

XII. Java po C++.

Pakiety, interfejsy i obsługa wyjątków w Javie

1. Pakiety

1.1. Pakiet to swoisty pojemnik na klasy, używany do odpowiedniego podziału przestrzeni nazw oraz sterowania dostępem do tych nazw.

1.2. Tworzenie pakietu odbywa się poprzez umieszczenie instrukcji `package` jako pierwszej instrukcji w pliku źródłowym Javy:

```
package nazwa_pakietu
```

Wszystkie klasy zdefiniowane w pliku zostaną przypisane do pakietu wymienionego w instrukcji. Pominięcie instrukcji `package` jest równoznaczne z przypisaniem klasy do domyślnej przestrzeni nazw.

1.3. Do przechowywania pakietów służy system plików. Każdy plik o rozszerzeniach `.java` i `.class`, który został utworzony dla pakietu o nazwie *nazwa_pakietu*, będzie umieszczony w katalogu o nazwie *nazwa_pakietu*.

1.4. Pakiety można składać w wielopoziomowe hierarchie; w nazwie pakietu poszczególne poziomy oddziela się znakiem kropki, np.

```
package java.awt.image; /* Abstract Window Toolkit */
```

Wyszukiwanie żądanych pakietów odbywa się poprzez przeszukiwanie katalogu bieżącego oraz katalogów zadanych przez zmienną środowiskową `CLASSPATH`.

1.5. Oto przykład prostego pakietu:

```
package MyPack;
class Balance {
    String name;
    double bal;
    Balance(String n, double b) {
        name = n;
        bal = b;
    }
    void show() {
        if(bal<0)
            System.out.print("-->> ");
            System.out.println
                (name + ": " + bal + "zł");
    }
}
class AccountBalance {
    public static void main(String args[]) {
        Balance current[] = new Balance[3];
        current[0] =
            new Balance("Romeo i Julia",151.30);
        current[1] =
            new Balance("Grona gniewu", 1142.05);
        current[2] =
            new Balance("Mercedes-benz",-28.50);
        for(int i=0;i<3;i++) current[i].show();
    }
}
```

Plik źródłowy, o nazwie `AccountBalance.java` zostanie umieszczony w katalogu `MyPack`. Uruchomienie programu nastąpi na podstawie polecenia:

```
java MyPack.AccountBalance
```

w którym nazwa klasy musi być koniecznie poprzedzona nazwą pakietu.

1.6. Pakiety, współdziałając z klasami, tworzą nowe kategorie widoczności składowych klas. Możliwe poziomy dostępu podsumowuje następująca tabela:

	private	brak	protected	public
ta sama klasa	tak	tak	tak	tak
ten sam pakiet, podklasa	nie	tak	tak	tak
ten sam pakiet, inna klasa	nie	tak	tak	tak
inny pakiet, podklasa	nie	nie	tak	tak
inny pakiet, inna klasa	nie	nie	nie	tak

1.7. Za pomocą instrukcji `import` można uwidocznić w programie konkretne klasy lub całe pakiety. Używanie tej instrukcji jest konieczne także w stosunku do wszystkich wbudowanych klas Javy, z wyjątkiem tych umieszczonych w pakiecie `java.lang`. Ogólna postać instrukcji `import` to:

```
import pakiet1[.pakiet2... .pakietn].(nazwa_klasy | *);
```

Należy ją umieszczać bezpośrednio po instrukcji `package`, przed definicjami klas.

1.8. Gdyby zmienić w zdefiniowanym wyżej pakiecie `MyPack` wybrane modyfikatory dostępu na publiczne:

```
package MyPack;
public class Balance {
    String name;
    double bal;
    public Balance(String n, double b) {
        name = n;
        bal = b;
    }
    public void show() {
        if(bal<0)
            System.out.print("-->> ");
            System.out.println
                (name + ": " + bal + "zł");
    }
}
```

to będzie można użyć metod `Balance` i `show` w kontekście jak poniżej:

```
import MyPack.*;
class TestBalance {
    public static void main(String args[]) {
        Balance test =
            new Balance("Program. w C++", 2451.00);
        test.show();
    }
}
```