

2025년 상반기 K-디지털 트레이닝

인터페이스

[KB] IT's Your Life

✓ RemoteControl 인터페이스를 완성하세요.

- 메서드는 리턴값이 없는 turnOn() 메서드

```
package ch08.sec02;  
  
public _____ RemoteControl {  
  
}
```

✓ RemoteControl 인터페이스를 구현하는 Television, Audiot 클래스를 완성하세요.

```
package ch08.sec02;  
  
public class Television {  
  
}
```

```
package ch08.sec02;  
  
public class Audio {  
  
}
```

✓ 앞에서 정의한 인터페이스와 클래스를 이용하여 다음과 같이 출력되도록 완성하세요.

```
package ch08.sec02;

public class RemoteControlExample {
    public static void main(String[] args) {
        _____ rc;

        _____;
        rc.turnOn();

        _____;
        rc.turnOn();
    }
}
```

TV를 켭니다.
Audio를 켭니다.

✓ 다음과 같이 인터페이스를 정의하세요.

```
package ch08.sec04;

public interface RemoteControl {
    int MAX_VOLUME = 10;
    int MIN_VOLUME = 0;

    void turnOn();
    void turnOff();
    void setVolume(int volume);
}
```

✓ 앞에서 정의한 RemoteControl 인터페이스를 구현하는 Television 클래스를 정의하세요.

○ setVolume() 메서드

- 매개변수 volume 값으로 필드 volume 값으로 설정
- 매개변수 volume 값이 최대 볼륨보다 큰 값이면 최대 볼륨으로 설정
- 최소 볼륨보다 작은 값이면 최소 볼륨으로 설정
- "현재 TV 볼륨: 값" 출력

```
package ch08.sec04;  
  
public class Television {  
    private int volume;  
  
}
```

✓ 앞에서 정의한 RemoteControl 인터페이스를 구현하는 Audio 클래스를 정의하세요.

○ setVolume() 메서드

- 매개변수 volume 값으로 필드 volume 값으로 설정
- 매개변수 volume 값이 최대 볼륨보다 큰 값이면 최대 볼륨으로 설정
- 최소 볼륨보다 작은 값이면 최소 볼륨으로 설정
- "현재 Audio 볼륨: 값" 출력

```
package ch08.sec04;

public class Audio {
    private int volume;
}
```

- ✔ 앞에서 정의한 인터페이스와 클래스를 이용하여 다음과 같이 출력되도록 RemoteControlExample 클래스를 완성하세요

```
package ch08.sec04;

public class RemoteControlExample {
    public static void main(String[] args) {

        _____ rc;

        rc = _____;

        rc = _____;

    }
}
```

TV를 켭니다.
현재 TV 볼륨: 5
TV를 끕니다.
Audio를 켭니다.
현재 Audio 볼륨: 5
Audio를 끕니다.

✓ 다음과 같이 인터페이스를 정의하세요.

```
package ch08.sec08;  
  
public interface RemoteControl {  
    void turnOn();  
    void turnOff();  
}
```

```
package ch08.sec08;  
  
public interface Searchable {  
    void search(String url);  
}
```


- ✔ 앞에서 정의한 RemoteControl, Searchable 인터페이스를 모두 구현한 SmartTelevision 클래스를 정의하세요.

```
package ch08.sec08;  
  
public class SmartTelevision {  
  
}
```

✓ 앞에서 만든 인터페이스와 클래스를 이용하여 다음 코드를 완성하세요.

```
package ch08.sec08;

public class MultiInterfaceImplExample {
    public static void main(String[] args) {

        _____ rc = new _____();
        rc.turnOn();
        rc.turnOff();

        _____ searchable = new _____();
        searchable.search("https://www.youtube.com");

    }
}
```

TV를 켭니다.

TV를 끕니다.

https://www.youtube.com을 검색합니다.

✓ 다음 인터페이스를 정의하세요.

```
package ch08.sec09;  
  
public interface InterfaceA {  
    void methodA();  
}
```

```
package ch08.sec09;  
  
public interface InterfaceB {  
    void methodB();  
}
```

✓ InterfaceA, InterfaceB를 상속하는 InterfaceC를 정의하세요.

```
package ch08.sec09;  
  
public interface InterfaceC _____ {  
    void methodC();  
}
```

✓ InterfaceC를 구현하는 InterfaceCImpl 클래스를 정의하세요.

- 어떤 메서드가 호출되는지 출력하는 코드로 각 메서드를 정의
 - InterfaceCImpl-methodA() 실행
 - InterfaceCImpl-methodB() 실행
 - InterfaceCImpl-methodC() 실행

```
package ch08.sec09;  
  
public class InterfaceCImpl {  
  
}
```

✓ 앞에서 정의한 인터페이스와 클래스를 이용하여 다음 코드를 완성하세요.

```
package ch08.sec09;

public class ExtendsExample {
    public static void main(String[] args) {
        InterfaceCImpl impl = new InterfaceCImpl();

        _____ ia = impl;
        ia.methodA();
        System.out.println();

        _____ ib = impl;
        ib.methodB();
        System.out.println();

        _____ ic = impl;
        ic.methodA();
        ic.methodB();
        ic.methodC();
    }
}
```

InterfaceCImpl-methodA() 실행

InterfaceCImpl-methodB() 실행

InterfaceCImpl-methodA() 실행

InterfaceCImpl-methodB() 실행

InterfaceCImpl-methodC() 실행

✓ 다음 인터페이스와 클래스를 정의하세요.

```
package ch08.sec10.exam02;
```

```
public interface Vehicle {  
    void run();  
}
```

```
package ch08.sec10.exam02;
```

```
public class Bus implements Vehicle {  
    @Override  
    public void run() {  
        System.out.println("버스가 달립니다.");  
    }  
  
    public void checkFare() {  
        System.out.println("승차요금을 체크합니다.");  
    }  
}
```

- ✓ 앞의 인터페이스와 클래스를 이용하여 다음 코드에서 잘못된 부분을 찾아 수정하고, 코드를 완성하세요.

```
package ch08.sec10.exam02;

public class CastingExample {
    public static void main(String[] args) {
        Vehicle vehicle = new Bus();

        vehicle.run();
        vehicle.checkFare();

        Bus bus = _____ vehicle;
        bus.run();
        bus.checkFare();
    }
}
```

버스가 달립니다.
버스가 달립니다.
승차요금을 체크합니다.