**Тип данных Dictionary**

Словарь хранит объекты, которые представляют пару ключ-значение. Класс словаря Dictionary<K, V> типизируется двумя типами: параметр K представляет тип ключей, а параметр V предоставляет тип значений.

Инициализация словаря:

Dictionary<int, string> people = new Dictionary<int, string>();

var people = new Dictionary<int, string>()

{

    { 5, "Tom"},

    { 3, "Sam"},

    { 11, "Bob"}

};

var people = new Dictionary<int, string>()

{

    [5] = "Tom",

    [6] = "Sam",

    [7] = "Bob"

};

Стоит отметить, что каждый элемент в словаре представляет структуру KeyValuePair<TKey, TValue>, где параметр TKey представляет тип ключа, а параметр TValue - тип значений элементов. Эта структура предоставляет свойства Key и Value, с помощью которых можно получить соответственно ключ и значение элемента в словаре. И одна из версий конструктора Dictionary позволяет инициализировать словарь коллекцией объектов KeyValuePair:

var mike = new KeyValuePair<int, string>(56, "Mike");

var employees = new List<KeyValuePair<int, string>>() { mike};

var people = new Dictionary<int, string>(employees);

Конструктор типа KeyValuePair принимает два параметра - ключ элемента и его значения. То есть в данном случае создается один такой элемент - mike с ключом 56 и значением "Mike". И этот элемент добавляется в список employees, которым затем инициализируется словарь.

Для перебора словаря можно применять цикл foreach:

var people = new Dictionary<int, string>()

{

    [5] = "Tom",

    [6] = "Sam",

    [7] = "Bob"

};

foreach(var person in people)

    Console.WriteLine($"key: {person.Key}  value: {person.Value}");

**Методы и свойства Dictionary**

Среди методов класса Dictionary можно выделить следующие:

* void Add(K key, V value): добавляет новый элемент в словарь
* void Clear(): очищает словарь
* bool ContainsKey(K key): проверяет наличие элемента с определенным ключом и возвращает true при его наличии в словаре
* bool ContainsValue(V value): проверяет наличие элемента с определенным значением и возвращает true при его наличии в словаре
* bool Remove(K key): удаляет по ключу элемент из словаря
* Другая версия этого метода позволяет получить удленный элемент в выходной параметр: bool Remove(K key, out V value)
* bool TryGetValue(K key, out V value): получает из словаря элемент по ключу key. При успешном получении передает значение элемента в выходной параметр value и возвращает true
* bool TryAdd(K key, V value): добавляет в словарь элемент с ключом key и значением value. При успешном добавлении возвращает true

Из свойств следует отметить свойство Count, которое возвращает количество элементов в словаре.