PE5

Question 1. (5 points)

You are asked to write a console application that includes:

- A class `Employee` that has four public properties: `Id` (int), `Name` (string), `Dob` (DateOnly), `IsMale` (bool), along with necessary constructors and methods.
- A generated delegate: `void Presentation<T>(T item)`.
- A generic class `Department<T>` that has:
  - A public property `Title` (string).
  - A private member that can store a list of type `T`.
  - A necessary constructor.
  - Three public methods:
    - `void Add(T item)`: Adds a new item of type `T` into the private list.
    - `bool Contains(T item)`: Checks if the `item` is in the private list.
    - `void Display(Presentation<T> presentation)`: Displays the `Title` and items in the private list to the console. Each item in the list must be displayed according to the definition of the `presentation`.

Using everything you have defined, the `Program` class (shown in Figure 1) will produce the result as shown in Figure 2. Note that two employees are considered equal if all their information is the same.

Program.cs

```
namespace Question1
{
    internal class Program
    {
        private static void Main(string[] args)
        {
            Console.WriteLine("Requirement 1:");
            Employee s = new Employee(1, "Nguyen Van A", new DateOnly(1999, 10, 20), false);
            Console.WriteLine("You have entered:");
            Console.WriteLine(s);

            Console.WriteLine(Environment.NewLine + "----------");
            Console.WriteLine("Requirement 2:");
            Department<Employee> department = new Department<Employee>("Accounting Department");
            department.Add(new Employee(2, "Nguyen Van B", new DateOnly(1999, 10, 20), false));
            department.Add(new Employee(3, "Nguyen Van C", new DateOnly(1989, 11, 15), true));
            department.Add(new Employee(4, "Nguyen Van D", new DateOnly(2000, 4, 2), true));
            department.Display(DisplaysFullInfoOfEmployee);

            Console.WriteLine(Environment.NewLine + "----------");
            Console.WriteLine("Requirement 3:");
            Employee employee = new Employee(3, "Nguyen Van C", new DateOnly(1989, 11, 15), true);
            if (department.Contains(employee))
                Console.WriteLine("The employee you are looking for belongs to the department");
```

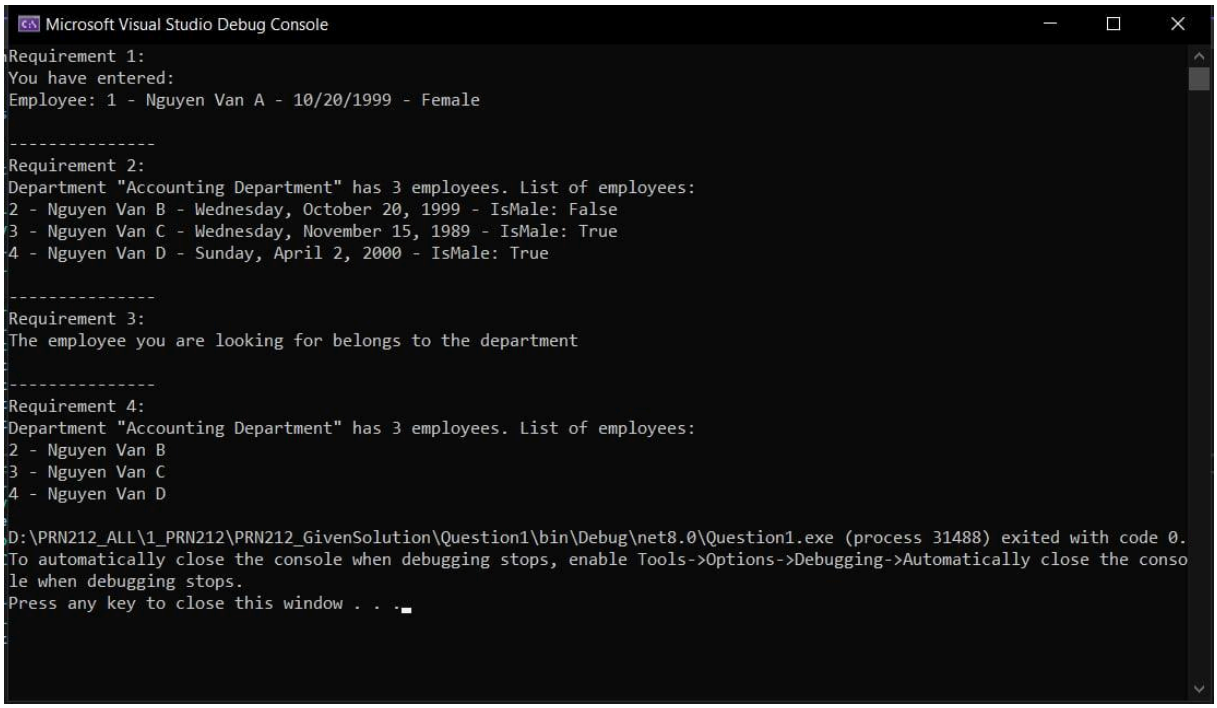
```
            else Console.WriteLine("The employee you are looking for does not belong to the department");

            Console.WriteLine(Environment.NewLine + "---------------");
            Console.WriteLine("Requirement 4:");
            department.Display(DisplaysBriefInfoOfEmployee);
        }

        private static void DisplaysFullInfoOfEmployee(Employee employee)
        {
            Console.WriteLine($"{employee.Id} - {employee.Name} - {employee.Dob.ToLongDateString()} - IsMale:
{employee.IsMale}");
        }

        private static void DisplaysBriefInfoOfEmployee(Employee employee)
        {
            Console.WriteLine($"{employee.Id} - {employee.Name}");
        }
    }
}
```

