C# OOP (Object-Oriented Programming)

INSTRUCTOR: ADEEN BIN WAHEED

LECTURE 7

Inheritance

Few More Examples Demo

Polymorphism

Polymorphism is the ability of an object to take on many forms. The most common use of polymorphism in OOP occurs when a parent class reference is used to refer to a child class object.

Polymorphism Example

Let us look at an example.

public interface Vegetarian{}

public class Animal{}

public class Deer : Animal : Vegetarian

Now, the Deer class is considered to be polymorphic since this has multiple inheritance. Following are true for the above example:

A Deer IS-A Animal

A Deer IS-A Vegetarian

;

Base Keyword in C#

It is used inside a sub-class method definition to call a method defined in the superclass. Private methods of the super-class cannot be called. Only public and protected methods can be called by the base keyword.

Polymorphism

Running Demo

Composition

Composition is the design technique to implement has-a relationship in classes.

Composition takes the relationship one step further by ensuring that the containing object is responsible for the lifetime of the object it holds. If Object B is contained within Object A, then Object A is responsible for the creation and destruction of Object B. Unlike aggregation, Object B cannot exist without Object A.

Aggregation VS Composition

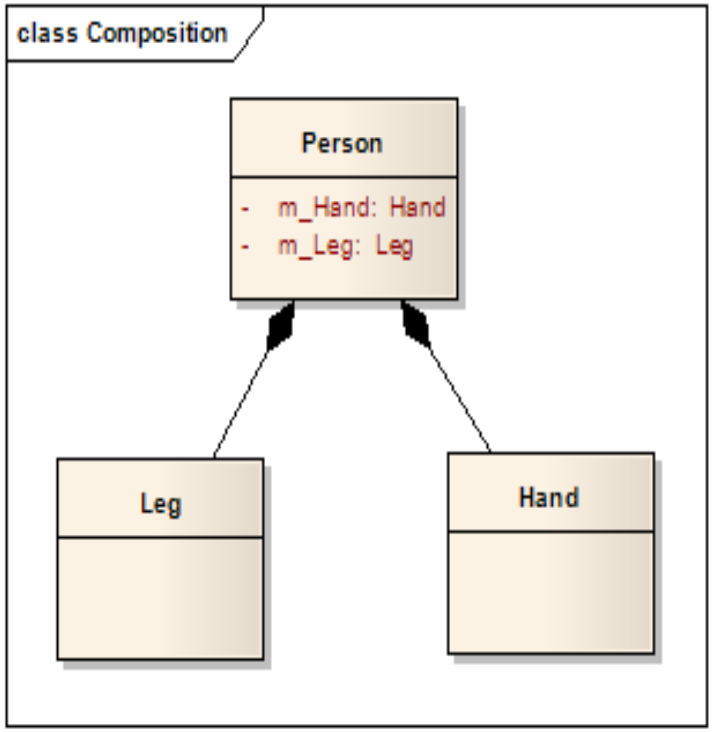
Aggregation is a specialised form of Association where all objects have their own lifecycle, but there is ownership and child objects can not belong to another parent object.

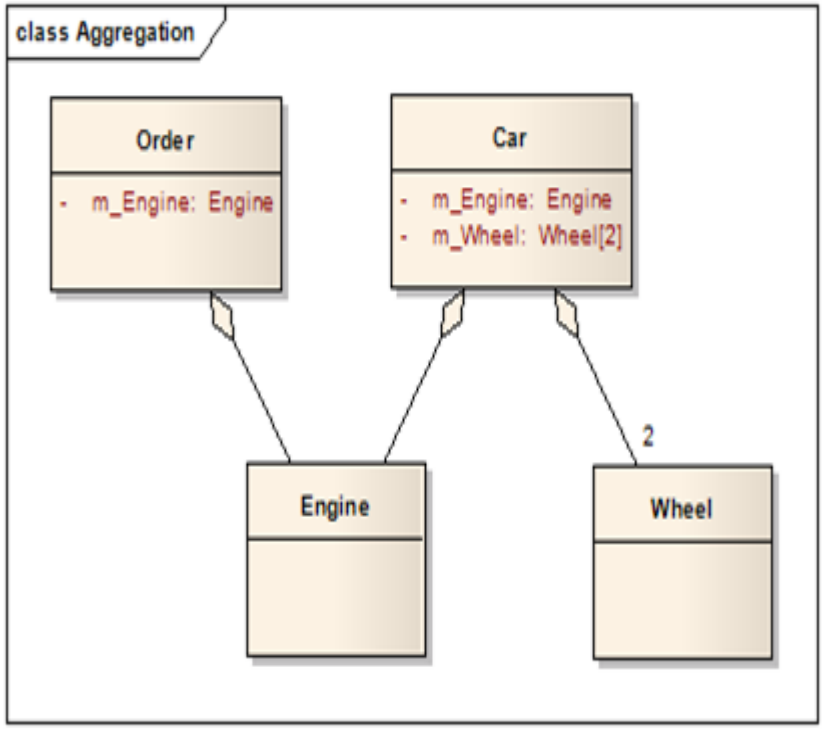
Let’s take an example of Department and teacher. A single teacher can not belong to multiple departments, but if we delete the department, the teacher object will *not* be destroyed. We can think about it as a “has-a” relationship.

Composition is again specialised form of Aggregation and we can call this as a “death” relationship. It is a strong type of Aggregation. Child object does not have its lifecycle and if parent object is deleted, all child objects will also be deleted.

Let’s take again an example of relationship between House and Rooms. House can contain multiple rooms - there is no independent life of room and any room can not belong to two different houses. If we delete the house - room will automatically be deleted.

Aggregation VS Composition





Aggregation or Composition Demo

Running Demo