

Лекция 14

Референции. Капсулация на данни.

Задача за домашно:

Задача 1: Да се създадат класовете Task и Employee, **като се спазва принципа за капсулация на данните**.

Класът Task представя някакво количество работа(задача) което трябва да бъде свършено.

Класът да има следните полета:

- име

- workingHours – числов тип, който показва колко часа остават на задачата за да бъде изпълнена

Класът да има конструктор който задава валидни стойности на 2-те полета.

Да се направят подходящи методи за достъпване на полетата (setters and getters).

Да се създаде клас Employee репрезентиращ работник.

Класът да има следните полета:

- name – име на работника

- currentTask – обект от тип Task, който показва текущата задача в/у която работи работника

- hoursLeft – числова стойност, която показва колко работни часа остават на работника (за днес)

Класът да има конструктор с 1 параметър, който инициализира полето name.

Да се направят подходящи getter-и и setter-и за полетата на класа (setter-ите да проверяват дали данните са валидни - името да не празно, часовете да са положително число...).

Да се направи метод без параметри work() на класа Employee. С извикването на този метод, работника за който е извикан метода работи по текущата си задача (ако има такава). Например ако работника има останали 4 часа за работа, а по задачата има останали 7 часа работа, след изпълнението на метода работника ще е останал с 0 часа работа за деня, а по задачата ще е останало 3 часа работа. Ако

работника има 8 часа, след изпълнението на метода `work()` работника ще е останал с 1 час работа за деня, а времето което остава на задачата да бъде свършена ще е 0.

Да се направи и метод `showReport`, който се извиква след като работника поработи в/у текущата си задача (извикване на `work`) и показва информация (принтирайки на конзолата) за:

- името на работника
- името на задачата
- работните часове които остават на работника
- часовете които остават на текущата задача на работника за да бъде изпълнена

Да се направи и клас с `main` метод, който използва функционалността на класовете `Task` и `Employee`.