[08. 인터페이스]

1. 인터페이스에 대한 설명으로 틀린 것은 무엇입니까?

➊ 인터페이스로 객체(인스턴스)를 생성할 수 있다.ㅇㅇ

➋ 인터페이스는 다형성의 주된 기술로 사용된다.

➌ 인터페이스를 구현한 객체는 인터페이스로 동일하게 사용할 수 있다.

➍ 인터페이스를 사용함으로써 객체 교체가 쉬워진다.

2. 인터페이스의 구성 멤버에 대한 설명으로 틀린 것은 무엇입니까?

➊ 인터페이스는 인스턴스 필드가 없고 상수를 멤버로 가진다.

➋ 추상 메소드는 구현 클래스가 재정의해야 하는 멤버이다.

➌ 디폴트 메소드는 구현 클래스에서 재정의할 수 없다.ㅇㅇ

➍ 정적 멤버는 구현 객체가 없어도 사용할 수 있는 멤버이다.

3. 인터페이스 다형성에 대한 설명으로 틀린 것은 무엇입니까?

➊ 필드가 인터페이스 타입일 경우 다양한 구현 객체를 대입할 수 있다.

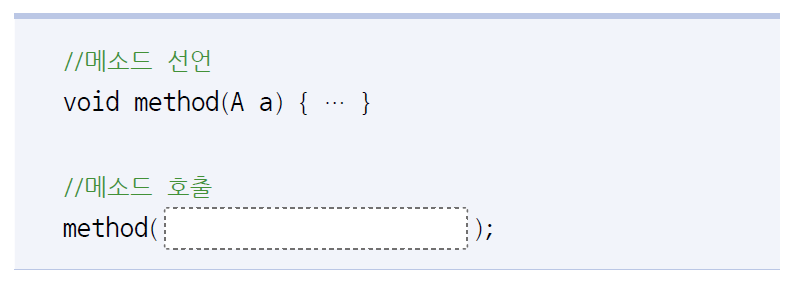
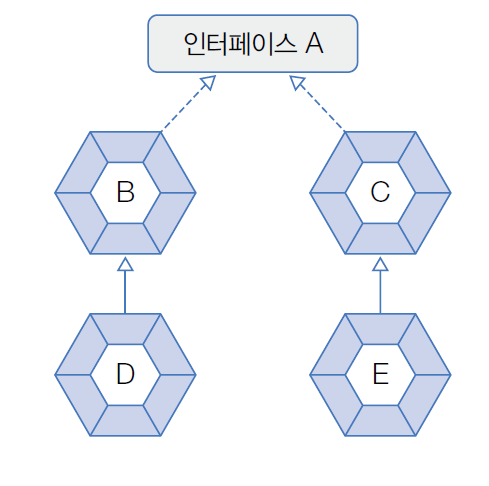
➋ 매개변수가 인터페이스 타입일 경우 다양한 구현 객체를 대입할 수 있다.

