Model View Presenter - An Example Implementation With WinForms
Creating Services (Notes For Video #4)

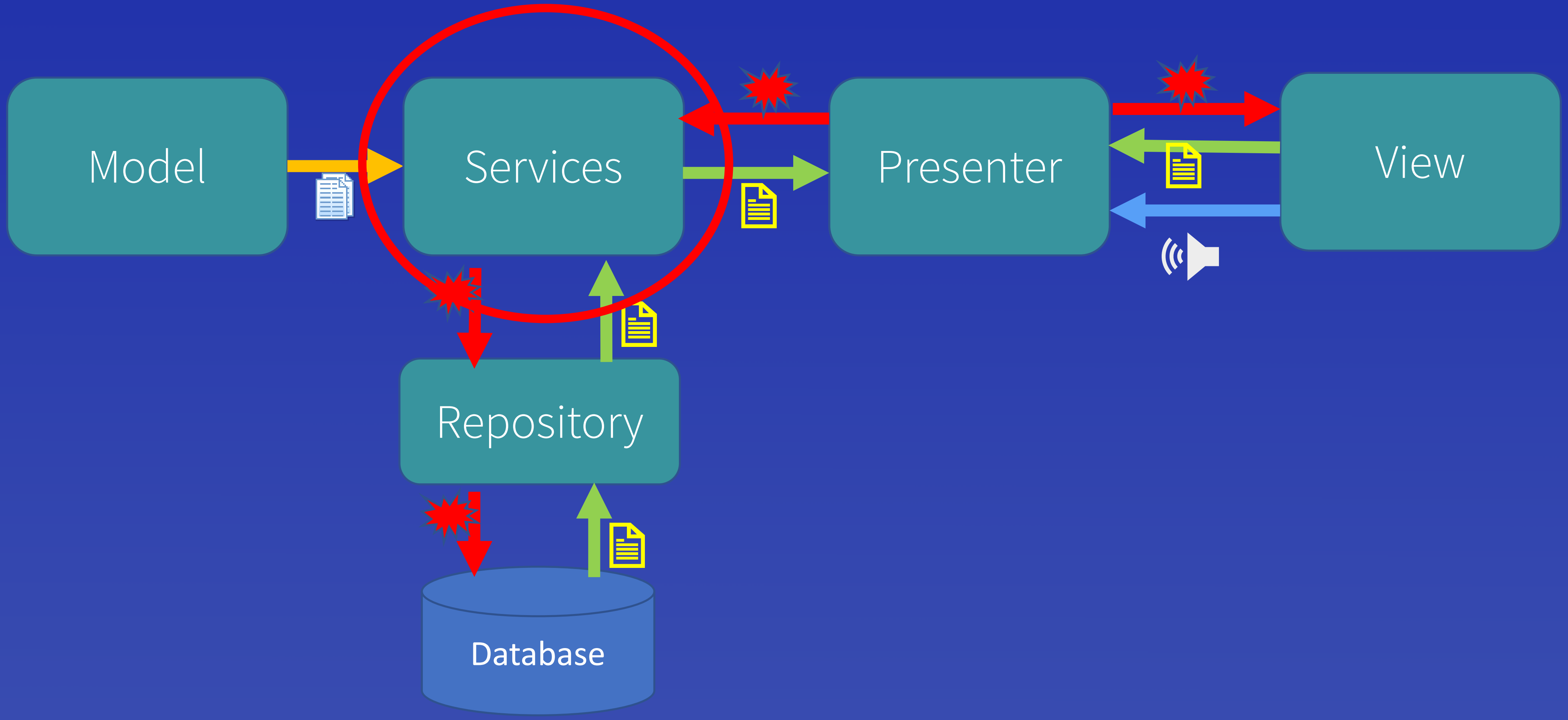
Presented by Robert G. Marquez

Release #2

Updated December 16, 2019

The Services Layer In Model View Presenter Design

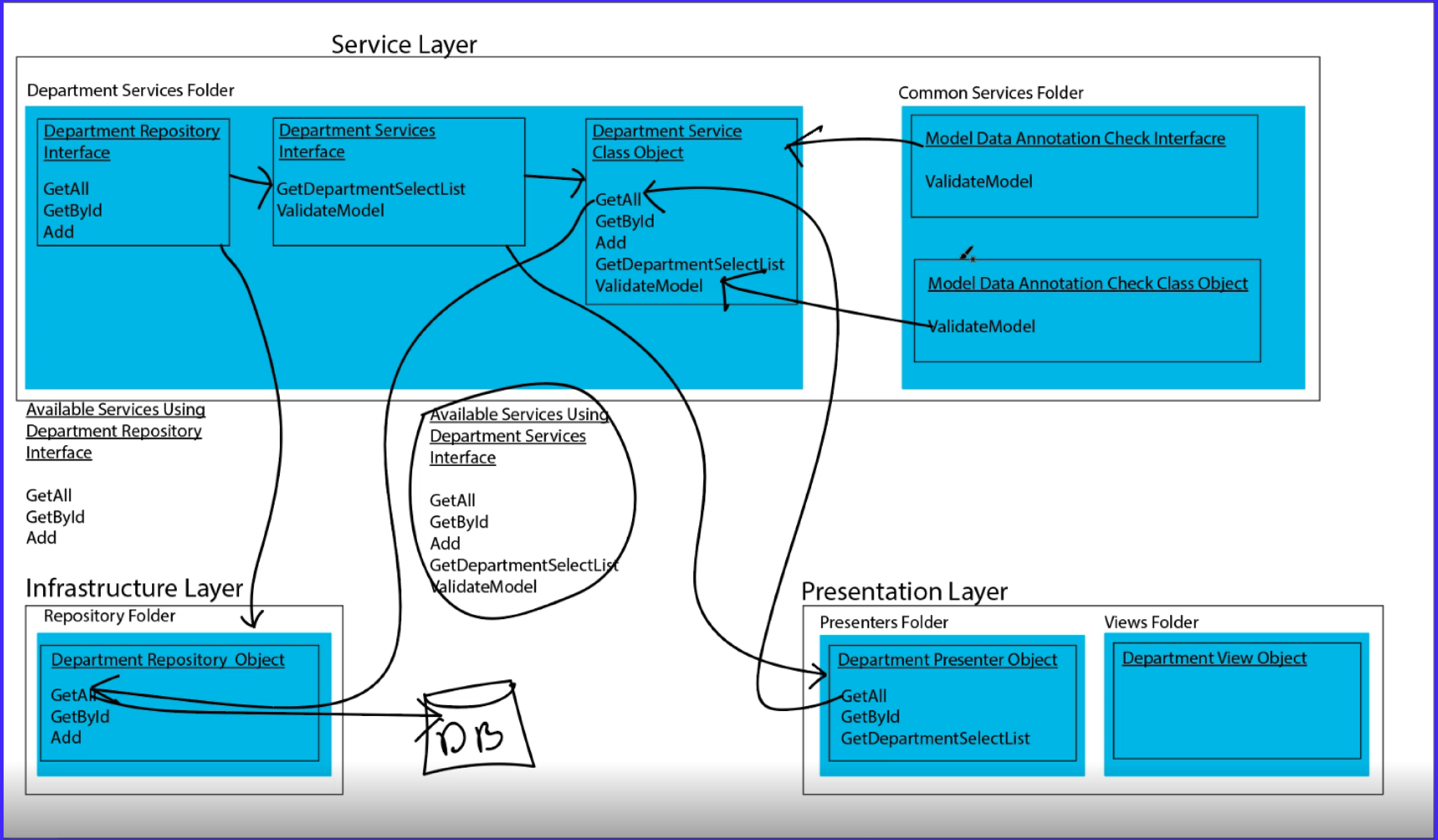
Areas Covered In This Video
- Create Service Layer class library in Visual Studio 2017.
- Create Service Layer folder structure.
- Create Service Layer interface files.



