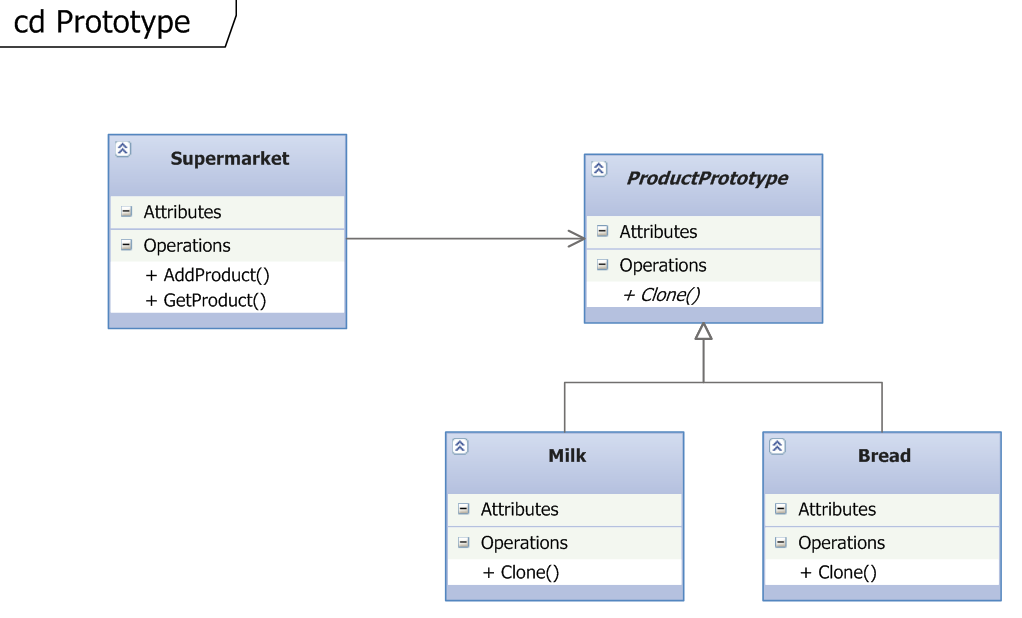
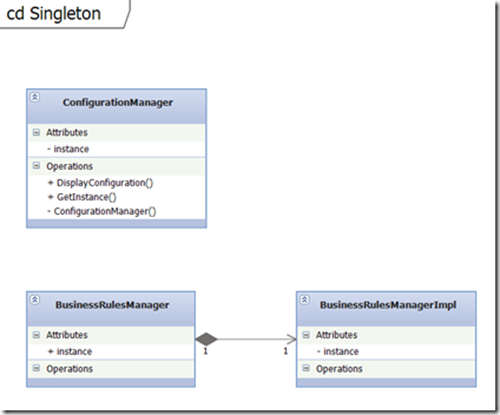
**1.Prototype Pattern (Creational Pattern): Клониране или копиране на инициализирани инстанции.**

****

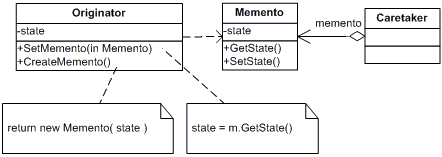
   public abstract class ProductPrototype  
    {  
        public decimal Price { get; set; }  
   
        public ProductPrototype(decimal price)  
        {  
            Price = price;  
        }  
   
        public abstract ProductPrototype Clone();  
    }

**2.Singleton Pattern (Creational Pattern): Клас с една единствена възможна инстанция.**

public sealed class ConfigurationManager  
    {  
        private static ConfigurationManager instance;  
        private static object syncRoot = new Object();  
   
        private ConfigurationManager()   
        {   
          
        }  
   
        public static ConfigurationManager GetInstance  
        {  
            get  
            {  
                lock (syncRoot)  
                {  
                    if (instance == null)  
                    {  
                        instance = new ConfigurationManager();  
                    }  
                }  
   
                return instance;  
            }  
        }  
   
        public void DisplayConfiguration()  
        {  
            Console.WriteLine("Single instance object");  
        }  
    }

****

**3.Memento (Behavioral Pattern):** Without violating encapsulation, capture and externalize an object's internal state so that the object can be restored to this state later. (звучи тъпо на български ☺ )

****

Източници:

<http://csharpdesignpatterns.codeplex.com/>

<http://www.dofactory.com/Patterns/PatternMemento.aspx#_self1>