

Programowanie obiektowe w Javie - dziedziczenie, polimorfizm, klasy abstrakcyjne

1. Utwórz klasę `Pojazd` z polami `marka`, `model` i `rokProdukcji`. Utwórz klasy `Samochod` i `Motocykl`, które dziedziczą po klasie `Pojazd`. Klasa `Samochod` powinna mieć dodatkowe pole `liczbaDrzwi`, a klasa `Motocykl` pole `pojemnoscSilnika`. Dodaj konstruktory, metody gettery i settery, metodę `toString()` oraz `equals()` dla każdej z klas. Napisz program testujący zdefiniowane klasy i metody.
2. Utwórz klasę `Pracownik` z polami `imie`, `nazwisko` i `placa`. Utwórz klasy `Programista` i `Tester`, które dziedziczą po klasie `Pracownik`. Klasa `Programista` powinna mieć dodatkowe pole `jezykProgramowania`, a klasa `Tester` pole `typTestowania`. Dodaj konstruktory, metody gettery i settery, metodę `toString()` oraz `equals()` dla każdej z klas. Napisz program testujący zdefiniowane klasy i metody.
3. Utwórz klasę `Nieruchomosc` z polami `adres`, `metraż` i `cena`. Utwórz klasy `Dom` i `Mieszkanie`, które dziedziczą po klasie `Nieruchomosc`. Klasa `Dom` powinna mieć dodatkowe pole `liczbaPieter`, a klasa `Mieszkanie` pole `numerPietra`. Dodaj konstruktory, metody gettery i settery, metodę `toString()` oraz `equals()` dla każdej z klas. Napisz program testujący zdefiniowane klasy i metody.
4. Utwórz klasę `GraPlanszowa` z polami `nazwaGry`, `minLiczbaGraczy`, `maxLiczbaGraczy` oraz `zasadyGry` (jako `ArrayList` typu `String`). Utwórz klasy `GraEdukacyjna` i `GraStrategiczna`, które dziedziczą po klasie `GraPlanszowa`. Klasa `GraEdukacyjna` powinna mieć dodatkowe pole `przedmiot`, a klasa `GraStrategiczna` pole `czasTrwania`. Dodaj konstruktory, metody gettery i settery, metodę `toString()` oraz `equals()` dla każdej z klas. Napisz program testujący zdefiniowane klasy i metody.
5. Utwórz klasę `Druzyna` z polami `nazwa`, `miasto` oraz `punkty` (jako `ArrayList` typu `Integer`). Utwórz klasy `DruzynaPilkarska` i `DruzynaSiatkarska`, które dziedziczą po klasie `Druzyna`. Klasa `DruzynaPilkarska` powinna mieć dodatkowe pole `pozycjaWRankingu`, a klasa `DruzynaSiatkarska` pole `liczbaZwyciestw`. Dodaj konstruktory, metody gettery i settery, metodę `toString()` oraz `equals()` dla każdej z klas. Napisz program testujący zdefiniowane klasy i metody.

6. Utwórz klasę `Komputer` z polami `producent`, `model` oraz `cenyCzesci` (jako `ArrayList` typu `Double`). Utwórz klasy `Laptop` i `Stacjonarny`, które dziedziczą po klasie `Komputer`. Klasa `Laptop` powinna mieć dodatkowe pole `waga`, a klasa `Stacjonarny` pole `obudowa`. Dodaj konstruktory, metody `getter`y i `setter`y, metodę `toString()` oraz `equals()` dla każdej z klas. Napisz program testujący zdefiniowane klasy i metody.
7. Utwórz klasę `AlbumMuzyczny` z polami `tytul`, `artysta` oraz `oceny` (jako `ArrayList` typu `Integer`). Utwórz klasy `AlbumRockowy` i `AlbumJazzowy`, które dziedziczą po klasie `AlbumMuzyczny`. Klasa `AlbumRockowy` powinna mieć dodatkowe pole `gatunekRocka`, a klasa `AlbumJazzowy` pole `gatunekJazzu`. Dodaj konstruktory, metody `getter`y i `setter`y, metodę `toString()` oraz `equals()` dla każdej z klas. Napisz program testujący zdefiniowane klasy i metody.
8. Zdefiniuj abstrakcyjną klasę `NarzedziePracy` z polami `nazwa` typu `String` oraz `rokProdukcji` typu `java.time.LocalDate`. Dodaj metodę abstrakcyjną `uzyj()`, która będzie symulować użycie narzędzia. Następnie zdefiniuj klasy `Mlotek`, `Srubokret` i `Pila`, które dziedziczą po klasie `NarzedziePracy` i implementują metodę `uzyj()`. Stwórz listę tablicową odpowiednich 5 obiektów i wywołaj dla nich napisaną metodę.
9. Zdefiniuj abstrakcyjną klasę `GrafikaKomputerowa` z polami `szerokosc`, `wysokosc` typu `int` oraz `nazwaPliku` typu `String`. Dodaj abstrakcyjne metody `wczytajPlik()` i `zapiszPlik()`. Następnie zdefiniuj klasy `Bitmapa` i `Wektor`, które dziedziczą po klasie `GrafikaKomputerowa` i implementują metody `wczytajPlik()` oraz `zapiszPlik()`. Stwórz listę tablicową odpowiednich 5 obiektów i wywołaj dla nich napisaną metodę.
10. Zdefiniuj abstrakcyjną klasę `UrządzenieElektroniczne` z polami `producent` typu `String`, `model` typu `String` oraz `rokProdukcji` typu `java.time.LocalDate`. Dodaj abstrakcyjne metody `włącz()` i `wyłącz()`. Następnie zdefiniuj klasy `Smartfon`, `Telewizor` i `Laptop`, które dziedziczą po klasie `UrządzenieElektroniczne` i implementują metody `włącz()` oraz `wyłącz()`. Stwórz listę tablicową odpowiednich 5 obiektów i wywołaj dla nich napisaną metodę.