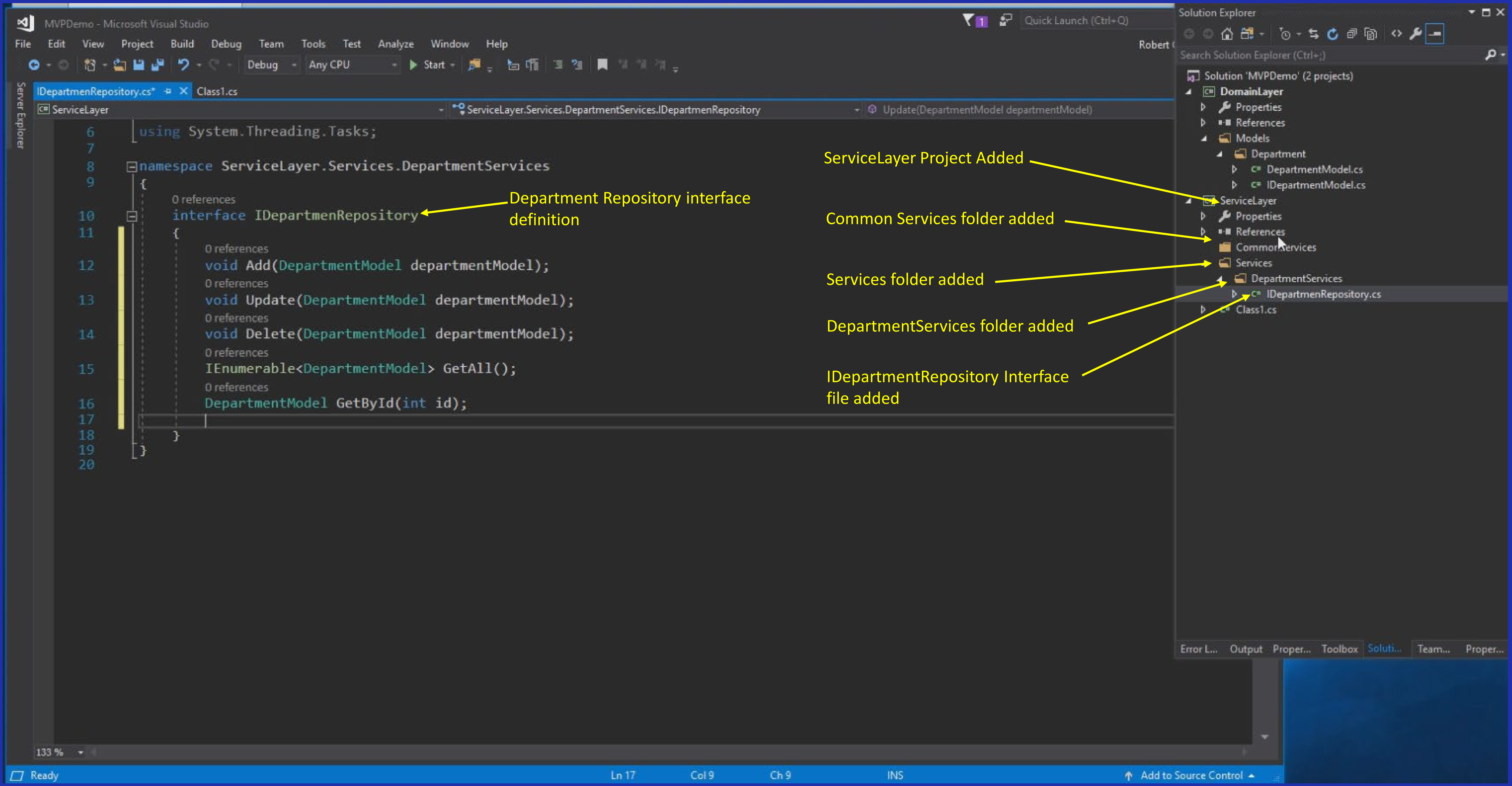
Legend:

Indicates the source is executing methods defined in the target.	🟥
Data from the sources is being sent to the target due to request from target.	🟩
The target is creating instances of the source but is not executing anything in the source	🟨
The target is listening to events from the source. The target will typically execute code in response to an event it receives	🟦

Service Layer Interactions With Other Layers and Objects



Service Layer Creation – Folders and Department Repository Interface File



```
using System.Threading.Tasks;

namespace ServiceLayer.Services.DepartmentServices
{
    0 references
    interface IDepartmenRepository
    {
        0 references
        void Add(DepartmentModel departmentModel);
        0 references
        void Update(DepartmentModel departmentModel);
        0 references
        void Delete(DepartmentModel departmentModel);
        0 references
        IEnumerable<DepartmentModel> GetAll();
        0 references
        DepartmentModel GetById(int id);

    }
}
```

