

실습문제 - Interface 해답

* 다음 괄호 안을 채우세요.

1. 자바에서 인터페이스는 추상 메소드와 (**상수**)만을 포함하며, (**interface**) 키워드를 사용하여 선언한다.
2. 인터페이스를 구현할 때 (**implements**) 키워드를 사용하며, 인터페이스의 모든 메소드를 구현한다.
3. 추상 클래스와 인터페이스의 (**객체**)는 생성 할 수 없다.

4. 인터페이스에 대한 설명으로 틀린 것은 무엇입니까? 3

- ① 인터페이스는 객체 사용 설명서 역할을 한다.
- ② 구현 클래스가 인터페이스의 추상 메소드에 대한 실체 메소드를 가지고 있지 않으면 추상 클래스가 된다.
- ③ **인터페이스는 인스턴스 필드를 가질 수 있다.**
- ④ 구현 객체는 인터페이스 타입으로 자동 변환된다.

5. 인터페이스의 다형성과 거리가 먼 것은? 4

- ① 필드가 인터페이스 타입일 경우 다양한 구현 객체를 대입할 수 있다.
- ② 매개 변수가 인터페이스 타입일 경우 다양한 객체를 대입할 수 있다.
- ③ 배열이 인터페이스 타입일 경우 다양한 구현 객체를 저장할 수 있다.
- ④ **구현 객체를 인터페이스 타입으로 변환하려면 강제 타입 변환을 해야 한다.**

6. 다음 중 인터페이스의 특징이 아닌 것은? 3

- ① 인터페이스의 객체를 생성할 수 없다.
- ② 여러 인터페이스를 상속받는 다중 상속을 지원한다.
- ③ **인터페이스는 추상 메소드, 상수, 필드로 구성된다.**
- ④ 인터페이스는 implements 키워드를 이용하여 구현한다.

7. 인터페이스 **AdderInterface**의 코드는 다음과 같다. **AdderInterface**를 상속받는 클래스 **MyAddress**를 작성하여 **main()**를 실행할 때 아래 실행 결과와 같이 출력 되도록 하세요.

```
interface AdderInterface{
    public int add(int x, int y);    // x와 y의 합
    public int add(int x);          // 1에서 n까지의 정수 합 리턴, n은 0보다 큰 수로 지정
}
// MyAddress 작성

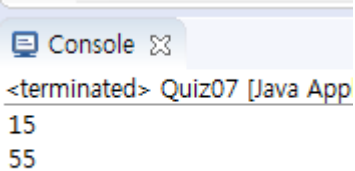
public class Quiz07 {
    public static void main(String[] args) {
        MyAddress my=new MyAddress();
    }
}
```

```

        System.out.println(my.add(5, 10));
        System.out.println(my.add(10));
    }
}

```

[결과화면]



```

<terminated> Quiz07 [Java App
15
55

```

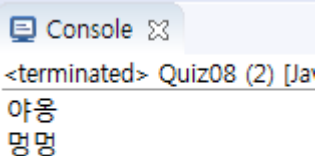
8. 다음 **Soundable** 인터페이스 입니다. **sound()** 추상 메소드는 객체의 소리를 리턴합니다. **Soundable**를 상속받는 클래스 **Cat, Dog** 작성하여 **main()**를 실행할 때 아래 실행 결과와 같이 출력 되도록 하세요.

```

interface Soundable{
    public String sound();
}
// Cat 작성
// Dog 작성
public class Quiz08 {
    public static void main(String[] args) {
        printSound(new Cat());
        printSound(new Dog());
    }
    private static void printSound(Soundable soundable){
        System.out.println(soundable.sound());
    }
}

```

[결과화면]



```

<terminated> Quiz08 (2) [Ja
야옹
멍멍

```

9. **dbWork()** 함수를 호출할 때 **OracleDao**와 **MySqlDao** 객체를 매개값으로 주고 호출했습니다. **dbWork()** 함수는 두 객체를 모두 매개값으로 받기 위해 **DataAccessObject** 타입의 매개 변수를 가지고 있습니다. 실행 결과를 보고 **DataAccessObject** 인터페이스와 **OracleDao, MySqlDao** 구현 클래스를 각각 작성해보세요.

```

public class Quiz09 {
    public static void main(String[] args) {
        dbWork(new OracleDao());
    }
}

```

```

        dbWork(new MySql());
    }
    public static void dbWork(DataAccessObject dao){
        dao.select();
        dao.insert();
        dao.update();
        dao.delete();
        System.out.println();
    }
}

```

[결과화면]

```

Console
<terminated> Quiz09 (2) [Java Ap
OracleDao 검색
OracleDao 삽입
OracleDao 수정
OracleDao 삭제

MySqlDao 검색
MySqlDao 삽입
MySqlDao 수정
MySqlDao 삭제

```

10. 다음은 **Action** 인터페이스입니다. **work()** 추상 메소드는 객체의 작업을 시작시킵니다. **main()** 메소드에서 **Action**의 익명 구현 객체를 만들어 다음과 같은 실행 결과가 나올 수 있도록 박스 안에 들어갈 코드를 작성해 보세요.

```

interface Action{
    void work();
}
public class Quiz10 {
    public static void main(String[] args) {
        Action action=

        action.work();
    }
}

```

[결과] 열심히 공부합니다.