**Scaffold a Model**

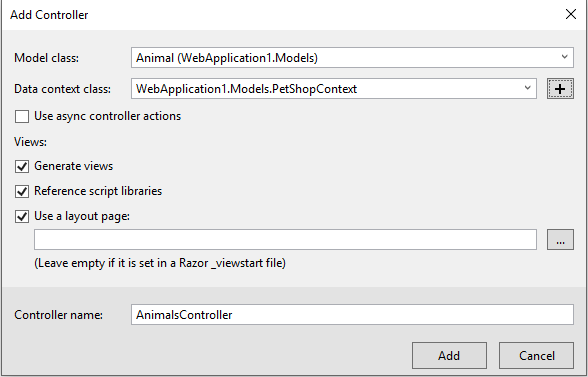
Create a new Class under Models (Right click Modes > Add > New Item, Class should be found under C# > Code) called Animal.

Create two public properties, a string **name** and an integer **ID**.

Scaffold the animal model by right clicking **Controller > Add > New Scaffolded Item**

Select **MVC 5 Controller with views, using Entity Framework.**

Create a new Data context class named **PetShopContext**



This will generate a newly created EF Context. You may receive an error prompting you to rebuild the project. You can do this by right clicking the project or solution and clicking build (or Build > Build Solution, at the top menu bar)

This will also create a controller with all the basic CRUD (Create, Replace, Update, Delete) actions, and their respective views.

Run the application and navigate to /Animals. You can add a breakpoint to the index of the AnimalsController to debug and see the request in process.

Add a new link to your Views/Shared/\_Layout.cshtml to the above link using @Html.ActionLink.

**Seed Sample Data**

Create a sub folder under Models called **Initializers**.

Create a PetShopInitiailizer.cs class

Import the name space System.Data.Entity

Inherit the class **DropCreateDatabaseAlways** and pass **PetShopContext** as the generic (ie DropCreateDatabaseAlways<PetShopContext>

Override the prototype void seed(PetShopContext) (ie protected override void Seed(PetShopContext context))

Create multiple Animal objects, then add them to the DBSet Animals of the PetShop Context, and call base.Seed(context) to commit the changes.

In Global.asax.cs, inside of Application\_Start(), we want to invoke our initializer.

Import the name spaces System.Data.Entity and [Project Name].Models.Initializers (or the name space of PetShopInitiailizer if you changed it) and tell the application to run the initializer on start up by adding the line;

Database.SetInitializer(new PetShopInitializer());

Your local database will now come with the sample seed data defined inside your initializer, you can verify this by running the application, clinking the link (or navigating to) /Animals