Department Repository interface definition

ServiceLayer Project Added

Common Services folder added

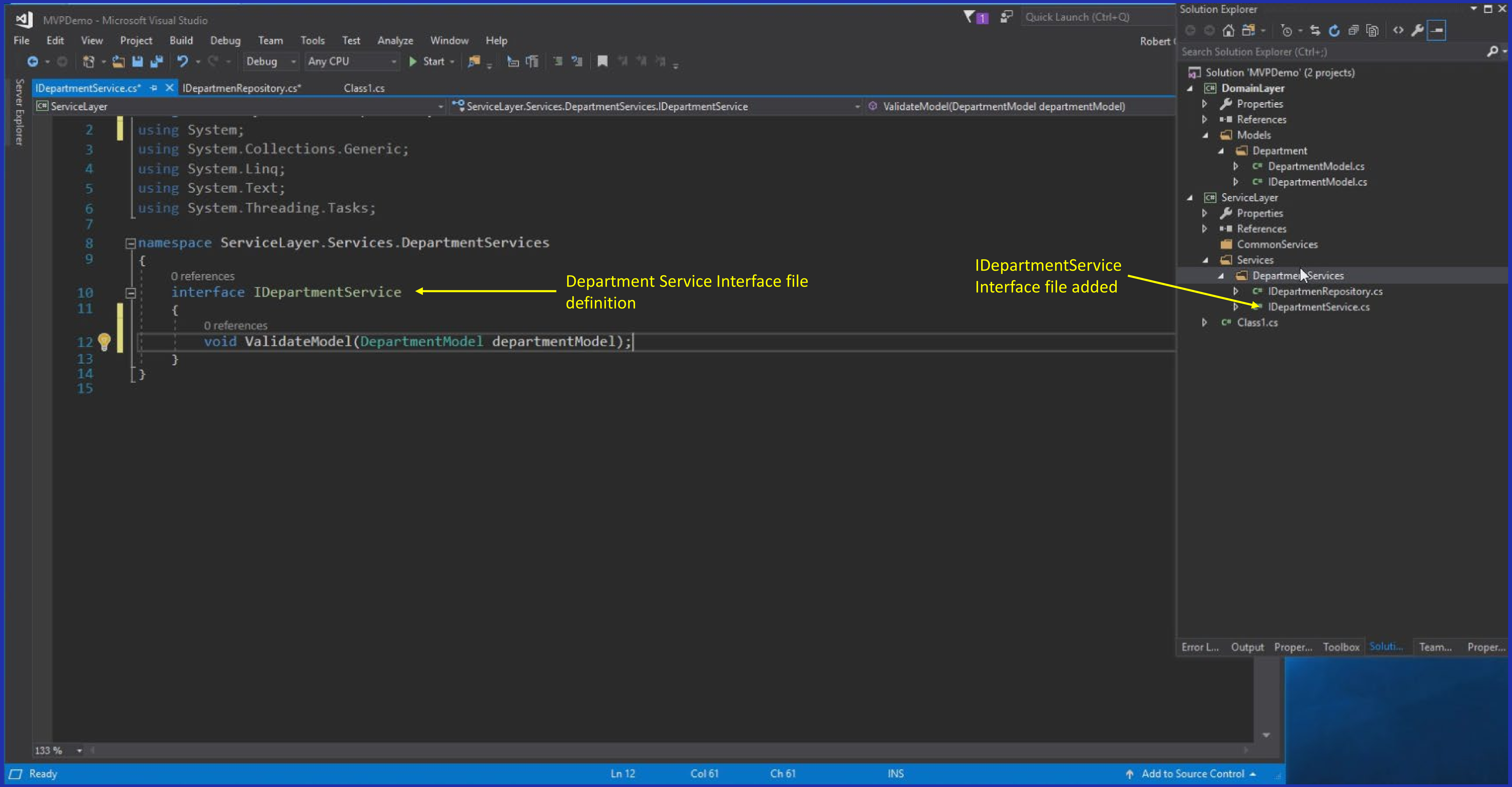
Services folder added

DepartmentServices folder added

IDepartmentRepository Interface file added

MVPDemo - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug Any CPU Start
IDepartmenRepository.cs* Class1.cs
ServiceLayer
ServiceLayer.Services.DepartmentServices.IDepartmenRepository
Update(DepartmentModel departmentModel)
Solution Explorer
Search Solution Explorer (Ctrl+;)
Solution 'MVPDemo' (2 projects)
DomainLayer
Properties
References
Models
Department
DepartmentModel.cs
IDepartmentModel.cs
ServiceLayer
Properties
References
CommonServices
Services
DepartmentServices
IDepartmenRepository.cs
Class1.cs
Error L... Output Proper... Toolbox Soluti... Team... Proper...
Ready Ln 17 Col 9 Ch 9 INS Add to Source Control

