

Encapsulation and UML Diagrams

Prepared By

Praveen Chakravarthi

Jan 28, 2022

Amazon

Customer Class

Code

```
Class Customer
{
    private string CustomerName;
    private string CustomerId;
    private int CustomerMobileNo;
    private string CustomerEmailId;
    private string CustomerAddress;

    public void AddCustomerData()
    {
        // ToDo
    }
    public void EditCustomerData()
    {
        // ToDo
    }
    public void DeleteCustomerData()
    {
        // ToDo
    }
    public void DisplayCustomerData()
    {
        // ToDo
    }
}
```

UML Diagram :

Class Customer

-CustomerName : String

-CustomerId : String

-CustomerMobileNo. : Int

-CustomerEmailId : String

-CustomerAddress : String

+AddCustomerData() : void

+EditCustomerData() : void

+DeleteCustomerData() : void
+DisplayCustomerData() : void

Employee Class

Code

```
Class Employee
{
    private string EmployeeName;
    private string EmployeeId;
    private int EmployeeMobileNo;
    private string EmployeeEmailId;
    private int EmployeeSalary;

    public void AddEmployeeData()
    {
        // ToDo
    }
    public void EditEmployeeData()
    {
        // ToDo
    }
    public void DeleteEmployeeData()
    {
        // ToDo
    }
    public void DisplayEmployeeData()
    {
        // ToDo
    }
}
```

UML Diagram :

Class Employee

-EmployeeName : String
-EmployeeId : String
-EmployeeMobileNo. : Int
-EmployeeEmailId : String
-EmployeeSalary : Int

+AddEmployeeData() : void
+EditEmployeeData() : void
+DeleteEmployeeData() : void
+DisplayEmployeeData() : void

Product Class

Code

```
Class Product
{
    private string ProductName;
    private int ProductPrice;
    private string ProductDetails;
    private int ProductReviews;
    private string ProductBrand;

    public void AddProductData()
    {
        // ToDo
    }
    public void EditProductData()
    {
        // ToDo
    }
    public void DeleteProductData()
    {
        // ToDo
    }
    public void DisplayProductData()
    {
        // ToDo
    }
}
```

UML Diagram :

Class Products

-ProductName : String
-ProductPrice : Int
-ProductDetails : String
-ProductReviews : Int

-ProductBrand : String

+AddProductData() : void

+EditProductData() : void

+DeleteProductData() : void

+DisplayProductData() : void

Orders Class

Code

```
Class Orders
{
    private string OrderName;
    private int OrderId;
    private string OrderDetails;
    private string ShippingAddress;
    private int PayableAmount;

    public void AddOrdersData()
    {
        // ToDo
    }
    public void EditOrdersData()
    {
        // ToDo
    }
    public void DeleteOrdersData()
    {
        // ToDo
    }
    public void DisplayOrdersData()
    {
        // ToDo
    }
}
```

UML Diagram :

Class Orders

-OrderName : String

-OrderId : Int

-OrderDetails : String

-ShippingAddress : String

-PayableAmount : Int

+AddOrdersData() : void

+EditOrdersData() : void

+DeleteOrdersData() : void

+DisplayOrdersData() : void

Seller Class

Code

```
Class Seller
{
    private string SellerName;
    private string SellerId;
    private int SellerMobileNo;
    private string SellerEmailId;
    private string SellerProducts;

    public void AddSellerData()
    {
        // ToDo
    }
    public void EditSellerData()
    {
        // ToDo
    }
    public void DeleteSellerData()
    {
        // ToDo
    }
    public void DisplaySellerData()
    {
        // ToDo
    }
}
```

UML Code :

