gedeeld door het aantal cursussen die completed zijn. Wanneer het aantal completed courses echter gelijk is aan 0, wordt een error opgeworpen.

Via `index.html` en `app.js` kan je de toepassing testen.

De output is van de vorm:



Student.test.js

Voorzie 2 (Jest) tests in het bestand `test/js/school/Student.test.js`.

(1) De functie `calculateGrade` moet een error opwerpen wanneer hij aangeroepen wordt voor een student-object die nog geen `completed course` heeft.

(2) De functie `calculateGrade` moet 12 als terugkeerwaarde hebben voor een student die één (`completed`) course heeft met grade 12.