Service Layer Creation – Department Service Interface File

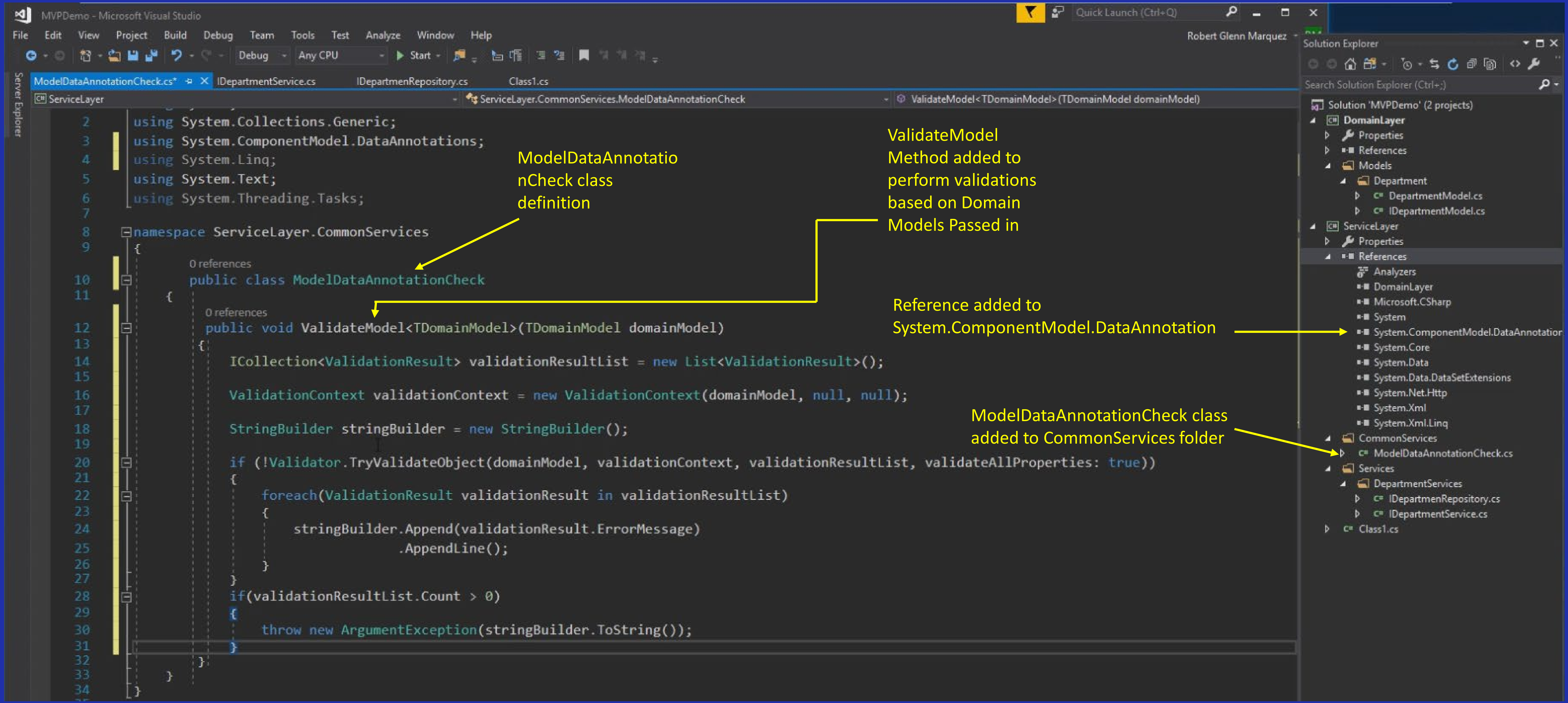

MVPDemo - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug Any CPU Start
IDepartmentService.cs* IDepartmenRepository.cs* Class1.cs
ServiceLayer
ServiceLayer.Services.DepartmentServices.IDepartmentService
ValidateModel(DepartmentModel departmentModel)
Solution Explorer
Solution 'MVPDemo' (2 projects)
DomainLayer
Properties
References
Models
Department
DepartmentModel.cs
IDepartmentModel.cs
ServiceLayer
Properties
References
CommonServices
Services
DepartmentServices
IDepartmenRepository.cs
IDepartmentService.cs
Class1.cs
Ready Ln 12 Col 61 Ch 61 INS Add to Source Control

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace ServiceLayer.Services.DepartmentServices
{
    0 references
    interface IDepartmentService
    {
        0 references
        void ValidateModel(DepartmentModel departmentModel);
    }
}
```

Department Service Interface file definition

IDepartmentService Interface file added

Service Layer Creation – ModelDataAnnotationCheck Class added to CommonServices Folder


MVPDemo - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug Any CPU Start
Quick Launch (Ctrl+Q)
Robert Glenn Marquez
ModelDataAnnotationCheck.cs* IDepartmentService.cs IDepartmenRepository.cs Class1.cs
ServiceLayer
ServiceLayer.CommonServices.ModelDataAnnotationCheck
ValidateModel<TDomainModel>(TDomainModel domainModel)
Server Explorer
Solution Explorer
Search Solution Explorer (Ctrl+;)
Solution 'MVPDemo' (2 projects)
DomainLayer
Properties
References
Models
Department
DepartmentModel.cs
IDepartmentModel.cs
ServiceLayer
Properties
References
Analyzers
DomainLayer
Microsoft.CSharp
System
System.ComponentModel.DataAnnotatior
System.Core
System.Data
System.Data.DataSetExtensions
System.Net.Http
System.Xml
System.Xml.Linq
CommonServices
ModelDataAnnotationCheck.cs
Services
DepartmentServices
IDepartmenRepository.cs
IDepartmentService.cs
Class1.cs

```
using System.Collections.Generic;
using System.ComponentModel.DataAnnotations;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace ServiceLayer.CommonServices
{
    0 references
    public class ModelDataAnnotationCheck
    {
        0 references
        public void ValidateModel<TDomainModel>(TDomainModel domainModel)
        {
            ICollection<ValidationResult> validationResultList = new List<ValidationResult>();

            ValidationContext validationContext = new ValidationContext(domainModel, null, null);

            StringBuilder stringBuilder = new StringBuilder();

