Шаблони за дизайн

Design pattern - някакво решение, някакъв шаблон за определен проблем на програмистта. Предлага някакво **абстрактно** (т.е. не е нещо строго конкретно, като някакъв алгоритъм) решение за някакъв определен проблем

Помагат за вдигането на абстракцията и подобряването на дизайна на кода

Един design pattern има:

* Pattern name
* Problem
* Solution
* Consequences

Използват се най-вече при големи проекти. Трябва добре да преценяваме да ли има нужда от използването на design pattern-и

Creational design patterns – създават обекти в нашето приложение (напр. Factory design pattern)

Structural design patterns – грижат се за това, отделните класове и модули да са максимално добре структурирани (напр. наследяването между отделните класове да е правилно), така че приложението ни да е гъвкаво и лесно разширяемо

Behavioral design patterns – грижат се за това, отделните компоненти в програмата да си комуникират по максимално ефективен начин, съобразен със SOLID принципите

Статичните членове по природа са lazy (ако не се извикат от дадено място в кода, не се заделят(създават) в паметта)

Singleton pattern – използва се когато ни трябва да съществува само една инстанция от даден клас и държим този клас да не е статичен

Problems that Singleton patterns solves:

* Global access
* Instantiation control

**Статичните класове не могат да наследяват други класове и не могат да имплементират интерфейси, за разлика от нестатичните**

Simple factory – не е design pattern. Проста имплементация за създаване на обекти

Factory method – предимството пред Simple factory-то е, че ако искаме да добавяме нови функционалности, си имплементираме отделни класове, по наше желание, които наследяват даден абстрактен клас, който създава обекти. Така няма нужда да бъркаме във вече написания код

Ако сме сигурни, че няма да има нужда да променяме нашето factory е добре да използваме Simple factory; в този случай няма нужда да използваме Factory method. Т.е. според ситуацията използваме различно factory

Разликата между Abstract factory и Factory method е, че чрез Abstract factory създаваме няколко обекта от класове от една и съща група: например едно Abstract factory може да създава няколко вида пици (различни класове), които се различават по продукти, цена и т.н.

Builder pattern-а описва процеса за построяване на даден обект. За конкретните стъпки (тяхната последователност) се грижи един “Director” клас.