

1. Class

1-1. Instantiation

2. MVC 패턴

1. Class

언어에서 의미는 나와 상대방이 공통으로 지칭하는 것을 말한다. 내 머릿속 생각을 컴퓨터에 전달할 때, 컴퓨터가 나의 뜻을 이해하려면 기계어로 번역이 되어야 한다. 컴퓨터의 의미는 기계어로 표현된다.

자바의 언어적 체계에 있어서 객체 지향 언어는 기본적으로 가장 기본적인 단위를 클래스로 잡는다. 클래스는 단어를 만드는데, 단어를 만드는 방법은 명확해야 한다. 클래스를 정의할 때는 행위적, 구조적, 속성적 세 가지로 정의할 수 있다. 행위적으로 정의하는 것은 무엇을 하느냐가 중요하고, 구성적 설명은 무엇으로 구성되어있는지가 중요하다. 속성적으로 정의하는 것은 그것이 가진 속성으로 정의하는 것을 말한다.

1-1. Instantiation

클래스는 타입을 얘기한다. 타입은 사람이 가상의 세상을 만들어내기 위해 만드는 것이다. 사람은 어떠한 장면을 볼 때 재사용할 수 있는 부분을 뜯어내듯이 기억한다. 즉, 타입은 실체들의 공통적인 추상화(특징)이다. 타입은 객체(인스턴스)를 만들 때 쓴다. 수학에서는 정수를 타입이라고 할 수 있으며, 정수는 머릿속에 존재하는 추상적인 집합이다. 정수나 실수처럼, 안에 있는 원소들이 연산했을 때, 여전히 원소가 그 안에 속하는 것을 닫힌 집합이라고 한다. 객체, 타입이라는 것은 정수나 실수 같은 닫힌 집합처럼 공통의 Operation(연산자)을 갖는다.

Instantiation이란, “객체 지향 프로그래밍에서, 특정한 클래스 또는 프로세스에 대하여 인스턴스를 만들어내는 일을 말한다. 물리적 공간에 영역을 할당하여, 실제로 이름을 가진 독립된 객체를 다룰 수 있게 된다.”[네이버 국어사전]

여기서 인스턴스(instance)는 “같은 클래스에 속하는 개개의 객체로, 하나의 클래스에서 생성된 객체를 말한다. 즉 클래스가 구체화되어, 클래스에서 정의된 속성과 성질을 가진 실제적인 객체로 표현된 것을 의미한다.”[네이버 지식백과] 즉 객체는 ‘실체’이다. 객체는 프로그램에서 행위를 하는 행위이다. 행위를 정의하려면, 예를 들어 등장인물이 나오고 등장인물의 대화가 나와야 한다.

즉 Instantiation은 클래스에서 객체로 나오는 과정, 실체를 만들어내는 과정이다. new는 클래스에게 내리는 명령어이며, 메모리에 객체를 만들어놓는다. 일을 시키기 위해 메모리에서 꺼내려면 지칭하는 이름을 정의해야 한다. 이름이 메모리에 주소를 저장하고 한다. 이름을 부르면 메모리로 가서 객체의 위치를 정확하게 알게 된다. 이를 Binding(연결)이라고 한다.

new라고 명령하면 메모리에 오브젝트가 생긴다. 클래스명은 오브젝트를 담는 그릇이다. 컴퓨터에서는 메모리 주소를 이야기한다. new라고 명령하면 주소가 반환된다. 이름은 주소를 할당하는 공간이다. 안에 주소가 들어간다. 이름 혹은 포인터라고 한다. 객체 중 지속성이 있는 벨류 오브젝트들은 생성과 소멸이 의미가 없다. 값이 있기만 하면 된다. 그러나 상태를 유지할 때는 이름이 필요하다. 클래스만이 new 명령을 할 수 있다(객체를 만들 수 있다.).

2. MVC 패턴

MVC 패턴은 Model - View - Controller로 이루어진 설계방식이다. “노르웨이 컴퓨터 과학자인 트리베 린스카우그(Trygve Reenskaug)가 1970년대에 최초로 MVC 개념을 제안하였다.”[IT용어사전]

View는 사용자 인터페이스(화면), Model은 데이터 관리자, Controller는 View에 필요한 정보를 수집해서 전달하는 역할을 한다.

Controller는 “Model에 명령을 보냄으로써 Model의 상태를 변경할 수 있다. 또, Controller가 관련된 View에 명령을 보냄으로써 Model의 표시 방법을 바꿀 수 있다.”[위키백과] View와 1 to 1으로 만들어야 한다.

Model은 “Model의 상태에 변화가 있을 때 Controller와 View에 이를 통보한다. 이와 같은 통보를 통해서 View는 최신의 결과를 보여줄 수 있고, Controller는 Model의 변화에 따른 적용 가능한 명령을 추가·제거·수정할 수 있다.”[위키백과]

View는 “사용자가 볼 결과물을 생성하기 위해 모델로부터 정보를 얻어 온다.”[위키백과]

우리가 만들 수강신청 시스템은 사용자에게 수강신청을 할 수 있는 화면을 보여주고 사용자가 수강신청 목록을 불러오거나 장바구니 목록을 불러오는 등의 기능이 있어야 한다. 수강신청 시스템을 Model - View - Controller 패턴으로 구현할 수 있다.