            if (!Validator.TryValidateObject(domainModel, validationContext, validationResultList, validateAllProperties: true))
            {
                foreach(ValidationResult validationResult in validationResultList)
                {
                    stringBuilder.Append(validationResult.ErrorMessage)
                                 .AppendLine();
                }
            }
            if(validationResultList.Count > 0)
            {
                throw new ArgumentException(stringBuilder.ToString());
            }
        }
    }
}
```

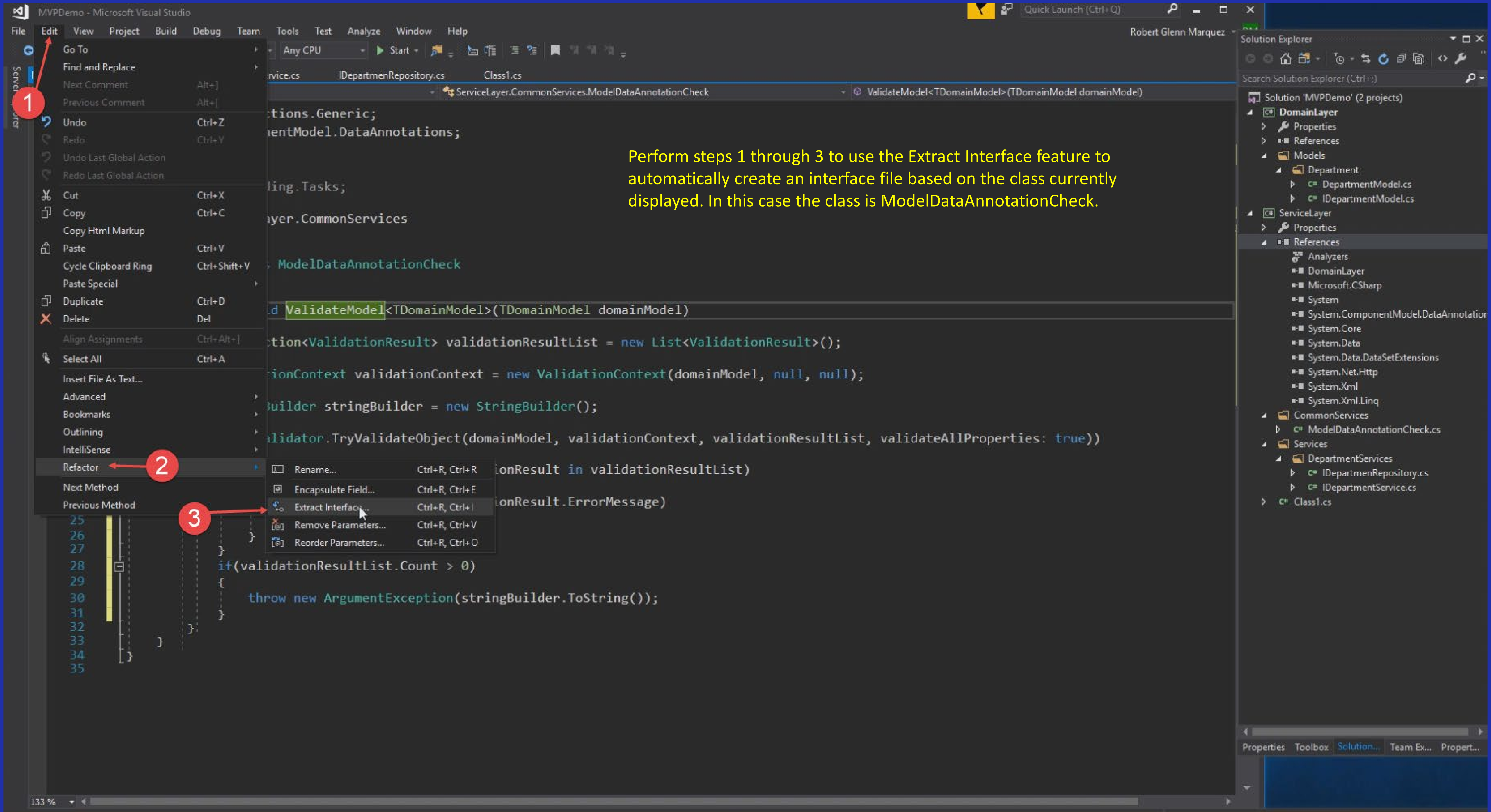
ModelDataAnnotationCheck class definition

ValidateModel Method added to perform validations based on Domain Models Passed in

Reference added to System.ComponentModel.DataAnnotation

ModelDataAnnotationCheck class added to CommonServices folder

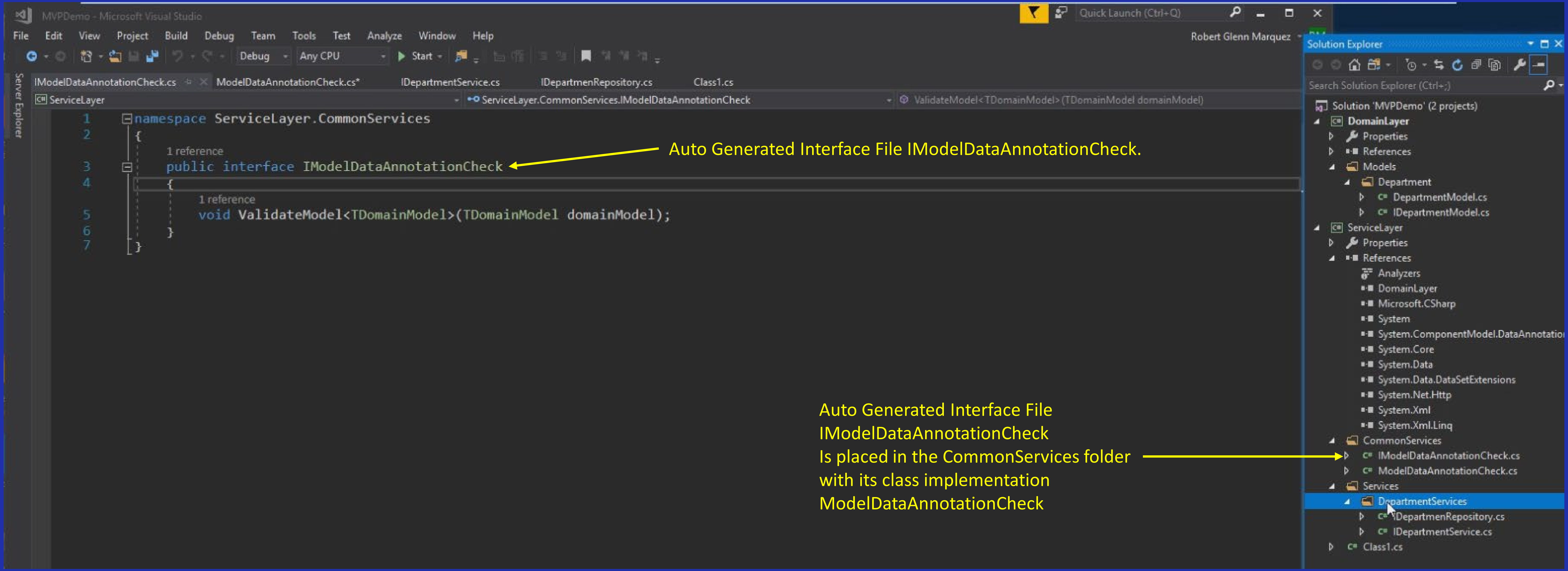
Service Layer Creation – Automatically Generate Interface File For ModelDataAnnotationCheck Class



```
tions.Generic;
nentModel.DataAnnotations;

ling.Tasks;

ayer.CommonServices

 ModelDataAnnotationCheck

d ValidateModel<TDomainModel>(TDomainModel domainModel)

tion<ValidationResult> validationResultList = new List<ValidationResult>();

ionContext validationContext = new ValidationContext(domainModel, null, null);

Builder stringBuilder = new StringBuilder();

alidator.TryValidateObject(domainModel, validationContext, validationResultList, validateAllProperties: true))

onResult in validationResultList)

onResult.ErrorMessage)

            }
        }
        if(validationResultList.Count > 0)
        {
            throw new ArgumentException(stringBuilder.ToString());
        }
    }
}
```

Perform steps 1 through 3 to use the Extract Interface feature to automatically create an interface file based on the class currently displayed. In this case the class is ModelDataAnnotationCheck.

Service Layer Creation –Generated Interface File For ModelDataAnnotationCheck Class

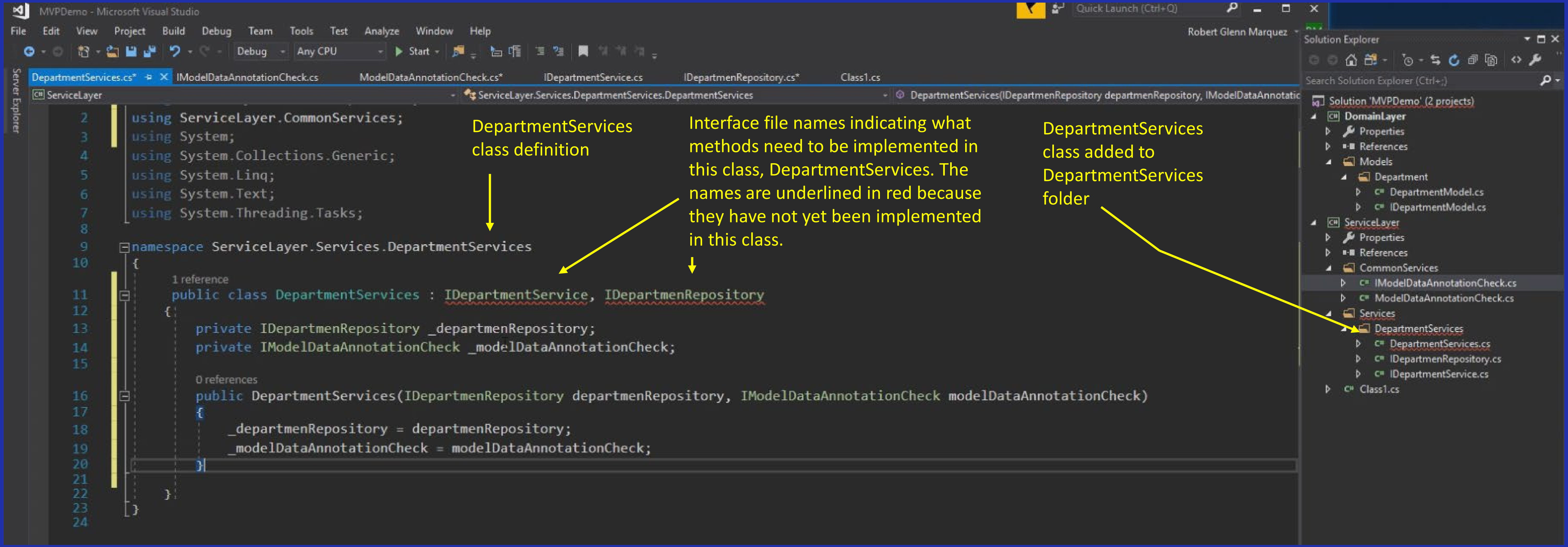


