**Part 72 - What is dictionary in c#**

**In thie video, we will discuss dictionary object in c#.**  
**1.** A dictionary is a collection of (key, value) pairs.

**2.** Dictionary class is present in System.Collections.Generic namespace.

**3.** When creating a dictionary, we need to specify the type for key and value.  
**4.** Dictionary provides fast lookups for values using keys.  
**5.** Keys in the dictionary must be unique.

**Here is an example. The code is commented and is self-explanatory.**  
public class Program  
{  
    public static void Main()  
    {  
        // Create a Dictionary, CustomerID is the key. Type is int  
        // Customer object is the value. Type is Customer  
        Dictionary<int, Customer> dictionaryCustomers = new Dictionary<int, Customer>();  
  
        // Create Customer Objects  
        Customer customr1 = new Customer()  
        {  
            ID = 101,  
            Name = "Mark",  
            Salary = 5000  
        };  
  
        Customer customr2 = new Customer()  
        {  
            ID = 102,  
            Name = "Pam",  
            Salary = 7000  
        };  
  
        Customer customr3 = new Customer()  
        {  
            ID = 104,  
            Name = "Rob",  
            Salary = 5500  
        };  
  
        // Add customer objects to the dictionary  
        dictionaryCustomers.Add(customr1.ID, customr1);  
        dictionaryCustomers.Add(customr2.ID, customr2);  
        dictionaryCustomers.Add(customr3.ID, customr3);  
  
        // Retrieve the value (Customer object) from the dictionary,  
        // using key (customer ID). The fastest way to get a value  
        // from the dictionary is by using its key  
        Console.WriteLine("Customer 101 in customer dictionary");  
        Customer customer101 = dictionaryCustomers[101];  
        Console WriteLine("ID = {0}, Name = {1}, Salary = {2}",  
                            customer101.ID, customer101.Name, customer101.Salary);  
        Console.WriteLine("--------------------------------------------------");

   // It is also possible to loop thru each key/value pair in a dictionary  
        Console.WriteLine("All customer keys and values in customer dictionary");  
        foreach (KeyValuePair<int, Customer> customerKeyValuePair in dictionaryCustomers)  
        {  
            Console.WriteLine("Key = " + customerKeyValuePair.Key);  
            Customer cust = customerKeyValuePair.Value;  
            Console.WriteLine("ID = {0}, Name = {1}, Salary = {2}", cust.ID, cust.Name, cust.Salary);  
        }  
        Console.WriteLine("--------------------------------------------------");  
  
        // You can also use implicitly typed variable VAR to   
        // loop thru each key/value pair in a dictionary. But try  
        // to avoid using var, as this makes your code less readable  
        Console.WriteLine("All customer keys and values in customer dictionary");  
        foreach (var customerKeyValuePair in dictionaryCustomers)  
        {  
            Console.WriteLine("Key = " + customerKeyValuePair.Key);  
            Customer cust = customerKeyValuePair.Value;  
            Console.WriteLine("ID = {0}, Name = {1}, Salary = {2}", cust.ID, cust.Name, cust.Salary);  
        }  
        Console.WriteLine("--------------------------------------------------");  
  
        // To get all the keys in the dictionary  
        Console.WriteLine("All Keys in Customer Dictionary");  
        foreach (int key in dictionaryCustomers.Keys)  
        {  
            Console.WriteLine(key);  
        }  
        Console.WriteLine("--------------------------------------------------");  
  
        // To get all the values in the dictionary  
        Console.WriteLine("All Customer objects in Customer Dictionary");  
        foreach (Customer customer in dictionaryCustomers.Values)  
        {  
            Console.WriteLine("ID = {0}, Name = {1}, Salary = {2}", customer.ID, customer.Name, customer.Salary);  
        }  
  
        // If you try to add a key that already exists in the dictionary you   
        // will get an exception - An item with same key has already been   
        // added. So, check if the key already exists  
        if (!dictionaryCustomers.ContainsKey(101))  
        {  
            dictionaryCustomers.Add(101, customr1);  
        }  
  
        // When accessing a dictionary value by key, make sure the dictionary   
        // contains the key, otherwise you will get KeyNotFound exception.  
        if (dictionaryCustomers.ContainsKey(110))  
        {  
            Customer cus = dictionaryCustomers[110];  
        }  
        else  
        {  
            Console.WriteLine("Key does not exist in the dictionary");  
        }  
    }  
}  
  
public class Customer  
{  
    public int ID { get; set; }  
    public string Name { get; set; }  
    public int Salary { get; set; }  
}