

Scenariusz 2

Cel ćwiczenia: Zrozumienie podstaw javy.

Zadania do wykonania:

Utworzyć program w języku JAVA, który będzie zawierał przynajmniej:

- 5 różnych typów danych (spośród: byte, boolean, char, float, double, int, long, short, void)
- Instrukcję warunkową (if - else)
- Instrukcję wyboru
- Dowolną pętlę
- Tablicę jednowymiarową oraz tablicę dwuwymiarową

Mnie przypadł temat wypożyczalni aut.

W programie są 2 klasy:

- Auto

```
class Auto{
    String rejestracja;
    short rok;
    int przebieg;
    boolean czyWypozyczone;
    String model;
    double silnik;

    Auto(String rej, short r, int prz, boolean czy, String mod, double sil ){
        rejestracja=rej;
        rok=r;
        przebieg=prz;
        czyWypozyczone=czy;
        model=mod;
        silnik=sil;
    }

    void wypisz()
    {
        System.out.println(rejestracja +"\t"+ rok+"\t"+ przebieg+"\t"+ czyWypozyczone+"\t"+ model+"\t"+ silnik +"\n");
    }
}
```

- Wypożyczalnia

```
public class Wypożyczalnia {  
    public static void main(String args[]) {  
        ArrayList<Auto> zbior1= new ArrayList<Auto>();  
        int [][]ileaut=new int[1][1];  
        int pom = 0;  
        do {  
            System.out.println("Menu: \n1 - Dodaj auto\n2 - Wyświetl auta\n3 - Usuń auto\n4 - Wyjście");  
  
            switch (pom) {  
                case 1:  
                    short rok=2014;  
                    zbior1.add(new Auto( rej: "sdfsdf",rok, prz: 45484, czy: true, mod: "sdf", sil: 2.2));  
                    ileaut[1][1]++;  
                    break;  
                case 2:  
                    Iterator <Auto> ite = zbior1.iterator();  
                    for(int i=0; i<zbior1.size();i++)  
                        ite.next().wypisz();  
                    break;  
                case 3:  
                    zbior1.remove( index: ileaut[1][1]-1);  
                    ileaut[1][1]--;  
                    break;  
                case 4:  
                    System.out.println("Zamykanie");  
                    break;  
            }  
        }  
    }  
}
```

Zostało stworzone proste menu do operacji na autach w wypożyczalni.

Menu:

```
1 - Dodaj auto  
2 - Wyświetl auta  
3 - Usuń auto  
4 - Wyjście  
1
```

Menu:

```
1 - Dodaj auto  
2 - Wyświetl auta  
3 - Usuń auto  
4 - Wyjście
```

Dodaj auto, dodaje auto bez wprowadzania jakichkolwiek danych, wyświetl auta, wyświetla wszystkie auta w wypożyczalni, usuń auto, usuwa ostatnie auto i sprawdza czy są jakieś auta w wypożyczalni, jeśli nie to wychodzi z programu.

Całość jest w miarę prosta do wykonania, w Javie wszystko jest klasą, dlatego Wypożyczalnia to klasa.

Można jeszcze rozbudować program o wprowadzenie danych z klawiatury i wczytywanie z pliku.

```
Menu:
1 - Dodaj auto
2 - Wyświetl auta
3 - Usuń auto
4 - Wyjście
2
sdfsdf 2014 45484 true sdf 2.2

sdfsdf 2014 45484 true sdf 2.2

Menu:
1 - Dodaj auto
2 - Wyświetl auta
3 - Usuń auto
4 - Wyjście
3
Brak aut, koniec programu!
```

Listing kodu:

```
import java.lang.reflect.Array;
```

```
import java.util.*;
```

```
import java.lang.Math.*;
```

```
class Auto{
```

```
    String rejestracja;
```

```
    short rok;
```

```
    int przebieg;
```

```
    boolean czyWypozyzione;
```

```
    String model;
```

```
    double silnik;
```

```
    Auto(String rej, short r, int prz, boolean czy, String mod, double sil ){
```

```
        rejestracja=rej;
```

```
        rok=r;
```

```
        przebieg=prz;
```

```
        czyWypozyzione=czy;
```

```
        model=mod;
```

```
        silnik=sil;
```

```
    }
```

```
    void wypisz()
```

```
    {
```

```
        System.out.println(rejestracja + "\t" + rok + "\t" + przebieg + "\t" + czyWypozyzione + "\t" +  
model + "\t" + silnik + "\n");
```

```
    }
```

```
}
```

```
public class Wypożyczalnia {
```

```
    public static void main(String args[]) {
```

```
        ArrayList<Auto> zbior1= new ArrayList<Auto>();
```

```
        int [][] ileaut=new int [1] [1] ;
```

;

int pom = 0;

do {

System.out.println("Menu: \n1 - Dodaj auto\n2 - Wyświetl auto\n3 - Usuń auto\n4 - Wyjście");

switch (pom) {

case 1:

short rok=2014;

zbior1.add(new Auto("sdfsdf",rok,45484,true,"sdf",2.2));

ileaut [1] [1] ++;

break;

case 2:

Iterator <Auto> ite = zbior1.iterator();

for(**int** i=0; i<zbior1.size();i++)

ite.next().wypisz();

break;

case 3:

zbior1.remove(ileaut [1] [1] -1);

ileaut [1] [1] --;

break;

case 4:

System.out.println("Zamykanie");

break;

default:

System.out.println("Nie ma takiej opcji");

}

```
    if(ileaut==0)
    {
        System.out.println("Brak aut, koniec programu!");
        break;
    }
    else{
        continue;
    }
} while (pom != 4);
}
}
```