

Function

Question 1 :

The main difference is passing by value is refer to passeing the value to the method, so any changes made inside the method do not affect the original variable. On the other hands , passing By Reference is taking a reference to the original variable is passed; changes made inside the method affect the original variable.

Example :

```
internal class Program
{
    1 reference
    static void PassingByValue(int x) { x += 10; }
    1 reference
    static void PassingByReference(ref int x) { x += 10; }

    0 references
    static void Main()
    {
        int value = 5;

        PassingByValue(value);
        Console.WriteLine(value); // Output: 5 (unchanged the original value)

        PassingByReference(ref value);
        Console.WriteLine(value); // Output: 15 (modified the original value)
    }
}
```

Question 2:

Passing by Value: A copy of the reference is passed; changes to the object's data persist, but reassigning the reference inside the method does not affect the original object.

Passing by Reference: The original reference is passed; changes to the object's data or reassignment affect the original object.

```
public class MyClass
{
    public int Data;
}
0 references
internal class Program
{
    1 reference
    static void PaddingByValue(MyClass obj) { obj = new MyClass { Data = 10 }; }
    1 reference
    static void PassingByReference(ref MyClass obj) { obj = new MyClass { Data = 10 }; }

    0 references
    static void Main()
    {
        MyClass obj = new MyClass();

        PaddingByValue(obj);
        Console.WriteLine(obj.Data);

        PassingByReference(ref obj);
        Console.WriteLine(obj.Data);
    }
}
```