➌ 배열이 인터페이스 타입일 경우 다양한 구현 객체를 저장할 수 있다.ㅇㅇ

➍ 구현 객체를 인터페이스 타입으로 변환하려면 강제 타입 변환을 해야 한다

4. 인터페이스 A를 B와 C가 구현하고 B를 상속해서 D 클래스를, C를 상속해서 E 클래스를 만들었

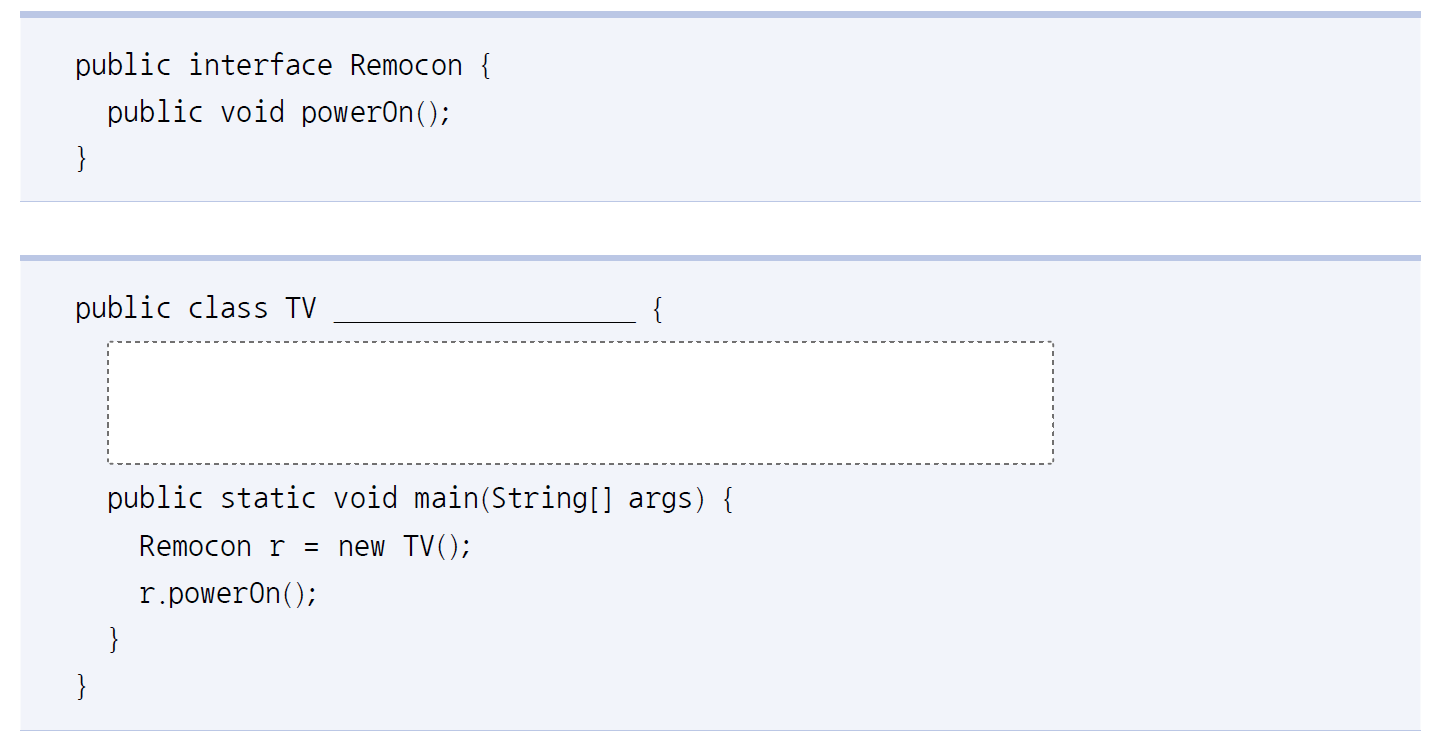
습니다. 다음 빈칸에 들어올 수 있는 것을 모두 선택하세요.



➊ new B( )ㅇㅇ ➋ new C( )ㅇㅇ ➌ new D( )ㅇㅇ ➍ new E( )ㅇㅇ

5. TV 클래스를 실행했을 때 “TV를 켰습니다.”라고 출력되도록 밑줄과 박스에 들어갈 코드를 작성

해보세요.



>>>> implements Remocon

>>>> Tv tv = new TV();

6. Soundable 인터페이스는 다음과 같은 sound ( ) 추상 메소드를 가지고 있습니다.

SoundableExample 클래스의 printSound( ) 메소드는 매개변수 타입으로 Soundable 인터페

이스를 가집니다. printSound( )를 호출할 때 Cat과 Dog 객체를 주고 실행하면 각각 “야옹”과 “멍

멍”이 출력되도록 Cat과 Dog 클래스를 작성해보세요.



public class Cat implements Soundable {

@override

public String sound() {

system.out.println(“야옹”)

}

}

public class Dog implements Soundable {

@override

public String sound() {

system.out.println(“멍”)

