

Návrhové vzory

- Anglicky Design pattern
- Představuje obecné řešení problému
- Například šablona, popis řešení, interakce mezi třídami
- Není to algoritmus (to je konkrétní způsob řešení)

Vytvářecí vzory (Creational patterns)

Řeší problémy související s vytvářením objektů v systému. Snahou těchto vzorů je popsat postup výběru třídy nového objektu a zajištění správného počtu těchto objektů. Většinou se jedná o dynamická rozhodnutí učiněná za běhu programu.

1. Abstraktní továrna
 - Definuje rozhraní pro vytváření rodin objektů
 - Create button <- příklad, nezajímá mě jak ho vytvořím
 - Aplikace nemá tušení jaký druh GUI továrna vytvoří
2. Object pool
 - Bazén objektů, které použijeme při potřebě
 - Půjčovna kostýmů – půjčím si kostým (objekt), upravím ho a použiji, vrátím do poolu
 - Umožňuje sdílení zdrojů pomocí recyklace objektů
 - Při nedostatku prostředků pořízení nových nebo časová prodleva
3. Prototyp
 - Nové objekty se vytváří kopírováním celého objektu (klon) nebo jeho částí

Strukturální vzory (Structural patterns)

Představují skupinu návrhových vzorů zaměřujících se na možnosti uspořádání jednotlivých tříd nebo komponent v systému. Snahou je zprehlednit systém a využít možností strukturalizace kódu.

1. Adaptér
 - API
 - Převádí rozhraní jedné třídy na rozhraní druhé třídy
2. Proxy
 - Nabízí zástupný objekt za nějaký jiný pro kontrolu přístupu k danému objektu
 - Využití v internetu, všechny lokální (neveřejné) IP adresy přistupují k internetu přes Proxy server pod jedinou IP adresou

Vzory chování (Behavioral patterns)

Zajímají se o chování systému. Mohou být založeny na třídách nebo objektech. U tříd využívají při návrhu řešení především principu dědičnosti. V druhém přístupu je řešena spolupráce mezi objekty a skupinami objektů, která zajišťuje dosažení požadovaného výsledku.

1. Iterátor
 - Umožňuje přístup k elementům skupinového objektu postupně bez toho, abyste vystavovali vnitřní reprezentaci tohoto objektu
2. Prostředník
 - Umožňuje zajistit komunikaci mezi dvěma komponentami programu, aniž by byly v přímé interakci a tím musely přesně znát poskytované metody

Souběžné vzory (Concurrent patterns)

Vzory řešící problémy vzniklé spouštěním programů ve vláknech, a tudíž při souběžném řešení úlohy. Např. 2 aktualizací SQL dotazy.

1. Guarded
 - Jde o vzor obstarávající operace, které požadují uzamčení, a navíc mají nějakou podmínku, která musí být splněna předtím, než může být operace provedena
2. Thread pool
 - V bazénku vláken je vytvořen nějaký počet vláken pro řešení nějakého množství úloh, které jsou organizovány ve frontě (FIFO). Zpravidla je výrazně více úloh než vláken