Krotki, kolekcje, interfejsy

Krotka zapewnia zwięzłą składnię do grupowania wielu elementów danych w lekkiej strukturze danych. Poniższy przykład pokazuje, jak zadeklarować krotkę, zainicjować ją i uzyskać dostęp do jej danych:

```
(double, int) krotka1 = (4.5, 3);
Console.WriteLine($"Krotka z elementami {krotka1.Item1} i {krotka1.Item2}.");

// Wyjscie
// Krotka z elementami 4.5 i 3.

(double suma, int liczba) krotka2 = (4.5, 3)
Console.WriteLine($"Suma {krotka2.liczba} elementów wynosi {krotka2.suma}.");

// Wyjscie
// Suma 3 elementów wynosi 4.5.
```

Dictionary<TKey, TValue> (słownik, mapa) jest generyczną kolekcją, który przechowuje pary klucz-wartość, kolejność nie ma znaczenia.
Cechy słownika:
Znajduje się w przestrzeni nazw System.Collections.Generic.
Implementuje interfejs **IDictionary<TKey, TValue>**.
Klucze muszą być unikatowe i **nie mogą mieć wartości null**.
Wartości mogą być puste lub zduplikowane.
Dostęp do wartości można uzyskać, przekazując powiązany klucz w indeksatorze, np. mójSłownik[klucz]
Elementy są przechowywane jako obiekty **KeyValuePair<TKey, TValue>**.

Obiekt **Dictionary**<TKey, TValue> można utworzyć, przekazując typ kluczy i wartości, które może przechowywać. Poniższy przykład pokazuje, jak utworzyć słownik i dodać pary klucz-wartość.

```
Dictionary<int, string> nazwyLiczb = new();
nazwyLiczb.Add(1,"One"); /*dodanie a klucza/wartości za pomocą metody Add() */
nazwyLiczb.Add(2,"Two");
nazwyLiczb.Add(3,"Three");

// Próba dodania poniższego elementu zakończy się błędem, ponieważ w
słowniku istnienie element o kluczu 3.
//nazwyLiczb.Add(3, "Three");

foreach(KeyValuePair<int, string> kvp in nazwyLiczb)
{   Console.WriteLine("Key: {0}, Value: {1}", kvp.Key, kvp.Value); }

//Aktualizacja słownika:
nazwyLiczb[1] = "Jedynka";
nazwyLiczb[2] = "Dwójka";
nazwyLiczb[3] = "Trójka";
```

Zadania 9

Zadanie 9.1 Utwórz typ wyliczeniowych EnumTaryfa o wartościach taryfa1 (=1), taryfa2 (=2), taryfa3 (=4).

Zadanie 9.2 Utwórz interfejs **IAbonent** zawierający metody:

public string **PodajDane**();

public void **Zadzwon**(**double** czas, **EnumTaryfa** taryfa);

public (int, decimal) **PodsumowanieRozmow**();

Zadanie 9.3 Utwórz klasę **Polaczenie**, która zawiera pola **czasTrwania** (double), **oplata** (decimal), **wykonane** (bool). Dodaj konstruktor, który ustawia wszystkie pola.

Zadanie 9.4 Utwórz klasę **Abonent**, która zawiera pola **imie** (string), **nazwisko** (string), **numerTelefonu** (string), **polaczenia** (List<Polaczenie>). Dopisz konstruktor, który ustawia wszystkie pola. Dopisz metodę **ToString**(), która będzie wypisywała informacje o abonencie:

Jan Kowalski {777-034-232}, [liczba rozmów: 1, opłata: 20,00 zł]

- Klasa **Abonent** ma dodatkowo implementować interfejs IAbonent:
- Metoda string **PodajDane**() ma wypisywać imię i nazwisko abonenta.
- Metoda void **Zadzwon**(double czas, EnumTaryfa taryfa) ma wykonać próbę połączenia w następujący sposób: wylosuj liczbę z przedziału [0, 1). Jeżeli wylosowana liczba jest mniejsza od 0,3 (połączenie się nie powiodło), należy dodać do połączeń abonenta połączenie o czasie równym 0 oraz należy ustawić status wykonane na false; jeżeli wylosowana została liczba większa lub równa 0,3 należy dodać połączenie o podanym czasie, opłatę za połączenie należy obliczyć według podanej taryfy (jako iloczyn wartości taryfy oraz czasu).
- Metoda (int, decimal) **PodsumowanieRozmow**() ma zwracać krotkę: liczbę udanych połączeń oraz sumę opłat za te połączenia.

Zadanie 9.5 Utwórz klasę OperatorSieci, która zawiera pola nazwa (string), abonenci (**Dictionary**<string, Abonent>). Dopisz konstruktor, który ustawia wszystkie pola. Dopisz metodę ToString(), która będzie wypisywała informacje o operatorze:

Operator: IDEA [sumaryczny zysk: 316,00 zł]
Jan Kowalski {777-034-232}, [liczba rozmów: 3, opłata: 76,00 zł]
Edyta Nowak {666-634-009}, [liczba rozmów: 3, opłata: 240,00 zł]

Dopisz metody:
- void **DodajAbonenta**(Abonent a), która dodaje abonenta do abonentów.
- Abonent **WyszukajAbonenta**(string numerTelefonu), która wyszukuje abonenta o podanym numerze telefonu.

Zadanie 9.6 Utwórz co najmniej trzech abonentów, wykonaj dla każdego co najmniej trzy połączenia. Utwórz operatora sieci i dodaj abonentów. Wypisz informacje o operatorze:

Operator: IDEA [sumaryczny zysk: 216,00 zł]
Jan Kowalski {777-034-232}, [liczba rozmów: 2, opłata: 56,00 zł]
Edyta Nowak {666-634-009}, [liczba rozmów: 4, opłata: 160,00 zł]
Marian Waligóra {744-934-229}, [liczba rozmów: 0, opłata: 0,00 zł]