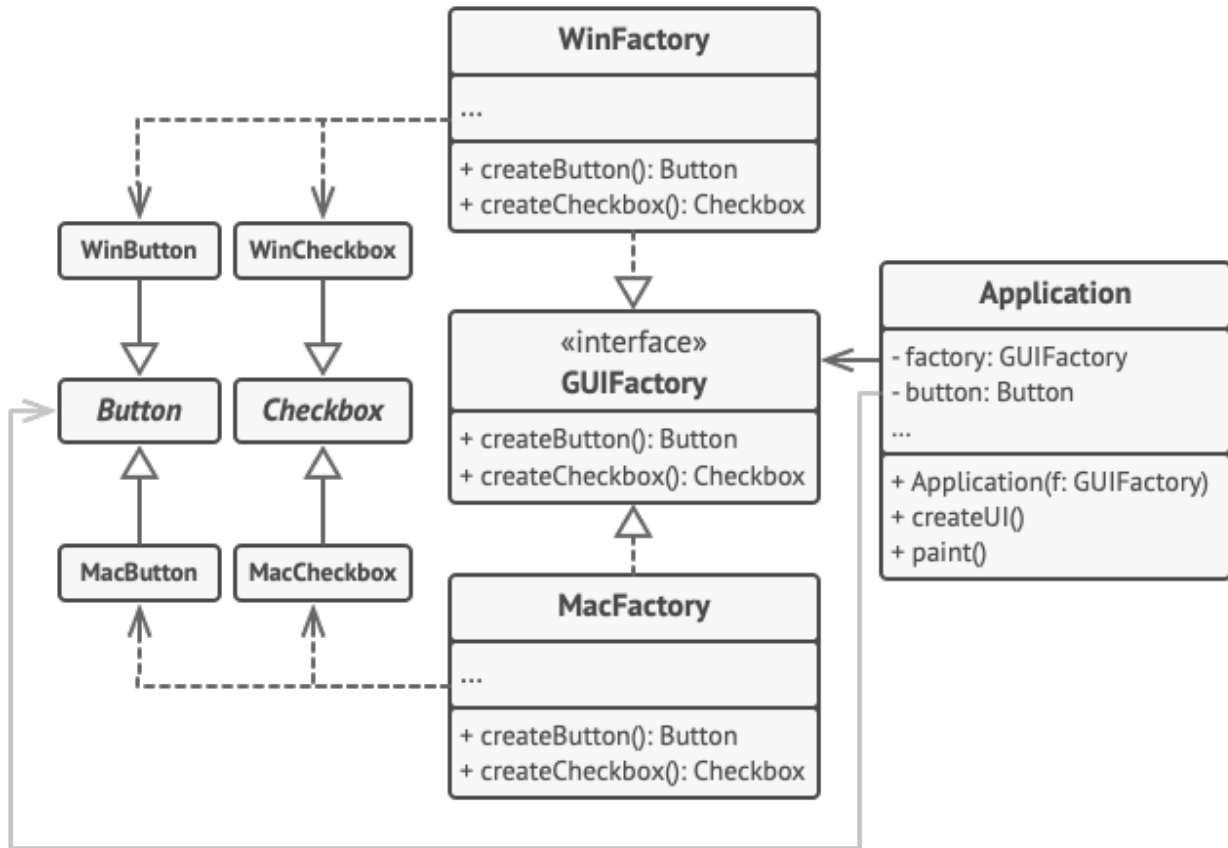


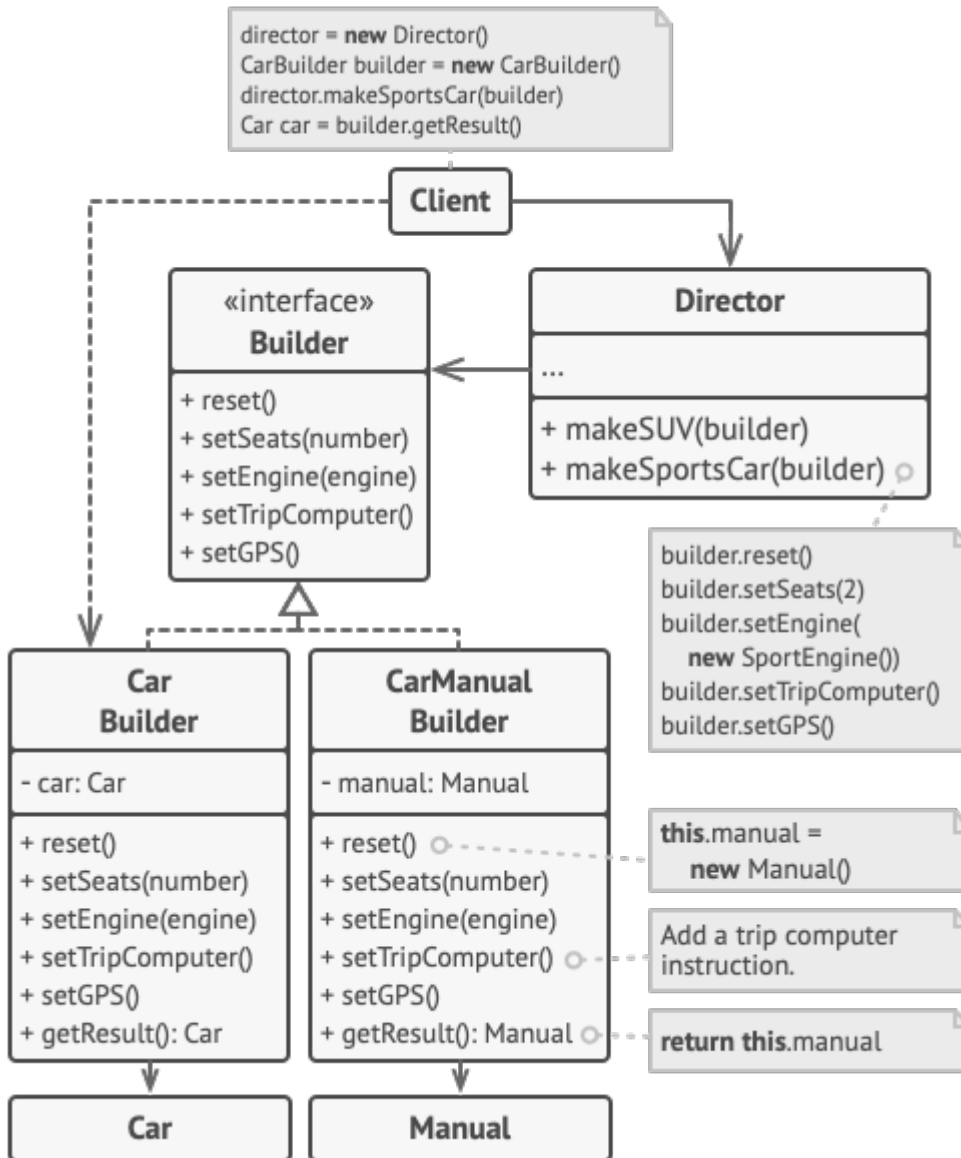
2. Abstract Factory



L'Abstract Factory fournit une interface pour créer des familles d'objets liés ou dépendants sans spécifier leurs classes concrètes. Cela permet de créer des objets avec une cohérence globale.

Enoncé: Dans un jeu vidéo, vous avez besoin de créer des personnages et des armes. Utilisez l'Abstract Factory pour créer des personnages et des armes qui correspondent les uns aux autres (par exemple, un personnage guerrier avec une épée).

