Function Backup in Entity Framework of FormOrder:

1. Book.cs:

public override string ToString()

{

return $"Book Information:\n+ ISBN: {ISBN}\n+ Book Title: {BookTitle}\n" +

$"+ Unit Price: {UnitPrice}\n+ QOH: {QOH}\n+ Publisher ID: {PublisherId}\n+ Category ID: {CategoryId}\n+ Status: {Status}";

}

public List<Book> GetAllBooks()

{

List<Book> listB = new List<Book>();

using (HiTechDBEntities1 db = new HiTechDBEntities1())

{

var selectAll = (from b in db.Books

select b).ToList();

Book book;

foreach (var item in selectAll)

{

book = new Book();

book.ISBN = item.ISBN;

book.BookTitle = item.BookTitle;

book.UnitPrice = item.UnitPrice;

book.QOH = item.QOH;

book.PublisherId = item.PublisherId;

book.CategoryId = item.CategoryId;

book.Status = item.Status;

listB.Add(book);

}

}

return listB;

}

1. Customer.cs:

public override string ToString()

{

return $"Customer Information:\n+ Customer ID: {CustomerId}\n+ Customer Name: {CustomerName}\n" +

$"+ Street Name: {StreetName}\n+ Province: {Province}\n+ City: {City}\n+ Postal Code: {PostalCode}\n" +

$"+ Phone Number: {PhoneNumber}\n+ Contact Name: {ContactName}\n+ Contact Email: {ContactEmail}\n" +

$"+ Credit Limit: {CreditLimit}\n+ Status: {Status}";

}

public List<Customer> GetAllCustomers()

{

List<Customer> listC = new List<Customer>();

using (HiTechDBEntities1 db = new HiTechDBEntities1())

{

var selectAll = (from c in db.Customers

select c).ToList();

Customer cus;

foreach (var item in selectAll)

{

cus = new Customer();

cus.CustomerId = item.CustomerId;

cus.CustomerName = item.CustomerName;

cus.StreetName = item.StreetName;

cus.Province = item.Province;

cus.PhoneNumber = item.PhoneNumber;

cus.City = item.City;

cus.PostalCode = item.PostalCode;

cus.ContactName = item.ContactName;

cus.CreditLimit = item.CreditLimit;

cus.Status = item.Status;

cus.ContactEmail = item.ContactEmail;

listC.Add(cus);

}

}

return listC;

}

1. Employee.cs:

public override string ToString()

{

return $"Employee Information:\n+ Employee ID: {EmployeeId}\n+ Last Name: {LastName}\n" +

$"+ First Name: {FirstName}\n+ Phone Number: {PhoneNumber}\n+ Email: {Email}\n+ Job ID: {JobId}";

}

public List<Employee> GetAllEmployees()

{

List<Employee> listE = new List<Employee>();

using (HiTechDBEntities1 db = new HiTechDBEntities1())

{

var selectAll = (from e in db.Employees

select e).ToList();

Employee emp;

foreach (var item in selectAll)

{

emp = new Employee();

emp.EmployeeId = item.EmployeeId;

emp.LastName = item.LastName;

emp.FirstName = item.FirstName;

emp.PhoneNumber = item.PhoneNumber;

emp.Email = item.Email;

emp.JobId = item.JobId;

listE.Add(emp);

}

}

return listE;

}

1. Order.cs:

public List<Order> GetAllOrders()

{

List<Order> listO = new List<Order>();

using (HiTechDBEntities1 db = new HiTechDBEntities1())

{

var selectAll = (from e in db.Orders

select e).ToList();

Order order;

foreach (var item in selectAll)

{

order = new Order();

order.OrderId = item.OrderId;

order.OrderDate = item.OrderDate;

order.OrderType = item.OrderType;

order.RequiredDate = item.RequiredDate;

order.ShippingDate = item.ShippingDate;

order.OrderStatus = item.OrderStatus;

order.CustomerId = item.CustomerId;

order.EmployeeId = item.EmployeeId;

listO.Add(order);

}

}

return listO;

}

public override string ToString()

{

return $"Order Information:\n+ Order ID: {OrderId}\n+ Order Date: {OrderDate}\n" +

$"+ Order Type: {OrderType}\n+ Required Date: {RequiredDate}\n+ Shipping Date: {ShippingDate}\n+ Order Status: {OrderStatus}\n" +

$"+ Customer ID: {CustomerId}\n+ Employee ID: {EmployeeId}";

}