3주차 예비보고서

전공: 아트엔테크놀로지 학년: 3학년 학번: 20191172 이름: 함승우

1. OOP

OOP란 (Object-Oriented-Programming)으로 컴퓨터 프로그램을 명령어의 목록으로 보는 시각(절차 지향)에서 벗어나, 여러 개의 ‘객체’들의 상호작용으로 프로그램을 로직을 구성하는 패러다임이다. 즉 프로그래밍에서 필요한 데이터를 추상화해서 속성과 매서드를 가진 객체를 만들고, 이 객체 간의 상호작용을 통해 로직을 구성한다.

2. 객체

객체는 현실에 존재하는 대상이다. Object이다. 객체는 그 내 부에 자료형과 함수가 존재한다. 예를 들어, ‘핸드폰’이라는 객체 모델을 만든다면, 제조사, 모델명, 용량 등과 같은 자료형 뿐만 아니라, 핸드폰 기능 사용하기, 핸드폰으로 사진찍기 등의 함수들 역시 ‘핸드폰’이라는 객체로 묶어져 있다.

3. 클래스

클래스는 객체에 대한 설계도라고 할 수 있다. OOP에서 특정 객체를 생성하기 위해 변수와 메소드를 정의하는 틀이다.

4. 인스턴스

객체 중 클래스를 생성하고 클래스 객체를 사용할 수 있게 만든 것이 인스턴스이다. 클래스를 바탕으로 만든 구체적인 실체이다.

5. OOP를 쓰는 이유

이전 패러다임인 절차 지향 프로그래밍은, 함수나 자료형이 따로따로 존재하여 이 둘을 따로 생각하여 프로그래밍을 해야 하고, 서로 관련 없는 로직(논리적으로 연결 되지 않는 로직)들이 하나의 프로그램 안에서 동작할 때 서로 영향을 끼치는 경우도 많다. 객체 지향 프로그래밍은 별개의 로직들을 논리적으로 객체로 묶어 프로그래밍 할 수 있다. OOP를 사용하면 코드를 재사용할 수 있고, 객체와 객체 간에는 독립적이기 때문에 디버깅에 용이하다.

6. 상속

객체들 간의 관계를 구축하는 방법인데, 부모 클래스의 메소드 혹은 필드를 자식 클래스에 물려주는 것을 뜻한다. 추가로, 기존 클래스에 기능을 추가하거나 재정의를 하는 방법이라고 할 수 있다. 앞에서 이야기한 코드가 재사용될 수 있다는 점이 바로 상속으로 인한 것이다. 객체들 간의 공통된 로직을 공유시켜 새로운 객체를 만든다거나, 코드를 재사용 하는 것이 상속이라고 볼 수 있다.

7.

class Array에 대해서, int \*data로 배열에 메모리를 동적으로 할당받아 자료를 저장한다. Array는 배열의 크기(size)를 인자로 받아 data를 할당한다. int& operator는 새로운 값은 left 값인데, 배열에 값을 할당하는데 사용될 수 있도록한다, int operator는 배열을 읽는데 필요한 값이다.

class RangeArray,나머지 다 동일하지만 Rangearray()는 len을 조절하며 메모리를 저장한다. 추가로 class Array와는 다르게 int& operator는 data의 위치를 변한하지 않고, clsass RangeArray에서는 data[i-low]의 위치를 변환한다.