

1. Dziedziczenie i kompozycja.

Dziedziczenie jest jedną z fundamentalnych cech podejścia obiektowego. Pozwala kojarzyć klasy obiektów w *hierarchie klas*. Te hierarchie, w zależności od użytego języka programowania, mogą przyjmować postać drzew, lasów drzew, bądź skierowanych grafów acyklicznych. W Javie hierarchia dziedziczenia dla klas ma postać drzewa. Jej korzeniem jest klasa *Object*. Realizacja dziedziczenia polega na tym, że klasa dziedzicząca dziedziczy po swojej nadklasie wszystkie jej atrybuty i metody (i nie ma znaczenia, czy te atrybuty i metody były zadeklarowane bezpośrednio w tej nadklasie, czy ona też odziedziczyła je po swojej z kolei nadklasie).

Dziedziczenie odzwierciedla relację *is-a* (*jest czymś*). Oznacza to, że każdy obiekt podklasy jest także obiektem nadklasy. Na przykład hierarchia klas zbudowana z nadklasy *Owoc* i dwu podklas *Jabłko* i *Gruszka* jest prawidłowo zbudowana, bo każde jabłko i każda gruszka jest owocem. Niestety często relacja *is-a* jest mylona z relacją *has-a* (*ma coś*) dotyczącą składania obiektów.

```
class Samochod {  
    }  
class Limuzyna extends Samochod {  
    }  
}
```

Każda klasa automatycznie dziedziczy po klasie *Object*.

```
class Samochod extends Object {  
    }  
}
```

Kompozycje wyraża relację „składa się z” lub „posiada” (*has-a*). Tworzona klasa składać się może z dowolnej liczby obiektów, tworzonych na podstawie istniejących już klas.

```
class Kolo {  
    }  
class Samochod {  
    Kolo[] kolo;  
}
```

- brak wielodziedziczenia
- `this` – wskazanie na aktualny obiekt
- `super` – wskazanie na obiekt klasy bazowej (nadrzędnej)

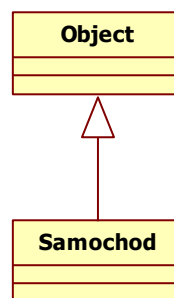
Klasa Object:

- `.clone()` – tworzy i zwraca kopię aktualnego obiektu
- `.equals(Object)` – sprawdza, czy dany obiekt jest równy aktualnemu
- `.finalize()` – metoda wywoływana przez Garbage Collector, gdy nie istnieje żadna referencja do aktualnego obiektu
- `.getClass()` – zwraca klasę reprezentującą obiekt
- `.hashCode()` – zwraca wartość reprezentującą obiekt dla metody mieszającej
- `.toString()` – zwraca reprezentację tekstową obiektu

2. Rzutowanie obiektów.

- rzutowanie w górę (ang. upcasting) – bezpieczne
- rzutowanie w dół (ang. downcasting) – wymaga testowania

```
Object object = (Object) new Samochod(); // rzutowanie w górę
    if (object instanceof Samochod) { // sprawdzenie przed rzut.
        Samochod samochod = (Samochod) object; // rzutowanie w dół
    }
```



– *instanceof* – sprawdzenie, czy obiekt jest danej klasy

Zadanie:

Zaimplementować aplikację wykorzystującą kompozycję i dziedziczenie. Wykorzystać w aplikacji `this` oraz `super`.