< 2023.08.30 >

**< 6 교시 > 테스트 주도 개발**

1. TDD (Test Driven Development)

* 테스트 주도 개발
* 리팩토링이 포함된 개념
* 테스트 코드를 먼저 작성하고 실패한 테스트를 최대한 빠르게 통과시키기 위한 기능만 구현, 그러고 나서 리팩토링

1. 테스트 코드 먼저 작성. 당연히 실패함

2. 테스트를 통과하기에 충분한 코드만 작성

3. 테스트 코드를 변경하지 않고 코드 개선

* 개발 V모델에 가장 부합하는 테스트

2. ATDD (Acceptance Test Driven Development)

* 인수 테스트 주도 개발
* TDD의 범위를 코드 수준에서 기능 테스트 수준까지 확장한 것
* UI 레벨이 아닌 API 레벨에서 테스트 작성

3. UAT (User Acceptance Test)

* 실제 사용자 환경에서 사용자의 입장으로 테스트하는 것
* 시스템의 인수를 위해 기능적/비기능적 요구 사항을 사용자가 직접 테스트하여 개발이 완료되었음을 증명하는 테스트

==================================================

**< 7 교시 > 소스코드 분석 실습**

1. 시퀀스 다이어그램(Sequence Diagram)

* UML(Unified Modeling Language, 통합 모델링 언어)
* 특정 Use Case에 대한 상세한 흐름
* 다른 객체들 간의 호출관계
* 수직은 발생 시간 순서로 메시지/호출 시퀀스, 수평은 메시지가 전송되는 객체 인스턴스

2. 클래스 다이어그램(Class Diagram)

* UML(Unified Modeling Language, 통합 모델링 언어)
* 엔티티들 사이의 관계 (클래스들 간의 상속 관계)
* 시스템의 정적 구조