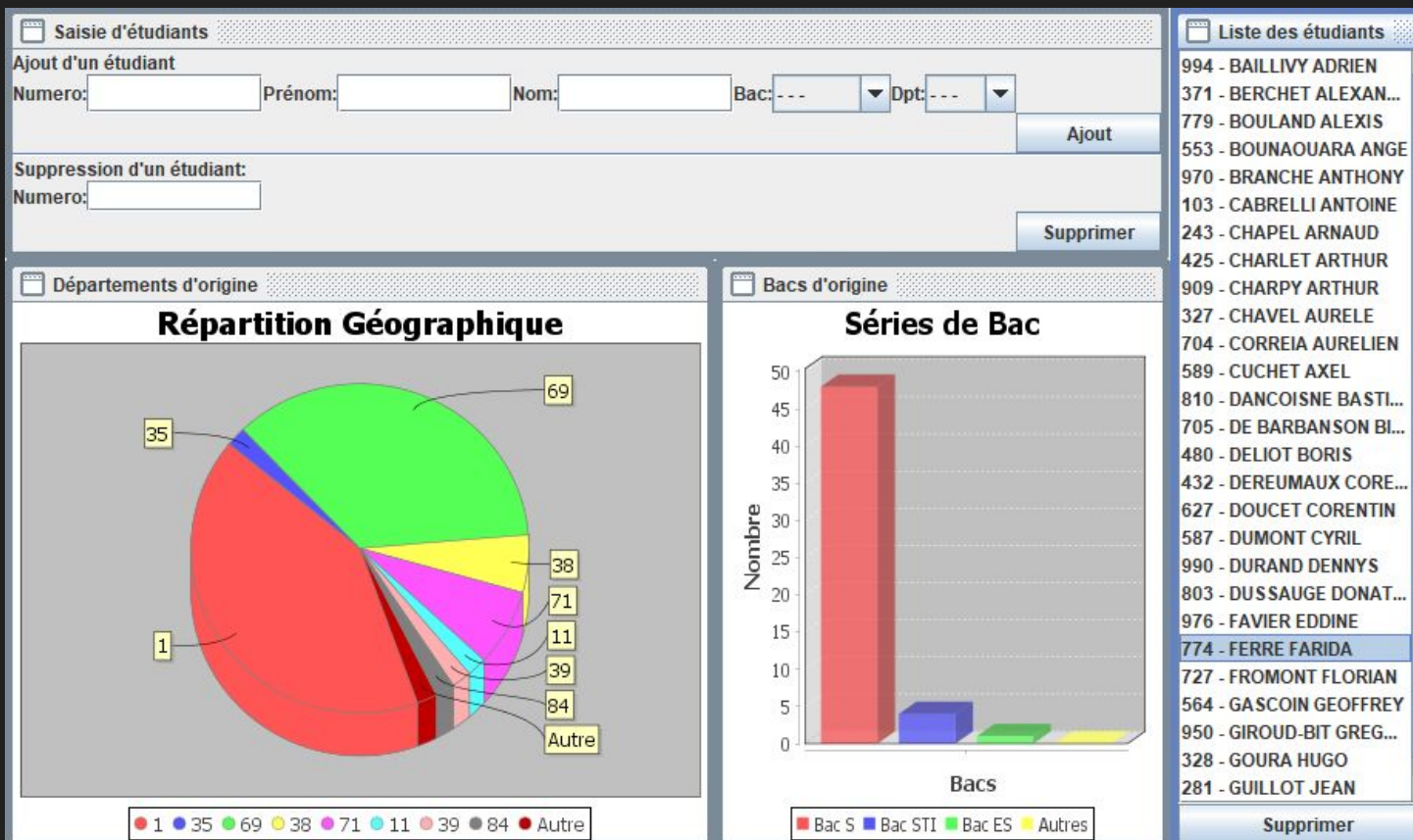


Pattern Observer

Intention > Mise en oeuvre d'une solution d'interdépendance de type un à plusieurs. Quand un objet va changer d'état, tous ceux qui en dépendant vont être notifié et mis à jour automatiquement.

