작성자 : 김영환

Q. 객체지향 프로그래밍에서 추상화, 은닉화, 캡슐화, 다형성, (+상속)이라는 특징 알기

객체지향: 프로그램의 **기본 단위를 객체**로 잡고, 그들간의 **상호작용**으로 문제를 해결하는 프로그래밍 방식으로 다음과 같은 특성을 가짐.

## 1. 추상화 (Abstraction)

- **객체지향의 핵심**은 이미 짜놓은 **코드를 재사용** 하는 데 있다. 이러한 코드 재사용성을 높이기 위해서, 객체의 **공통적인 속성과 행위를 추출**하는 것을 **추상화**라 한다. ex) 바바리안, 소서리스, 아마존 -> 기본캐릭터 기능(이동, 공격 가능)

## 2. 은닉화 (Hiding)

- 객체 **외부 접근을 허용하거나 차단**하는 개념을 **은닉화**라 한다. 이는 데이터의 변경을 관리할 수 있기 때문에 **유지보수에 유리**하다는 특성이 있다.

ex) 상점 정보 (외부 접근 허용) | 아이템 값, 캐릭터 레벨 값 (외부 접근 차단) -> 이차단을 푸는 프로그램이 핵(Hack)임

#### 3. 캡슐화 (Encapsulation)

- 동일한 쓰임새를 가진 속성과 행위를 **모아둠**으로써 **관리를 쉽게**하는 개념을 **캡슐화**라한다. 캡슐화를 통해, 프로그래머는 **소스 인지 단위**가 넓어지고, 복잡하고 거대한 프로그램도 쉽게 다룰 수 있게 되었다.

ex) 바바리안 클래스 내에는 레벨, str, dex, ··· 등과 스킬 메소드가 함께 모여있음

# 4. 다형성 (Polymorphism)

- 여러 클래스가 동일한 이름의 메소드를 사용할 수 있는 특성을 다형성이라 한다. 객체지향에서 가장 중요한 개념이며, 필요에 따라 전체 구조를 바꾸지 않으면서 프로그래머가 원하는 기능을 추가할 수 있다. 코드상에서는 메소드 오버라이딩과 오버로딩의 형태로 주로나타난다.

ex) 2.1 패치가 되었어도 블리자드 프로그래머들은 디아블로를 갈아엎지 않고, 다형성을 통해 '시너지'라는 기능을 무사히 스킬트리에 접목하였음

# 5. 상속 (Inheritance)

- 클래스의 속성과 메소드를 **하위 클래스에게 물려주거나**, 혹은 **상위 클래스로부터 받아 오는 것을 상속**이라 한다. 상위의 클래스를 바탕으로 하위 클래스에 새로운 기능을 추가하 거나, 다형성을 통해 필요에 따른 클래스를 손쉽게 새로 정의할 수 있다.

ex) 디아블로 확장팩에서 등장한 '드루이드'와 '어쎄신'은 상속 덕분에 기본캐릭터 기능(이 동, 공격 가능)을 물려받아 디아블로 세계에 존재할 수 있게 됨