

## 1 Pakiety

Pakiety są wykorzystywane do grupowania klas w hierarchiczne grupy. Zorganizowane są w nazwy reprezentowane przez adresy URL.

```
com.pakiet.podpakiet.NazwaKlasy
```

Przynależność klasy do pakietu reprezentowana jest za pomocą słowa kluczowego `package`. Występuje ono na początku pliku źródłowego z klasą:

```
package com.pakiet.podpakiet
```

Kolejne nazwy oddzielone kropkami w pakiecie określają katalogi w których zlokalizowana jest dana klasa

```
katalog/com/pakiet/podpakiet/NazwaKlasy.class
```

Użycie klasy z pakietu reprezentowane jest za pomocą słowa kluczowego `import`

```
import com.pakiet.podpakiet.NazwaKlasy
```

Powyższe polecenie używa klasy `NazwaKlasy` z pakietu `com.pakiet.podpakiet`

```
import com.pakiet.podpakiet.*
```

Użycie wszystkich klas z pakietu `com.pakiet.pakiet`

## 2 Kontrola widoczności elementu

Zarówno atrybuty jak i metody mogą posiadać atrybuty, które określają zasięg widoczności.

- `public` – elementy publiczne, widoczne z każdej klasy
- `private` – elementy widoczne tylko w aktualnej klasie
- `protected` – elementy widoczne w aktualnej klasie i klasach pochodnych
- bez atrybutu – dostęp pakietowy, w obrębie tego samego pakietu

### 3 Inne atrybuty

- static – element statyczny. Związany z klasą a nie z obiektem
- final – element którego wartości nie można zmienić

### 4 Zadania do wykonania

UWAGA: Zadania z instrukcji należy wykonać bez wykorzystania IDE!

1. Przygotować przykładową strukturę pakietów (minimum 3 poziomy)
2. Uruchomić program z określonego pakietu
3. Uruchomić program z dowolnego miejsca na dysku
4. Przetestować działanie modyfikatorów dostępności dla pól, metod i konstruktorów
5. Przetestować działanie modyfikatorów dostępności z innego pakietu
6. Przetestować działanie modyfikatorów dostępności z klasy pochodnej
7. Spróbować odwołać się do prywatnych elementów obiektu tej samej klasy
8. Zastosować modyfikatory dostępności dla klas
9. Stworzyć i przetestować dostęp do pól i metod statycznych
10. Sprawdzić możliwości inicjalizacji pól finalnych
11. Odwołać się do klasy z innej lokalizacji na dysku