

2.5. 클래스 다이어그램

2.5.1. 정적 모델링의 개념

정적 모델링은 사용자가 요구한 기능을 구현하는데 필요한 자료들의 논리적인 구조를 표현한 것이다.

- 시스템에 의해 처리되거나, 생성될 객체들 사이에 어떤 관련이 있는지를 구조적인 관점에서 표현한다.
- 객체들을 클래스로 추상화하여 표현한다.
- UML을 이용한 정적 모델링의 대표적인 것이 **클래스 다이어그램**이다.

2.5.2. 클래스 다이어그램의 개념

시스템을 구성하는 **클래스**, 클래스의 특성인 **속성**과 **오퍼레이션**, 속성과 오퍼레이션에 대한 **제약 조건**, 클래스 사이의 **관계**를 표현한 것이다.

- 시스템을 구성하는 요소에 대해 이해할 수 있는 구조적 다이어그램이다.
- 시스템 구성 요소를 문서화하는 데 사용된다.

2.5.3. 클래스

각각의 객체들이 갖는 속성과 오퍼레이션을 표현한다.

- 일반적으로 3개의 구획으로 나뉘 클래스의 **이름**, **속성**, **오퍼레이션**을 표기한다.
- 속성이나 오퍼레이션은 생략할 수 있지만, 이름은 반드시 명시해야 한다.
- 속성이나 오퍼레이션이 생략된 경우, 구획선을 그리지 않아도 된다.

2.5.3.1. 속성

[접근제어자] 속성명 : 자료형 [다중성] [=초기값]

클래스의 상태나 정보를 표현한다.

- **접근제어자**
 - 속성과 오퍼레이션을 어느 정도까지 외부에 노출시킬지를 제어한다.
- **속성명**
 - 속성의 이름으로, 사용자가 임의로 작성한다.
- **자료형**
 - UML에서 기본적으로 제공하는 자료형 또는 사용자가 필요에 의해 새롭게 정의한 자료형을 사용할 수 있다.
 - **다중성**
 - 동일한 속성명으로 여러 개의 속성 값을 가질 수 있는 것으로, 배열과 같은 의미이다.
- **초기값**
 - 데이터를 입력하지 않았을 때 기본적으로 입력되는 값을 지정한다.

2.5.3.2. 오퍼레이션

[접근제어자] 오퍼레이션명 (매개변수1 : 자료형1, 매개변수2 : 자료형2, ...) : 반환자료형

클래스가 수행할 수 있는 동작으로, 함수라고도 한다.

- **오퍼레이션명**

- 오퍼레이션의 이름으로, 사용자가 임의로 작성한다.
- **매개변수**
 - 오퍼레이션으로 어떤 작업을 할 때, 해당 오퍼레이션 수행에 필요한 값을 전달하기 위해 사용된다.
- **반환자료형**
 - 오퍼레이션 수행 수 반환되는 값에 대한 자료형으로, 반환되는 값이 없으면 반환자료형을 void로 지정한다.

2.5.4. 제약조건

속성에 입력될 값에 대한 제약조건이나, 오퍼레이션 수행 전후에 지정해야 할 조건이 있다면 이를 적는다.

- 주석 도형 안에 제약조건을 적은 후, 제약조건이 적용될 속성이나 오퍼레이션을 점선으로 연결한다.
- 클래스 안에 제약조건을 적을 때는 중괄호 {}를 이용한다.

2.5.5. 관계

클래스와 클래스 사이의 연관성을 표현한다.

- 클래스 다이어그램은 클래스가 서로 연결되어 있음을 의미하는 연관 관계를 기본으로 한다.
- 관계에 참여하는 객체의 수를 연관 관계 선 위에 표기한다.
- 클래스 다이어그램에 표현하는 관계에는 **연관 관계, 집합 관계, 포함 관계, 일반화 관계, 의존 관계**가 있다.

2.5.5.1. 연관 관계

- 두 클래스 간의 관계를 명확하게 표현하기 위해, <관계 표현 실선의 중간 지점에 관계의 이름을 표기할 수 있다.
- 해당 클래스의 관계 표현 실선 바로 옆에 클래스의 역할을 표기할 수 있다.
- 연관 관계에서 추가적으로 표현해야 할 속성이나 오퍼레이션이 있는 경우, 연관 클래스를 사용한다.
- 두 클래스가 서로 연관 관계에 있을 때는 클래스 안에 연관된 클래스를 이용하여 객체 변수를 생성할 수 있다.

2.5.5.2. 집합 관계

두 클래스가 서로 집합 관계에 있을 때는 집합 관계에 있는 클래스의 객체 변수를 매개변수로 사용할 수 있다.

2.5.5.3. 포함 관계

두 클래스가 서로 포함 관계에 있을 때는 포함 관계에 있는 클래스를 이용하여 생성된 객체 변수를 이용하여 새로운 객체 변수를 생성할 수 있다.

2.5.5.4. 일반화 관계

- 일반적인 개념을 상위(부모) 클래스, 구체적인 개념을 하위(자식) 클래스라고 부른다.
- 두 클래스가 서로 일반화 관계에 있을 때는 하위 클래스가 상위 클래스의 속성이나 메소드를 사용할 수 있다.

2.5.5.5. 의존 관계

두 클래스가 의존 관계에 있을 때는 영향을 주는 클래스의 특정 오퍼레이션이 수행될 때만 영향을 받는 클래스가 사용된다.