**일일학습일지** 로고

**제20기 자바&파이썬 웹애플리케이션 전문가 양성과정**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **학습일시** | 20240129 | **장소** | kt ds University 211호 | **시 간** | 09:00~18:00 |
| **학습범위** | 자바\_final메소드추상메소드추상클래스인터페이스구현및특징익명클래스 | | | | |
| **작 성 자** | 김민성 | | | **강 사** | 장민창 |

|  |  |
| --- | --- |
| **학습안건** | 자바\_final메소드추상메소드추상클래스인터페이스구현및특징익명클래스 |

|  |  |
| --- | --- |
| **학습내용** | **내용** |
| final method - 클래스의 메소드 앞에 final을 붙인 메소드로 오버라이딩이 불가능하다.  추상클래스 - 공통된 부분은 모두 구현(추상)해놓고 다른 부분들만 구현하지 않은(사상)메소드가 있는 클래스  Abstract Method - 구현하지 않은 메소드  Abstract Class - 구현하지 않은 메소드가 있는 클래스  추상클래스(추상메소드 + 사상메소드) <- 어느 정도 구현이 되어있음  유연함의 정도는 추상클래스 < 인터페이스가 높다.  인터페이스의 의미  1. 인터페이스는 클래스와 클래스를 잇는 매개체  2. 개발자간 커뮤니케이션을 위한 표준 제공  인터페이스 만드는 방법(405P)  1. 인터페이스 클래스에 추상클래스만 선언  public interface SomeInterface{  public abstract String getSomthing(  );2. 구현클래스 생성 implements 키워드 사용  public cass SomeClass implemetns SomeInterFace{}  @override  public void doSomething(){}  3. 인터페이스는 new 키워드를 통해 인스턴스화시킬 수 있다.  SomeInterFace some = new SomeCalss();  인터페이스의 특징  1. 인터페이스만 상속 가능하다.  2.인터페이스 클래스에서 메소드 정의 시 abstract 키워드는 생략이 가능하다.  Java의 추상화 방법  1. 추상클래스 -> 상속 -> 구현(클래스)  2. 인터페이스 -> 구현(클래스)  익명클래스 - 추상 클래스나 인터페이스를 강제로 인스턴스화하는 방법  -> 클래스를 만드는 이유는 재사용을 위해서인데 단 한번의 사용을 위해서 작성한다면 비효율적이므로 익명클래스를 사용한다.  웹개발에서는 콜백에서 많이 사용한다.  익명클래스예시)  Rideable car1 = new Rideable() {  @Override  public void StartEngine(){  내용  }  };  car1.startEngine(); |