

1. Stwórz nowy projekt. Napisz w nim pustą klasę `Vehicle`, oraz dwie dziedziczące po niej klasy `Car` i `Bicycle`.
2. Teraz utwórz interfejs `IRideable`, zawierający deklarację metody `Ride()` (z okna Solution Explorer wybierz Add/New Item/Interface/ew.ctr+shit+a).
3. Podepnij interfejs do klas `Car` i `Bicycle` i zaimplementuj metodę `Ride()` wyświetlając w konsoli komunikat „Jadę samochodem” lub „Jadę rowerem”.
4. Stwórz obiekt `car1` typu `Car`, oraz `bicycle1` typu `Bicycle`. Dla nowo utworzonych obiektów wywołaj metodę `Ride()` i zobacz co wyświetli się na ekranie.
5. Stwórz klasę `Skateboard`, dodaj w niej metodę `Ride` i przetestuj jej działanie na stworzonym obiekcie.
6. Stwórz nowy interfejs `IMusic` z metodą `PlayMelody()` i podobnij go do klasy `Bicycle`. Następnie za pomocą `Console.Beep` stwórz dźwięk dzwonka roweru (jeśli nie masz opcji usłyszenia użyj wyświetl łańcuch „dryń”). Metodę umieść w klasie `Bicycle`.
7. W klasie `Car` dodaj pola `year` (rok produkcji) i `brand` (marka) oraz konstruktor z parametrami (`year`, `brand`).
8. W pliku `Program.cs` stwórz kolekcję `Cars` i dodaj na nią kilka samochodów za pomocą konstruktora np.:

```
List<Car> cars = new List<Car>();
cars.Add(new Car(2010, "Ford"));
cars.Add(new Car(2016, "Fiat"));
cars.Add(new Car(2012, "Skoda"));
cars.Add(new Car(2010, "BMW"));
```
9. Do klasy `Car` dodaj interfejs `IComparable` i metodę `CompareTo`. Następnie posortuj elementy na liście i wyświetl je kolejno na konsoli. Na koniec zmień kolejność sortowania.

#### 1) Zwierzaki:

Stwórz klasę `Zwierze`, po której dziedziczyć będą następujące klasy: `Pies`, `Wilk`, `Rekin`, `Orzeł` i `Krokodyl`.

- a. Stwórz interfejs zawierający dwie metody: `void JakSiePoruszam()` i `void CoJem()`.
- b. Obiekt każdej klasy powinien umieć wywołać powyższe metody i odpowiedzieć na nie w sensowny sposób (np. „poruszam się na czterech łapach”)
- c. W każdej klasie potomnej po `Zwierze` dodaj pola: imię, waga, ilość nóg. Następnie poćwicz używanie interfejsów `IComparable`, `IEnumerable`, `ICloneable`.

#### 2) Zbiór figur:

Zaimplementuj klasy, które pozwolą reprezentować obiekty w dwuwymiarowej przestrzeni.

Punkt – reprezentowany przez dwie zmienne typu rzeczywistego `X` i `Y`

Koło – reprezentowane przez środek koła (punkt), oraz promień.

Kwadrat – reprezentowany przez 4 punkty krawędziowe.

(Wszystkie powyższe klasy posiadają też pole `nazwa` typu `string`.)

Stwórz potrzebne konstruktory.

Dodaj interfejsy:

- metoda `LiczObwód` (obwód punktu = 0)
- metoda `PobierzDaneZKlawiatury`

- metoda Wyświetl

Po zakończeniu implementacji stwórz kilka obiektów, policz ich obwód, a następnie je wyświetl. Na koniec stwórz kolekcję, posortuj obiekty.