}

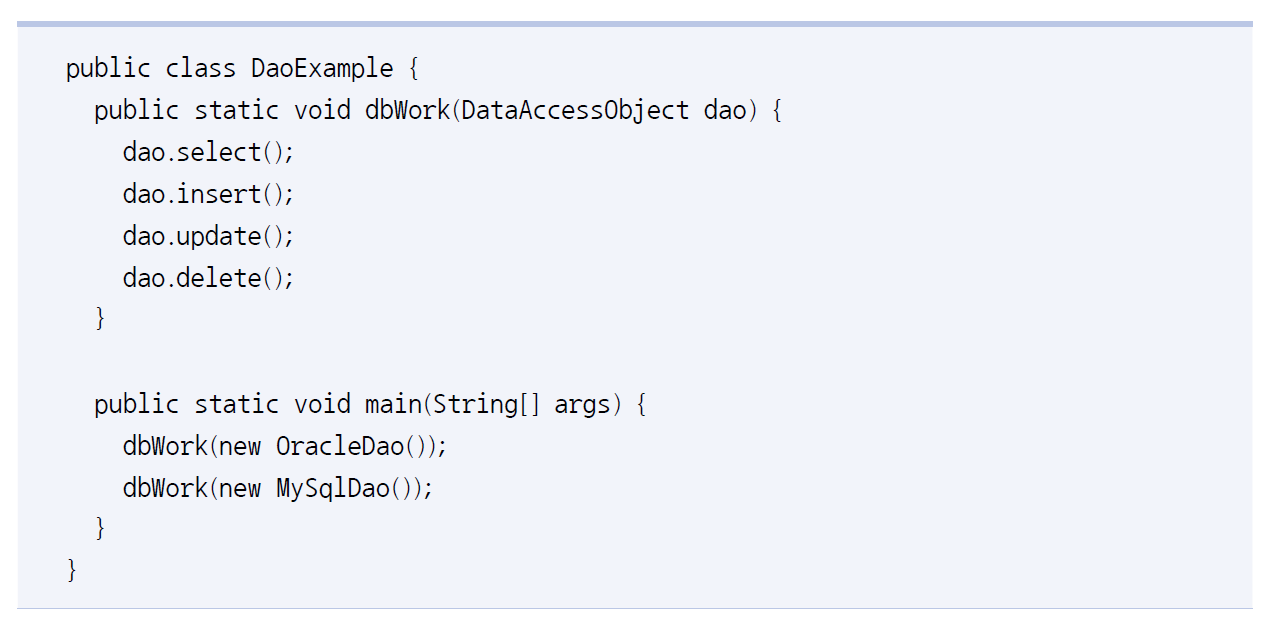
}

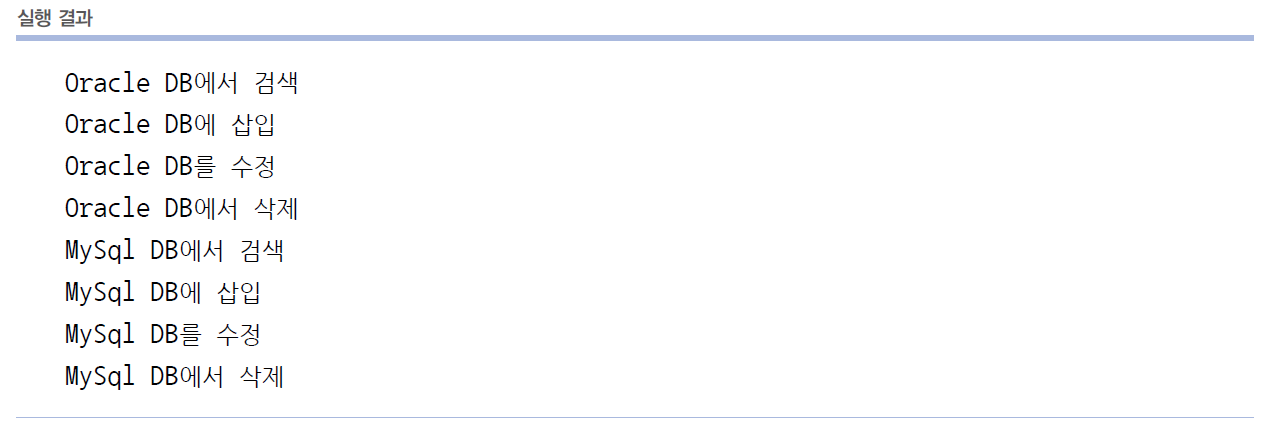
7. DaoExample 클래스의 main( ) 메소드에서 dbWork( ) 메소드를 호출할 때 OracleDao

와 MySqlDao 객체를 매개값으로 주고 호출했습니다. dbWork( ) 메소드는 두 객체를 모두 매

개값으로 받기 위해 DataAccessObject 타입의 매개변수를 가지고 있습니다. 실행 결과를 보고

DataAccessObject 인터페이스와 OracleDao, MySqlDao 구현 클래스를 각각 작성해보세요.





**public** **interface** DataAccessObject {

**void** select();

**void** insert();

**void** update();

**void** delete();

}

**public** **class** OracleDao **implements** DataAccessObject{

@Override

**public** **void** select() {

System.***out***.println("Oracle DB 에서 검색");

}

**public** **void** insert() {

System.***out***.println("Oracle DB 에 삽입");

}

**public** **void** update() {

System.***out***.println("Oracle DB 를 수정");

}

**public** **void** delete() {

System.***out***.println("Oracle DB 에서 삭제");

}

}

**public** **class** MySqlDao **implements** DataAccessObject{

@Override

**public** **void** select() {

System.***out***.println("MySql DB 에서 검색 ");

}

**public** **void** insert() {

System.***out***.println("MySql DB 에 삽입 ");

}

**public** **void** update() {

System.***out***.println("MySql DB 를 수정 ");

}

**public** **void** delete() {

System.***out***.println("MySql DB 에서 삭제 ");

}