**IncidentDAL Class**

1. Create a method that returns a DataTable variable.
2. Create a new SqlCommand object.
3. Create a new SqlConnection object.
4. Get & Open the Connection
5. Create a new DataTable object.
6. Load the contents of the SqlCommand object with ExecuteReader. Return the DataTable.



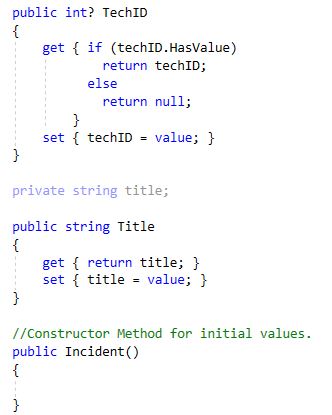
**Data Access Layer**

**Business Logic Layer**

**Incident Class**

This class is going to utilized in the FOREACH loop we use in the IncidentBLL class. Allows us to create an object of an Incident.

1. Create private variables and public property methods for all the retrieved values.
2. Easy method of doing this is doing propfull 🡪 “tab” 🡪 “tab”
3. Constructor method at the end for initial values.





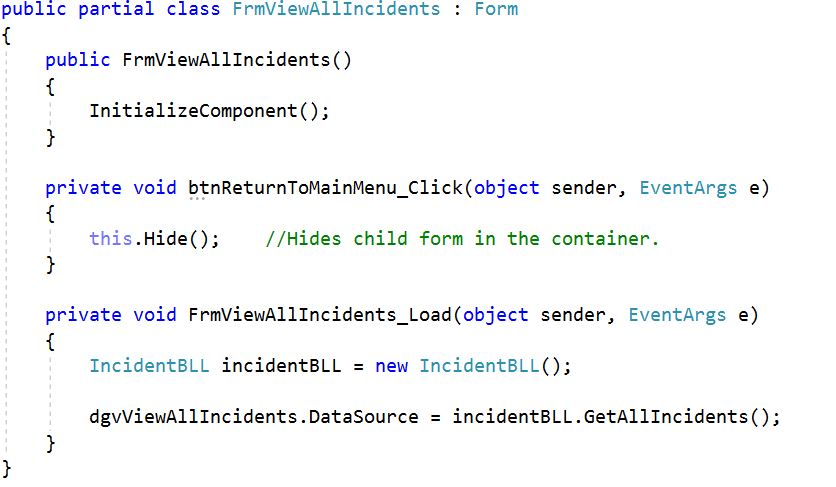
**IncidentBLL Class**

1. Create a reference to the data access layer and make a using statement (not pictured).
2. Create a new list object
3. Create a new DataTable object
4. Create a new IncidentDAL object
5. Fill DataTable with Incident info.
6. For each loop to add values. Ensure conversion methods are correct.
7. Return the list.

**USER INTERFACE LAYER**

**View All Incidents Form**

1. This.Hide(); allows the form to be hidden in the parent container when the button is clicked.
2. Create a refence and using statement for the IncidentBLL Class/Layer (not pictured).
3. Create a new IncidentBLL object.
4. Pass the incidentBLL object with incidents to the datagridview DataSource.



**Main Form**

1. Establish the form as the parent form by creating a setUpChildsForms method that receives a form.
2. Menu items clicked send a form into this method.
3. Create an exit button to exit all forms and the application

