

Object-Aggregation

- Object aggregation is a type of composition where an object (the container) contains a collection of other objects or references to other objects. In your example, `ClassOfInvoice` is a container that holds a collection of `ClassOfInvoiceItem` objects. This is a good example of object aggregation.

Example:

```
namespace OOP_Aggregation;

public class Department
{
    private List<Employee> employees;

    public Department()
    {
        employees = new List<Employee>();
    }

    public void AddEmployee(Employee employee)
    {
        employees.Add(employee);
    }

    public void RemoveEmployee(Employee employee)
    {
        employees.Remove(employee);
    }

    public void PrintDepartmentInfo()
    {
        Console.WriteLine("Department Info:");
        foreach (var employee in employees)
        {
            Console.WriteLine($" {employee.Name} - {employee.Role}");
        }
    }
}

public class Employee
{
    public string Name { get; set; }
    public string Role { get; set; }

    public Employee(string name, string role)
    {
        Name = name;
        Role = role;
    }
}
```

```
class SVM
{
    static void Main(string[] args)
    {
        Department salesDepartment = new Department();
        salesDepartment.AddEmployee(new Employee("John Doe", "Sales Manager"));
        salesDepartment.AddEmployee(new Employee("Jane Smith", "Sales
Representative"));
        salesDepartment.AddEmployee(new Employee("Bob Johnson", "Sales
Engineer"));

        salesDepartment.PrintDepartmentInfo();
    }
}
```



In this example:

- Department is the container object that holds a collection of Employee objects.
- Employee is the contained object that represents an individual employee.
- The Department class has methods to add, remove, and print information about the employees in the department.
- The Employee class has properties for the employee's name and role.
- This is an example of object aggregation because the Department object contains a collection of Employee objects, and the Department object is responsible for managing the lifetime of those Employee objects.
- Note that in object aggregation, the contained objects (in this case, Employee objects) can exist independently of the container object (in this case, Department object). This is in contrast to object composition, where the contained objects cannot exist without the container object.