```
namespace ServiceLayer.CommonServices
{
    1 reference
    public interface IModelDataAnnotationCheck
    {
        1 reference
        void ValidateModel<TDomainModel>(TDomainModel domainModel);
    }
}
```

Auto Generated Interface File IModelDataAnnotationCheck.

Auto Generated Interface File IModelDataAnnotationCheck Is placed in the CommonServices folder with its class implementation ModelDataAnnotationCheck

Service Layer Creation – Creation of DepartmentServices Class


MVPDemo - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug Any CPU Start
Quick Launch (Ctrl+Q)
Robert Glenn Marquez
DepartmentServices.cs* IModelDataAnnotationCheck.cs ModelDataAnnotationCheck.cs* IDepartmentService.cs IDepartmenRepository.cs* Class1.cs
ServiceLayer
ServiceLayer.Services.DepartmentServices.DepartmentServices
DepartmentServices(IDepartmenRepository departmenRepository, IModelDataAnnotatic
Server Explorer
Solution Explorer
Search Solution Explorer (Ctrl+;)
Solution 'MVPDemo' (2 projects)
DomainLayer
Properties
References
Models
Department
DepartmentModel.cs
IDepartmentModel.cs
ServiceLayer
Properties
References
CommonServices
IModelDataAnnotationCheck.cs
ModelDataAnnotationCheck.cs
Services
DepartmentServices
DepartmentServices.cs
IDepartmenRepository.cs
IDepartmentService.cs
Class1.cs

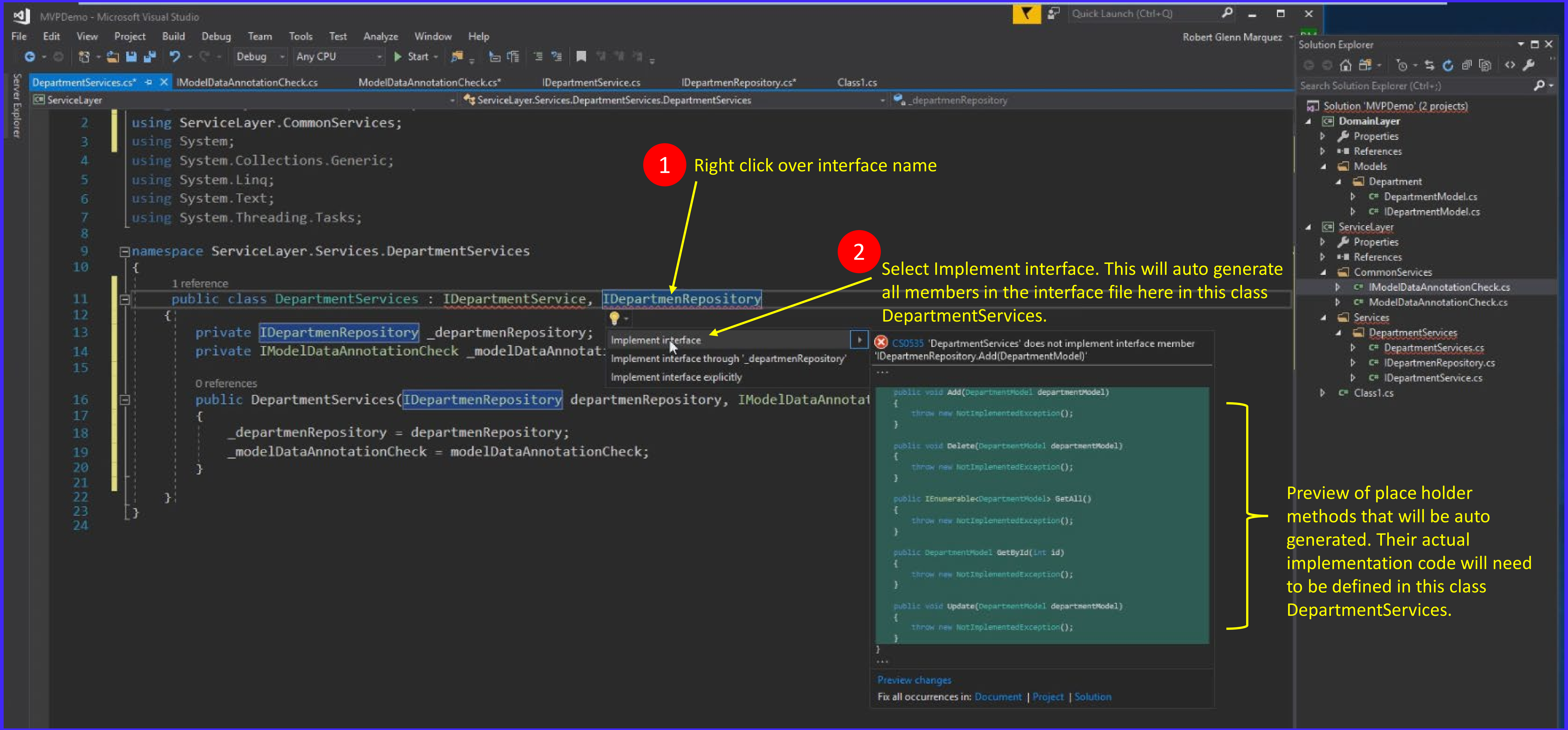
```
using ServiceLayer.CommonServices;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace ServiceLayer.Services.DepartmentServices
{
    1 reference
    public class DepartmentServices : IDepartmentService, IDepartmenRepository
    {
        private IDepartmenRepository _departmenRepository;
        private IModelDataAnnotationCheck _modelDataAnnotationCheck;