Class Seller

-SellerName : String

-SellerId : String

-SellerMobileNo. : Int

-SellerEmailId : String

-SellerProducts : String

+AddSellerData() : void

+EditSellerData() : void

+DeleteSellerData() : void

+DisplaySellerData() : void

Apollo Hospital

Doctor Class

Code

```
Class Doctor
{
    private string DoctorName;
    private string DoctorSpecialisation;
    private int DoctorMobileNo;
    private int DoctorAge;
    private string DoctorAddress;

    public void AddDoctorData()
    {
        // ToDo
    }
    public void EditDoctorData()
    {
        // ToDo
    }
    public void DeleteDoctorData()
    {
        // ToDo
    }
    public void DisplayDoctorData()
    {
        // ToDo
    }
}
```

UML Code :

Class Doctor

-DoctorName : String
-DoctorSpecialisation : String
-DoctorMobileNo. : Int
-DoctorAge : Int
-DoctorAddress : String

+AddDoctorData() : void
+EditDoctorData() : void
+DeleteDoctorData() : void
+DisplayDoctorData() : void

Patient Class

Code

```
Class Patient
{
    private string PatientName;
    private int PatientSINo.;
    private int PatientMobileNo;
    private string PatientDisease;
    private string PatientAddress;
    private int PatientAge;

    public void AddPatientData()
    {
        // ToDo
    }
    public void EditPatientData()
    {
        // ToDo
    }
    public void DeletePatientData()
    {
        // ToDo
    }
    public void DisplayPatientData()
    {
        // ToDo
    }
}
```

UML Code :

Class Patient

-PatientName : String
-PatientSINo. : Int
-PatientMobileNo. : Int

-PatientDiseaseType : String

-PatientAddress : String

-PatientAge : Int

+AddPatientData() : void

+EditPatientData() : void

+DeletePatientData() : void

+DisplayPatientData() : void

Medicine Class

Code

```
Class Medicine
{
    private string MedicineName;
    private string MedicineType;
    private int MedicineCount;
    private int MedicinePrice;
    private bool MedicineAvailability;
    private int MedicineExpiry;
    private string HealthCardHolders;

    public void AddMedicineData()
    {
        // ToDo
    }
    public void EditMedicineData()
    {
        // ToDo
    }
    public void DeleteMedicineData()
    {
        // ToDo
    }
    public void DisplayMedicineData()
    {
        // ToDo
    }
}
```

UML Code :

Class Medicine

-MedicineName : String
-MedicineType : String
-MedicineCount : Int
-MedicinePrice : Int
-MedicineAvailability : Bool
-MedicineExpiry : int
-HealthCardHolders : String

+AddMedicineData() : void
+EditMedicineData() : void
+DeleteMedicineData() : void
+DisplayMedicineData() : void

Equipment Class

Code

```
Class Equipment
{
    private string EquipmentName;
    private string EquipmentType;
    private int EquipmentPrice;
    private string EquipmentStatus;
    private bool EquipmentAvailability;

    public void AddEquipmentData()
    {
        // ToDo
    }
    public void EditEquipmentData()
    {
        // ToDo
    }
    public void DeleteEquipmentData()
    {
        // ToDo
    }
    public void DisplayEquipmentData()
    {
        // ToDo
    }
}
```

```
}  
}
```

UML Code :

Class Equipment

-EquipmentName : String
-EquipmentType : String
-EquipmentPrice : Int
-EquipmentStatus : String
-EquipmentAvailability : Bool

+AddEquipmentData() : void
+EditEquipmentData() : void
+DeleteEquipmentData() : void
+DisplayEquipmentData() : void

Staff Class

Code

Class **Staff**

```
{  
    private string StaffMember;  
    private string StaffShift;  
    private int Count;  
    private int Salary;  
    private bool Availability;  
    private string Address;  
  
    public void AddStaffData()  
    {  
        // ToDo  
    }  
    public void EditStaffData()  
    {  
        // ToDo  
    }  
    public void DeleteStaffData()  
    {  
        // ToDo  
    }  
}
```

```
}  
public void DisplayStaffData()  
{  
    // ToDo  
}  
}
```

