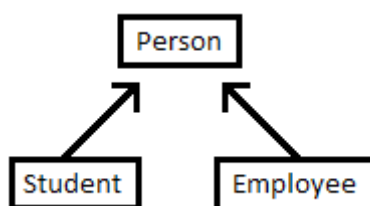


## Наследяване на класове. Идентификатори за достъп. Използване на ключова дума `super`.

### Задача за домашно:

Да се направи иерархия от класовете **Person**, **Student** и **Employee**



Класът **Person** репрезентиращ човек да има следните  
Поleta:

- `name` - име
- `age` - години
- `isMale` - поле, показващо пола

Конструктори:

- Единствен конструктор, с параметри за всичките полета на класа

Метод:

- Метод `showPersonInfo`, който показва информация за човека (изписва на конзолата стойността на всичките му полета по подходящ начин)

Класът **Student** наследяващ **Person**, репрезентиращ ученик да има  
Поleta:

- `score` - показва оценката на ученика (число между 2 и 6, не е задължително да е цяло)

Единствен конструктор:

- `Student(String name, int age, boolean isMale, double score)`  
задаващ стойности и на наследените полета

Метод `showStudentInfo()` който показва информация за човека и информация за оценката му (чрез надписи на екрана)

Класът Employee да наследява Person и да дефинира:

Полета:

–`daySalary` показва дневната заплата на работника

Конструктор който инициализира всичките му полета (собствени и наследени)

Методи:

В случай че на работник се наложи да работи извън работно време, на него му се дължи допълнителна сума пари (overtime) за часовете които е работил. Сумата се пресмята по следния начин: ако работника, няма навършени 18 години, дължимата сума е 0. В противен случай, за всеки отработен час, на работника се заплаща сумата която получава на час (зависи от дневната заплата) умножена по 1,5.

–метод `calculateOvertime(double hours)` който пресмята и връща стойността на сумата която му се дължи при работа извън работно време

–метод `showEmployeeInfo()`, който показва информация за човека, както и информация за дневната му заплата

Да се напише и демонстративен клас с `main` метод в който:

1. Да се състави масив от 10 елемента от тип `Person`.
2. Да се създадът по два обекта от тип `Person`, `Student` и `Employee` и да се вкарат в масива от хора.
3. Да се обходим масива и в зависимост от това дали обекта е от тип `Person`, `Student` или `Employee` да се извика съответно метода `showPersonInfo`, `showStudentInfo` ИЛИ `showEmployeeInfo`.
4. Да се обходи още веднъж масива с хора и за всеки човек, който е работник, да се изведе на екрана сумата, която му се дължи при 2 часа работа извън работно време.

**При писането на класовете, ДА СЕ СПАЗИ ПРИЦИПА ЗА КАПСУЛАЦИЯ НА ДАННИ.**

И един въпрос, какво ще стане ако дефинирате конструктор без параметри `Student()` в класа `Student` ??? Възможно ли е, и ако не защо. Как може да се оправи това?