

JAVA 12. WORKSHOP



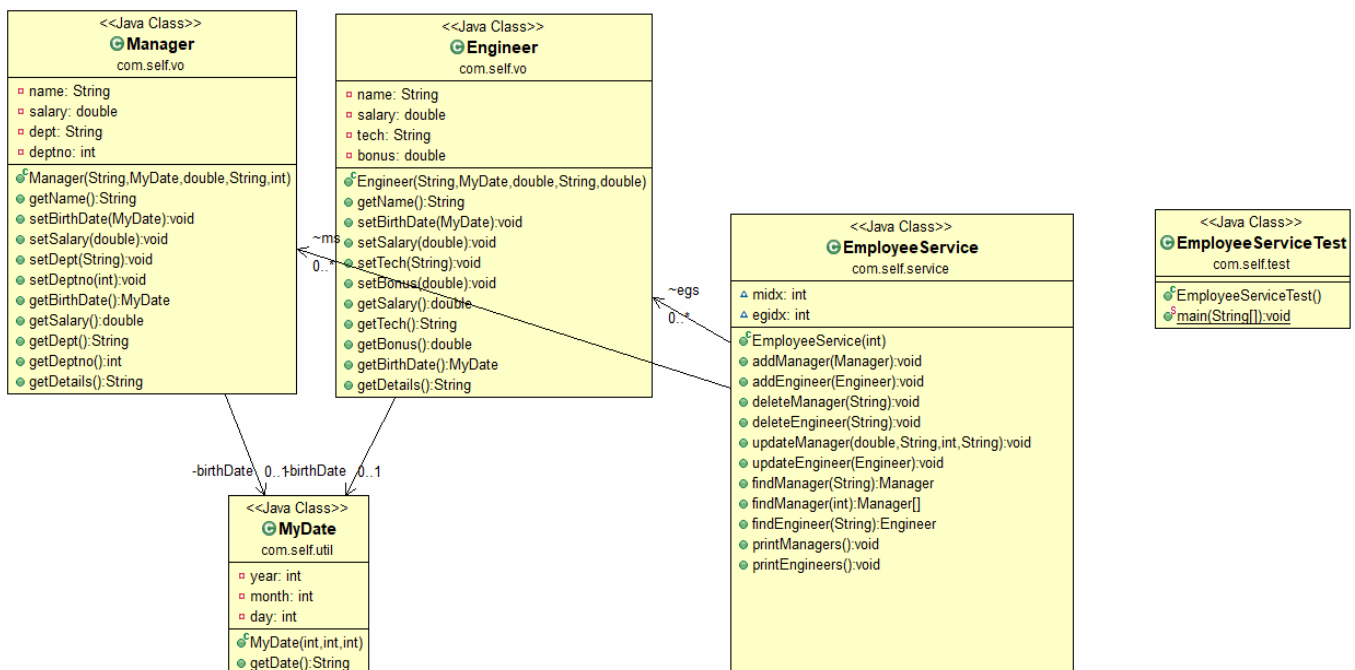
1

- ✓ 추상화를 이해하고 설계할 수 있다.
- ✓ Super Class Type으로 단일한 관리를 할 수 있다
- ✓ Polymorphic Argument를 적용할 수 있다
- ✓ Object Casting기법을 활용할 수 있고 instanceof 키워드를 적절하게 사용할 수 있다

아래 UML은 상속기법을 적용하지 않은 Diagram입니다. 추상화 과정을 거쳐서 Super 클래스를 도출하고 Service 클래스의 CRUD기능을 Employee[] 타입으로 단일한 관리가 되도록 기능을 수정합니다. 아래를 참고하세요.

작업 순서

1. 아래 UML을 상속기반의 UML로 다시 수정한다.
2. Secretary 클래스를 추가하고 Secretary에 맞는 필드와 기능을 추가한다.
3. Manager, Engineer, Secretary클래스로부터 일반화 과정을 거친 Super class를 작성하고 상속을 적용한다.
4. EmployeeService 클래스 기능은 Manager, Engineer, Secretary 서로 다른 자식 클래스들을 단일하게 관리하기 위해서 Employee[] 타입을 사용한다.
5. EmployeeService 클래스 기능을 완성하기 위해서 Polymorphic Argument, Object Casting, Overriding, Overloading 같은 기법들을 적극적으로 활용한다



기존 UML을 변경한 이미지 파일과 함께 새롭게 변경, 추가된 자바파일을 슬랙 다이렉트 메시지로 파일 업로드 해주기 바랍니다.

(PNG, Employee.java , EmployeeService.java, EmployeeServiceTest.java)