Class Staff

-StaffMember : String

-StaffShift : String

-Count : Int

-Salary : Int

-Availability : Bool

-Address : String

+AddStaffData() : void

+EditStaffData() : void

+DeleteStaffData() : void

+DisplayStaffData() : void

Police Station

Station Class

Code

```
Class Station
{
    private string StationName;
    private int StationNo.;
    private string StationAddress;
    private string DepartmentType;
    private string StationZone;
    private string StationLimits;

    public void AddStationData()
    {
        // ToDo
    }
    public void EditStationData()
    {
        // ToDo
    }
    public void DeleteStationData()
    {
        // ToDo
    }
    public void DisplayStationData()
    {
        // ToDo
    }
}
```

UML Code :

Class Station

```
-StationName : String
-StationNo. : Int
-StationAddress : String
-DepartmentType : String
-StationZone : String
```

-StationLimits : String

+AddStationData() : void

+EditStationData() : void

+DeleteStationData() : void

+DisplayStationData() : void

Staff Class

Code

```
Class Staff
{
    private string StaffMember;
    private int StaffCount;
    private string StaffShift;
    private int Salary;
    private bool Availability;
    private string Designation;

    public void AddStaffData()
    {
        // ToDo
    }
    public void EditStaffData()
    {
        // ToDo
    }
    public void DeleteStaffData()
    {
        // ToDo
    }
    public void DisplayStaffData()
    {
        // ToDo
    }
}
```

UML Code :

Class Staff

-StaffMember : String

-StaffCount : Int

-StaffShift : String

-Salary : Int

-Availability : Bool

-Designation : String

+AddStaffData() : void

+EditStaffData() : void

+DeleteStaffData() : void

+DisplayStaffData() : void

Case Class

Code

Class **Case**

```
{
    private string CaseName;
    private int CaseNo.;
    private string CaseType;
    private string Status;
    private string CaseHandler;

    public void AddCaseData()
    {
        // ToDo
    }
    public void EditCaseData()
    {
        // ToDo
    }
    public void DeleteCaseData()
    {
        // ToDo
    }
    public void DisplayCaseData()
    {
        // ToDo
    }
}
```

UML Code :

Class Case

-CaseName : String

-CaseNo. : Int

-CaseType : String

-Status : String

-CaseHandler : String

+AddCaseData() : void

+EditCaseData() : void

+DeleteCaseData() : void

+DisplayCaseData() : void

Crime Class

Code

Class Crime

```
{
    private string CriminalName;
    private int CriminalId.;
    private string CrimeDetails;
    private string CrimeType;
    private string CrimeName;

    public void AddCrimeData()
    {
        // ToDo
    }
    public void EditCrimeData()
    {
        // ToDo
    }
    public void DeleteCrimeData()
    {
        // ToDo
    }
    public void DisplayCrimeData()
    {
        // ToDo
    }
}
```

UML Code :

Class Crime

-CriminalName : String

-CriminalId. : Int

-CrimeDetails : String

-CrimeType : String

-CrimeName : String

+AddCrimeData() : void

+EditCrimeData() : void

+DeleteCrimeData() : void

+DisplayCrimeData() : void

Prison Class

Code

Class **Prison**

```
{
    private string PrisonerName;
    private int PrisonerId;
    private string PrisonerDetails;
    private int PrisonNo.;
    private string PrisonerAdress;
    private string PrisonerAge;

    public void AddPrisonData()
    {
        // ToDo
    }
    public void EditPrisonData()
    {
        // ToDo
    }
    public void DeletePrisonData()
    {
        // ToDo
    }
    public void DisplayPrisonData()
    {
    }
```

```
    // ToDo  
    }  
}
```

UML Code :

Class Prison

-PrisonerName : String
-PrisonerId : Int
-PrisonerDetails : String
-PrisonNo : Int
-PrisonerAddress : String
-PrisonerAge : String

+AddPrisonData() : void
+EditPrisonData() : void
+DeletePrisonData() : void
+DisplayPrisonData() : void

Thank You