

Listy 3

Zadanie 1 - zwierzyniec

Stwórz klasę `Animal`, która będzie zawierała **nazwę**, **wydawany dźwięk** i **flagę** mówiącą, czy dane zwierze jest głośne. Ponadto, klasa ma zawierać metodę `void speak(int times)`, która wypisze dźwięk zwierzęcie `times` razy. Jeśli zwierze jest głośne, dźwięk ma być pisany wielkimi literami. W przeciwnym przypadku - małymi. Następnie w metodzie `main` stwórz listę zwierząt i wywołaj metodę `speak` dla wszystkich głośnych zwierząt. Potem wywołaj tę metodę dla wszystkich cichych zwierząt. Zwierzęta mają dawać głos od 1 do 5 razy (losowo). Przykładowy output:

```
Loud animals
Big dog : WOOF
Cow : MOO MOO MOO MOO
Silent animals
Small dog : woof woof
Frog : croak croak croak croak
```

Zadanie 2 - firma

Stwórz klasę `Employee`, która będzie zawierała **id**, **pensję** oraz **stanowisko** pracownika. Konstruktor tej klasy powinien przydzielać kolejno tworzonej obiektom id¹. Przeciąż metodę `toString` w tej klasie, by formatowała informacje o pracowniku jak w przykładowym outputcie. Następnie w metodzie `main`:

1. Stwórz listę pracowników i dodaj kilku przykładowych (w tym informatyków). Wypisz wszystkich, wykorzystując metodę `toString`.
2. Wyznacz średnią roczną pensję programistów.
3. Stwórz listę przechowującą id pracowników, którzy zarabiają powyżej 7000 i wypisz wszystkie id z tej listy.

```
(1) (Programmer): 12000
(2) (Secretary): 6500
(3) (Programmer): 13500
(4) (Programmer): 6000
(5) (Programmer): 8000
Mean annual salary for programmers is: 118500.00.
Employees who earn more than 7k: 1, 3, 5.
```

Zadanie 3 - asd

Output:

```
asd
```

Uwagi

1. Należy zadbać o podstawowe zasady programowania obiektowego - enkapsulację, gettery, settery, odpowiednią dostępność zmiennych... Te rzeczy nie są uwzględnione w treści zadania, ale są to dobre praktyki, które należy stosować.

¹Pierwszy obiekt - id 1, drugi obiekt - id 2, itd...