        0 references
        public DepartmentServices(IDepartmenRepository departmenRepository, IModelDataAnnotationCheck modelDataAnnotationCheck)
        {
            _departmenRepository = departmenRepository;
            _modelDataAnnotationCheck = modelDataAnnotationCheck;
        }

    }
}
```

DepartmentServices class definition

Interface file names indicating what methods need to be implemented in this class, DepartmentServices. The names are underlined in red because they have not yet been implemented in this class.

DepartmentServices class added to DepartmentServices folder

Service Layer Creation – Auto Generate Interface File Using Right Click Over Interface File Name


MVPDemo - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug Any CPU Start
Quick Launch (Ctrl+Q)
Robert Glenn Marquez
DepartmentServices.cs* IModelDataAnnotationCheck.cs ModelDataAnnotationCheck.cs* IDepartmentService.cs IDepartmenRepository.cs* Class1.cs
ServiceLayer
ServiceLayer.Services.DepartmentServices.DepartmentServices
_departmenRepository
Server Explorer
Solution Explorer
Search Solution Explorer (Ctrl+;)
Solution 'MVPDemo' (2 projects)
DomainLayer
Properties
References
Models
Department
DepartmentModel.cs
IDepartmentModel.cs
ServiceLayer
Properties
References
CommonServices
IModelDataAnnotationCheck.cs
ModelDataAnnotationCheck.cs
Services
DepartmentServices
DepartmentServices.cs
IDepartmenRepository.cs
IDepartmentService.cs
Class1.cs

```
using ServiceLayer.CommonServices;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace ServiceLayer.Services.DepartmentServices
{
    1 reference
    public class DepartmentServices : IDepartmentService, IDepartmenRepository
    {
        private IDepartmenRepository _departmenRepository;
        private IModelDataAnnotationCheck _modelDataAnnotat

        0 references
        public DepartmentServices(IDepartmenRepository departmenRepository, IModelDataAnnotat
        {
            _departmenRepository = departmenRepository;
            _modelDataAnnotationCheck = modelDataAnnotationCheck;
        }

    }
}
```

1 Right click over interface name

2 Select Implement interface. This will auto generate all members in the interface file here in this class DepartmentServices.

Implement interface
Implement interface through '_departmenRepository'
Implement interface explicitly

CS0535 'DepartmentServices' does not implement interface member 'IDepartmenRepository.Add(DepartmentModel)'

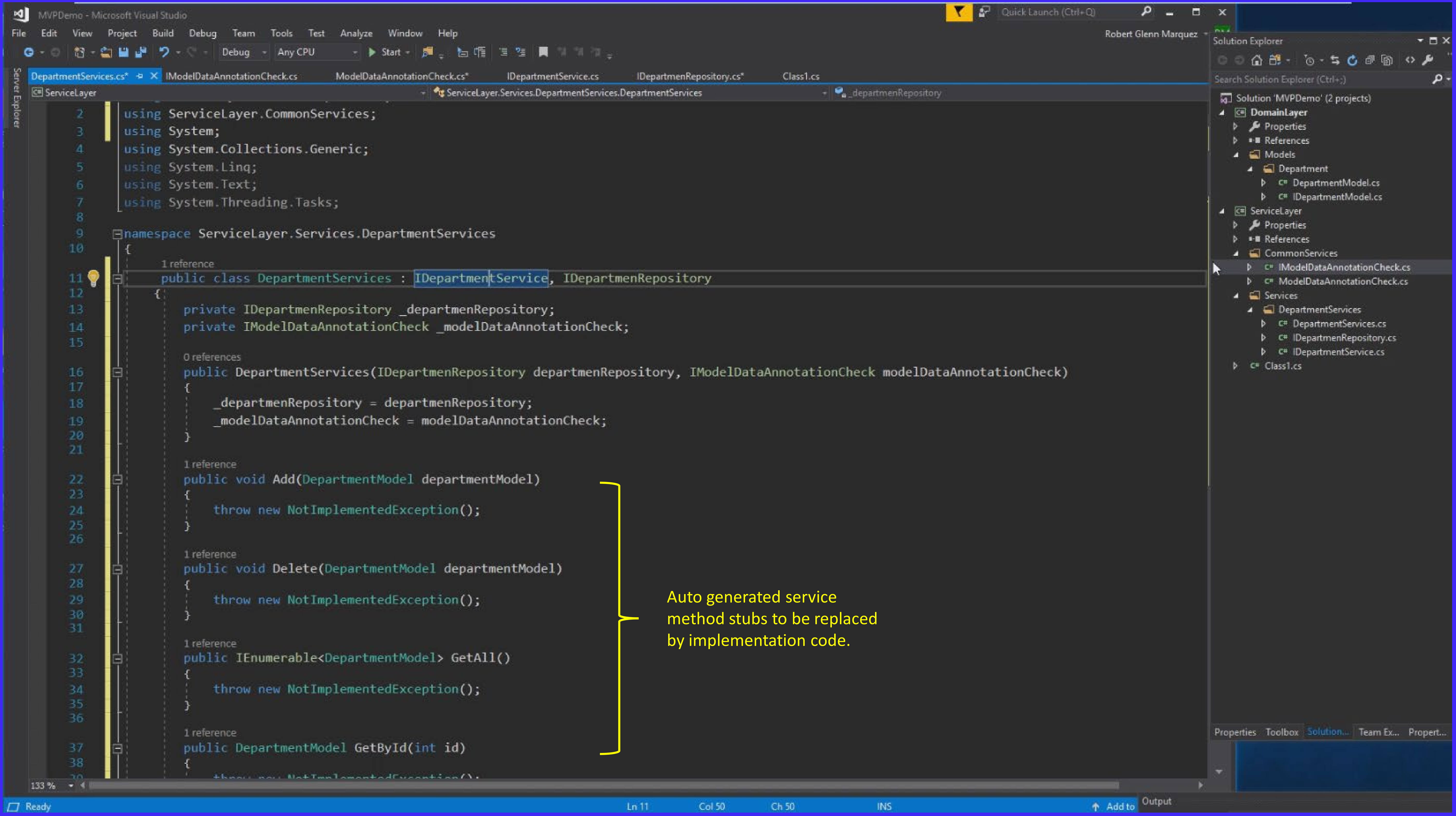
```
...
    public void Add(DepartmentModel departmentModel)
    {
        throw new NotImplementedException();
    }

    public void Delete(DepartmentModel departmentModel)
    {
        throw new NotImplementedException();
    }

    public IEnumerable<DepartmentModel> GetAll()
    {
        throw new NotImplementedException();
    }

    public DepartmentModel GetById(int id)
    {
        throw new NotImplementedException();
    }

