

Készíts egy *tax* nevű csomagot, majd készítsd el az alábbi osztályokat és interfészt!

Készíts egy *Employee* osztályt, amelyben legyen

- *name* szöveges mező
- *salaries* nevű, 12 darab egész számot tartalmazó mező
- konstruktor, amely paraméterként kapja a nevet (*name*)
- készíts getter és setter metódusokat, amelyek által a *name* csak olvasható, a *salaries* tömb elemei pedig írható és olvashatók lesznek (nem maga a tömb!).

Készíts egy *Taxpayer* nevű interfészt, amely az alábbi metódusokat tartalmazza:

- *getName()*, ami egy szöveget ad vissza
- *getYearlyIncome()*, ami egy számot ad vissza
- *getYearlyExpenditure()*, ami egy számot ad vissza

Az *Employee* osztály valósítsa meg ezt az interfészt úgy, hogy a dolgozó éves bevétele (*yearly income*) a fizetési összege, éves költsége (*yearly expenditure*) pedig nincsen.

Készíts egy *Company* osztályt, amelyben legyen

- *name* szöveges mező
- *income* egész szám mező
- *expenditure* egész szám mező
- konstruktor, amely paraméterként kapja a nevet (*name*)
- *sellProduct()* metódus, ami a paraméterként átadott összeggel növeli a bevételt (*income*)
- *buyTool()* metódus, ami a paraméterként átadott összeggel növeli a költséget (*expenditure*)

A *Company* osztály is valósítsa meg a *Taxpayer* interfészt. A vállalat éves bevétele/költsége legyen a bevétel/költség mező aktuális értéke.

Készíts egy *TaxCalculator* nevű osztályt, amelyben legyen

- *taxRate* nevű privát statikus, lebegőpontos számot tartalmazó mező, aminek kezdeti értéke legyen 0.1
- *changeTaxRate()* nevű publikus statikus metódus, ami beállítja az adókulcsot (*taxRate*) a paraméterként átadott értékre
- *calculateTax()* nevű publikus statikus metódus, amely paraméterként egy *Taxpayer* referenciát vár, visszatérési értéke pedig ennek adója:
 $\max(0, (\text{éves bevétel} - \text{éves költség}) * \text{adókulcs})$

Készíts egy *Manager* osztályt, ami az *Employee* osztály leszármazottja, és

- van egy *yearlyPremium* mezője, ami a konstruktorban kap értéket (a név mellett)
- az éves bevétel számításnál a fizetések összegéhez hozzá kell adni az éves prémiumot is (*yearlyPremium*)

Készíts egy főprogram osztályt, amely tartalmazza az alábbi statikus metódusokat, és main() metódusában meghívja azokat:

- *generateTestData()*: létrehoz egy *Taxpayer* tömböt, amelyet feltölt *Employee*, *Manager*, illetve *Company* objektumokkal (azokat is feltöltve a szükséges fizetési/bevételi/költség adatokkal), és visszaadja azt.
- *yearlyReport()*: paraméterként egy *Taxpayer* objektumokat tartalmazó tömböt kap, és egy szövegben összefűzve visszaadja az adóalanyok összesített bevételeit, költségét, illetve az összesített adó mértékét (számoljunk 0.19 adókulccsal). Ezután listázza ki az *Employee* típusú adóalanyok nevét és fizetéseit havi bontásban.

Az alábbihoz hasonló eredményt várunk:

```
Yearly report:
Total income: 301000
Total expenditure: 7000
Total tax paid: 55860,000

Employee salaries by month:
Employee name: John Snow
> 1. month: 10000
> 2. month: 10000
> 3. month: 10000
> 4. month: 10000
> 5. month: 10000
> 6. month: 10000
> 7. month: 10000
> 8. month: 10000
> 9. month: 10000
>10. month: 10000
>11. month: 10000
>12. month: 10000
Employee name: Daenerys Targaryen
> 1. month: 12000
> 2. month: 12000
> 3. month: 12000
> 4. month: 12000
> 5. month: 12000
> 6. month: 12000
> 7. month: 12000
> 8. month: 12000
> 9. month: 12000
>10. month: 12000
>11. month: 12000
>12. month: 12000
```