

Het Template Method Pattern

Objectgeoriënteerde analyse en ontwerp

Het Template Method Pattern

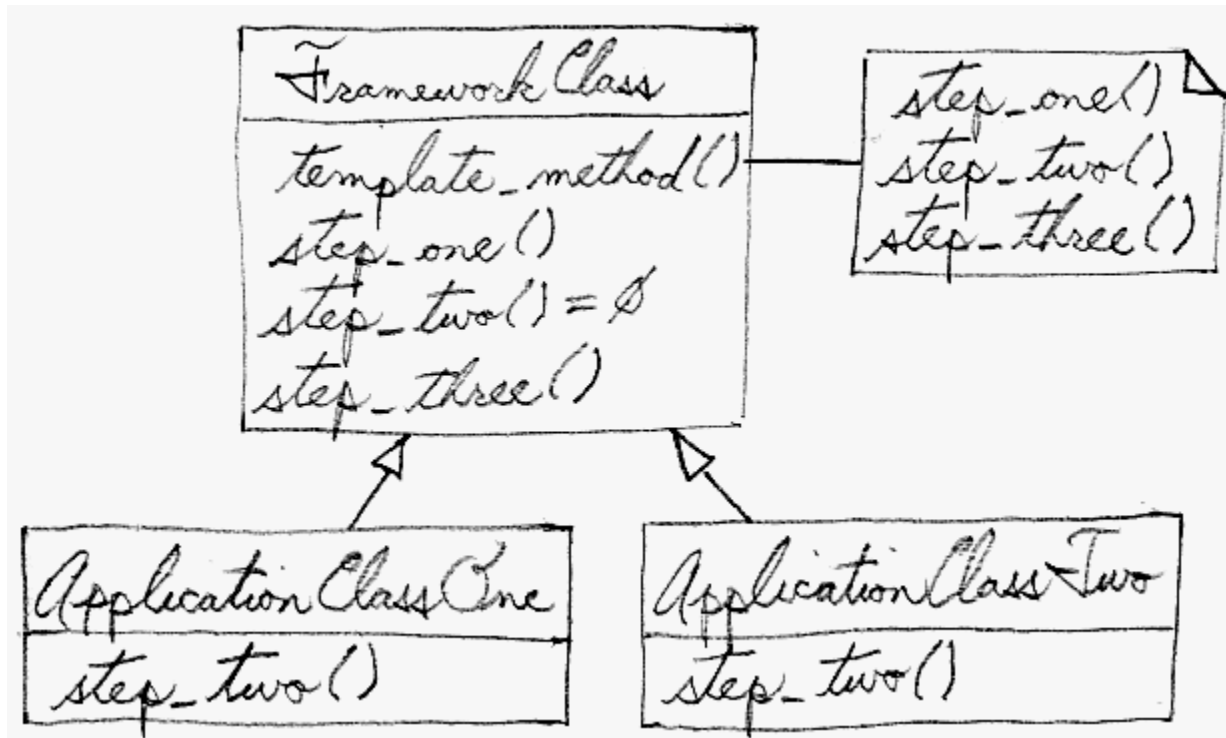
Doel

Een **algoritme** uitvoeren, waarbij **delen** van het algoritme aan subklassen **uitbesteed** kunnen worden.

De basisstructuur van het algoritme ligt vast.

Enkel de concrete uitvoering van bepaalde stappen kan veranderd worden.

Basisstructuur van het pattern



Basisstructuur van het pattern

- Het pattern werkt met **overerving**
- De (abstracte) basisklasse legt de structuur van het algoritme vast in de **template methode**.
Die is **final** zodat subklassen die niet kunnen wijzigen.
- Sommige stappen in het algoritme zijn abstract. Die moeten door subklassen concreet gemaakt worden.

Voordelen van het pattern

- Het algoritme staat maar op 1 plaats (onderhoudsvriendelijk)
- Er is een soort framework. Nieuwe concrete implementaties zijn gemakkelijk toe te voegen.

Hooks

- Een **hook** is een stap in het algoritme waar een subklasse de **optie** heeft om die te overriden.

Als dat niet gebeurt, heeft de basisklasse een defaultimplementatie (die kan ook leeg zijn).

Dependency inversion



Dit pattern past het
dependency inversion
principe toe.
Hoezo?