1. 값 형식과 참조 형식을 각각 설명하고 차이점을 설명해 보세요. 구체적인 예를 들어보세요.

값형식은 일반적으로 선언하는 double , int , float….등등 각각의 값을 할당 받는 형식이다.

참조 형식은 위처럼 값을 할당 받은 데이터의 주소 값만 참조하여 가져오는 형식이다.

예를 들어

int a = 3;

int b = a;

위의 각각은 값형식이다.

int[] a = new int[2] {0,1};

int[]b = a;

여기서 b는 a의 주소 값을 참조하는 참조형식이다.

2. 암시적인 형변환과 명시적인 형변환의 차이는 무엇인가요?

선조클래스로부터 파생된 클래스를 선조클래스로 변환할 시에는 암시적으로 변환이 가능하다.

하지만 선조클래스를 파생된 클래스로 변환할 시에는 명시적인 형변환이 필요하다.

3. 기반 클래스와 파생 클래스는 서로 간의 형 변환에 가능할 수도, 없을 수도 있다. 예를 들어 파생 클래스를 기반 클래스로 형 변환하는 의미는 어떤 것이 있을까요?

파생 클래스는 기반클래스를 상속받음으로 기반클래스의 성질을 가지고 있다. 이 때 형변환을 통하여 파생클래스가 가지고 있는 특성을 여러 클래스에 활용 할 수 있게 된다.

4. “is”, “as” 연산자에 관해서 각각의 예와 활용법에 관해서 설명하세요. 단 설명할 때 적절한 상속관계를 가지는 예를 사용하세요. 수업시간에 사용한 Car와 Vehicle은 사용하지 마세요.

“is” 연산자는 클래스가 형변환이 호환되는지 확인해주는 연산자이다.

“as” 연산자는 클래스가 형변환이 되도록 해주는 연산자이다.

예를 들어

Class Cellphone

{

}

Class Galaxy

{

}

Cellphone cell = new Cellphone();

Galaxy gal = new Galaxy();

If(gal is cell)

{

………

}

Else

{

………

}

//“is”연산자는 이런 식으로 사용이 가능하고

cell = gal as Cellphone;

//”as” 연산자는 이렇게 사용이 가능하다.