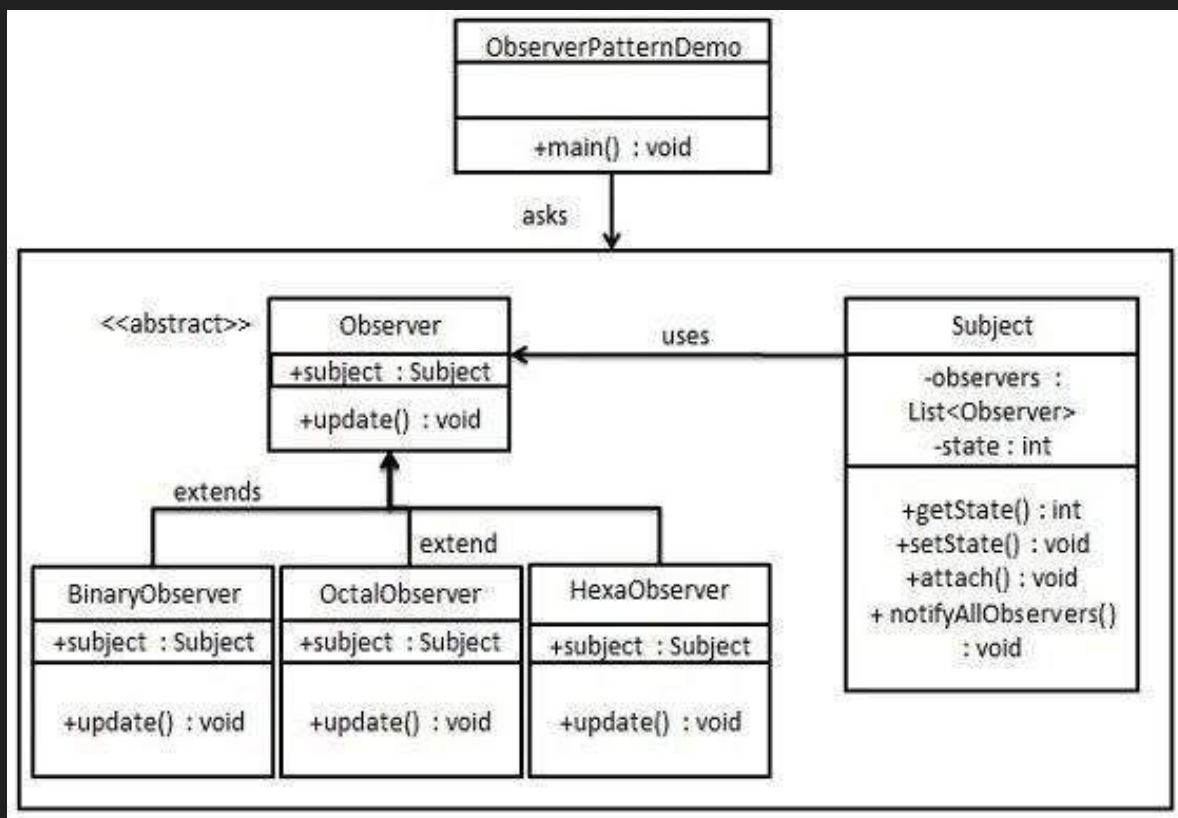


Avantages >

- Supporte le principe de perte de couplage entre deux objets
- Envoyer des données d'autres objets sans changer les classes Subject ou Observer
- Les "observers" peuvent être supprimé ou ajouté à n'importe quelle moment

Désavantages >

- L'interface "Observer" doit être implémenté par ConcreteObserver
- Si c'est pas correctement implémenté les "Observers" peuvent ajouter de la complexité et des baisses de performances



lien du code :

<http://design-patterns.fr/observateur-en-java>