Results



Question 2

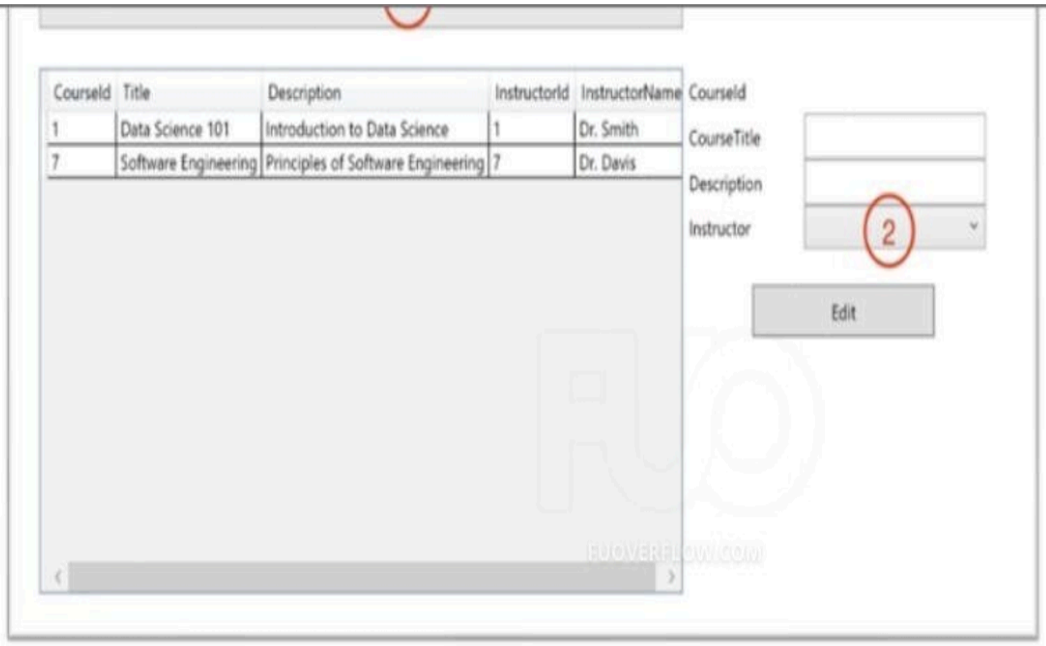


Figure 2. The required window.

When the window is loaded, display a list of all categories and instructors in the two ComboBoxes labeled 1 and 2 (see Figure 2), respectively. By default, the first category is selected (see Figure 3).

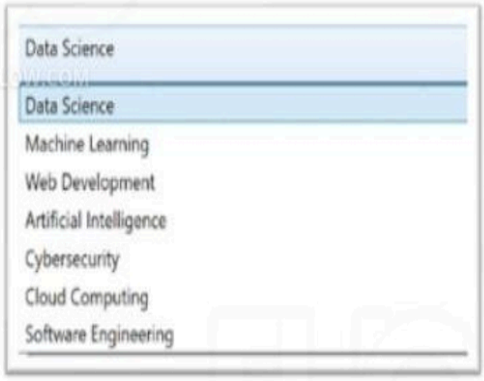
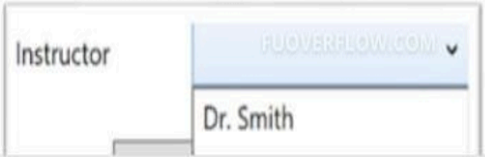


Figure 3. ComboBox labeled as 1 displays a list of all categories.



- When the window is loaded, display a list of all categories and instructors in the two ComboBoxes labeled 1 and 2 (see Figure 2), respectively. By default, the first category is selected (see Figure 3).

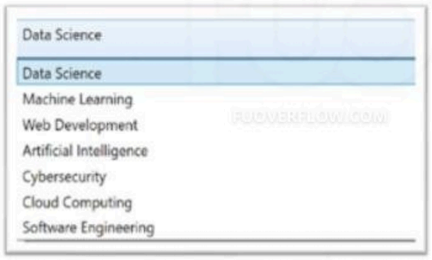


Figure 3. ComboBox labeled as 1 displays a list of all categories.

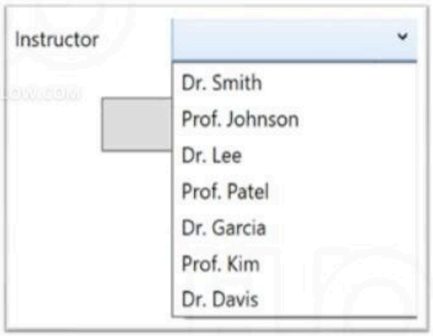


Figure 4. ComboBox labeled as 2 displays a list of all instructors.

- Display a list of all courses belonging to the first category.