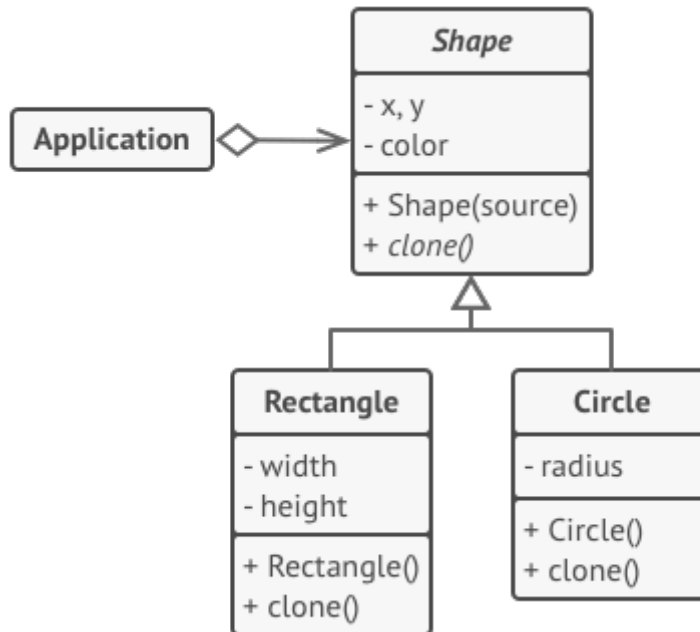
3. Builder



Le pattern Builder permet de construire un objet complexe étape par étape. Il sépare la construction d'un objet de sa représentation.

Enoncé: Créez un générateur de rapports qui vous permet de construire des rapports personnalisés en spécifiant des sections, des en-têtes, des images, etc. Le Builder vous aide à construire un rapport par étapes.

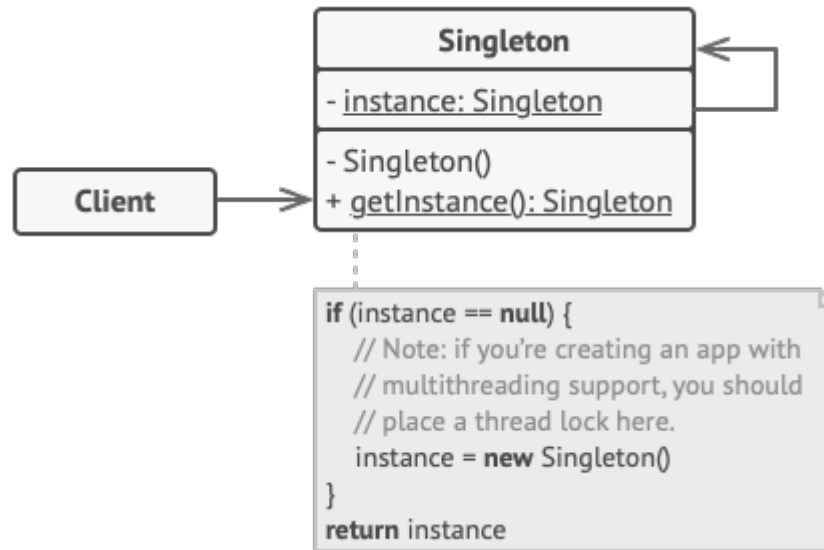
4. Prototype



Le pattern Prototype permet de créer de nouveaux objets en copiant un objet existant, appelé le prototype. Cela évite la création d'objets coûteux à partir de zéro.

Enoncé: Dans un système de gestion de documents, utilisez le Prototype pour cloner un document existant et créer de nouvelles instances avec des modifications mineures.

5. Singleton



Le Singleton garantit qu'une classe n'a qu'une seule instance et fournit un point d'accès global à cette instance. Cela est utile lorsque vous avez besoin d'une unique ressource partagée.

Enoncé: Créez un gestionnaire de configuration qui stocke les paramètres de l'application. Le Singleton garantit qu'il n'y a qu'une seule instance de ce gestionnaire et que les paramètres sont cohérents dans toute l'application.