    public void Update(DepartmentModel departmentModel)
    {
        throw new NotImplementedException();
    }
}
...
```

Preview changes

Fix all occurrences in: Document | Project | Solution

Preview of place holder methods that will be auto generated. Their actual implementation code will need to be defined in this class DepartmentServices.

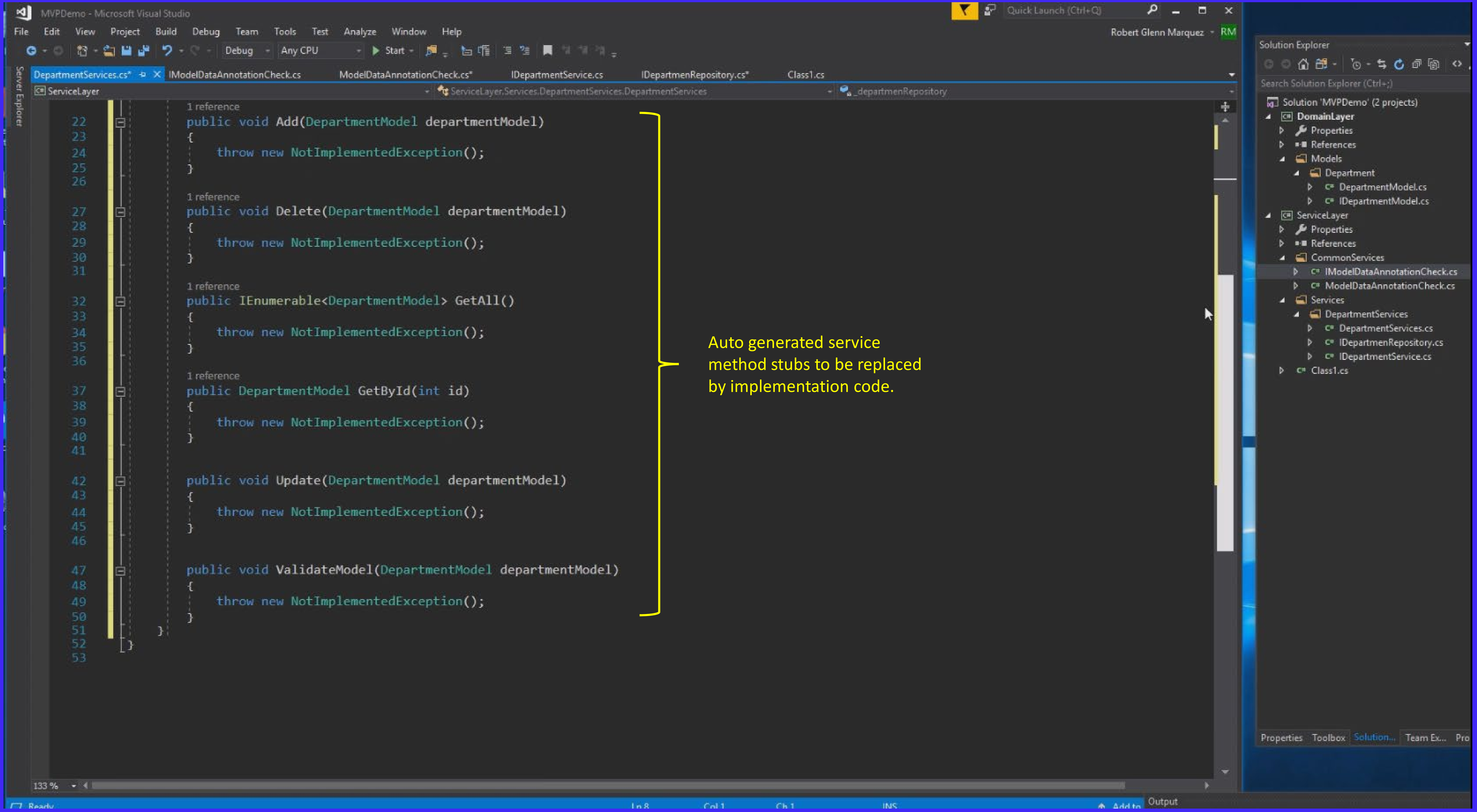
Service Layer Creation – Auto Generated Service Method Stubs



```
using ServiceLayer.CommonServices;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace ServiceLayer.Services.DepartmentServices
{
    1 reference
    public class DepartmentServices : IDepartmentService, IDepartmenRepository
    {
        private IDepartmenRepository _departmenRepository;
        private IModelDataAnnotationCheck _modelDataAnnotationCheck;

        0 references
        public DepartmentServices(IDepartmenRepository departmenRepository, IModelDataAnnotationCheck modelDataAnnotationCheck)
        {
            _departmenRepository = departmenRepository;
            _modelDataAnnotationCheck = modelDataAnnotationCheck;
        }

        1 reference
        public void Add(DepartmentModel departmentModel)
        {
            throw new NotImplementedException();
        }

        1 reference
        public void Delete(DepartmentModel departmentModel)
        {
            throw new NotImplementedException();
        }

        1 reference
        public IEnumerable<DepartmentModel> GetAll()
        {
            throw new NotImplementedException();
        }

        1 reference
        public DepartmentModel GetById(int id)
        {
```

Auto generated service method stubs to be replaced by implementation code.

Service Layer Creation – Auto Generated Service Method Stubs



```
        1 reference
        public void Add(DepartmentModel departmentModel)
        {
            throw new NotImplementedException();
        }

        1 reference
        public void Delete(DepartmentModel departmentModel)
        {
            throw new NotImplementedException();
        }

        1 reference
        public IEnumerable<DepartmentModel> GetAll()
        {
            throw new NotImplementedException();
        }

        1 reference
        public DepartmentModel GetById(int id)
        {
            throw new NotImplementedException();
        }

        public void Update(DepartmentModel departmentModel)
        {
            throw new NotImplementedException();
        }

        public void ValidateModel(DepartmentModel departmentModel)
        {
            throw new NotImplementedException();
        }
    }
}
```

Auto generated service method stubs to be replaced by implementation code.

MVPDemo - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
Debug Any CPU Start
Robert Glenn Marquez RM
DepartmentServices.cs* IModelDataAnnotationCheck.cs ModelDataAnnotationCheck.cs* IDepartmentService.cs IDepartmenRepository.cs* Class1.cs
ServiceLayer
ServiceLayer.Services.DepartmentServices.DepartmentServices
_departmenRepository
Solution Explorer
Solution 'MVPDemo' (2 projects)
DomainLayer
Properties
References
Models
Department
DepartmentModel.cs
IDepartmentModel.cs
ServiceLayer
Properties
References
CommonServices
IModelDataAnnotationCheck.cs
ModelDataAnnotationCheck.cs
Services
DepartmentServices
DepartmentServices.cs
IDepartmenRepository.cs
IDepartmentService.cs
Class1.cs
133 %
Ready

Service Layer Creation – End of Notes For Video #4