

실습문제 - 상속 해답

* 다음 밑줄에 알맞은 답을 넣으세요.

1. 자바에서 상속은 부모 클래스의 필드와 메소드를 자식 클래스에게 물려주는 것이다. 부모 클래스를 슈퍼 클래스, 자식 클래스를 서브 클래스 라고 한다.
2. 슈퍼 클래스의 접근제한자 protected는 패키지 소속과 상관없이 서브 클래스에서 접근이 가능하며 동일한 패키지 내의 클래스에서도 접근이 가능하다.
3. 서브 클래스 객체는 슈퍼 클래스 타입으로 자동 타입 변환이 가능하며 이를 업캐스팅 이라고 하며, 다시 원래의 타입으로 강제 타입 변환하는 것을 다운캐스팅이라고 한다.
4. 슈퍼 클래스에 선언된 메소드를 서브 클래스에서 재정의하는 것을 메소드 오버라이딩 이라고 한다.
5. 추상 메소드는 실행 코드 없이 원형만 선언된 미완성 메소드이며, 반드시 메소드 이름 앞에 abstract 를 붙여 선언하여야 한다.

6. 자바의 상속에 대한 설명 중 틀린 것은 무엇입니까? 1

- ① 자바는 다중상속을 허용한다.
- ② 부모의 메소드를 자식 클래스에 재정의(오버라이딩) 할수 있다.
- ③ 부모의 private 접근 제한을 갖는 필드와 메소드는 상속의 대상이 아니다.
- ④ final 클래스는 상속할 수 없고, final 메소드는 오버라이딩 할 수 없다.

7. 클래스 타입 변환에 대한 설명 중 틀린 것은 무엇입니까? 2

- ① 자식 객체는 부모 타입으로 자동 타입 변환된다.
- ② 부모 객체는 항상 자식 타입으로 강제 타입 변환된다.
- ③ 자동 타입 변환을 이용해서 필드와 매개 변수의 다형성을 구현한다.

8. final 키워드에 대한 설명으로 틀린 것은? 1

- ① final 클래스는 부모 클래스로 사용할 수 있다.
- ② final 필드는 값이 저장된 후에 변경할 수 없다.
- ③ final 메소드는 재정의(오버라이딩) 할 수 없다.
- ④ static final 필드는 상수를 말한다.

9. 오버라이딩(Overriding)에 대한 설명으로 틀린것은? 4

- ① 부모 메소드의 시그니처(리턴타입, 메소드명, 매개변수)와 동일해야 한다.
- ② 부모 메소드보다 좁은 접근 제한자를 붙일수 없다. (public(부모) ---> private(자식))

- ③ @Override 어노테이션을 사용하면 재정의가 확실한지 컴파일러가 검증한다.
- ④ **protected 접근 제한을 갖는 메소드는 다른 패키지의 자식 클래스에서 재정의할 수 없다.**

10. 메소드 오버라이딩에 대한 설명으로 틀린것은? 4

- ① 상속관계에서 발생한다.
- ② 슈퍼클래스의 메소드를 서브클래스에서 재작성하는 것이다.
- ③ 서브클래스에서 슈퍼클래스의 메소드 이름, 매개변수의 개수와 타입, 리턴타입을 동일하게 선언해야 성공한다.
- ④ **메소드의 이름이 동일할 필요는 없다.**

11. 생성자에 관한 문제이다. 다음 빈칸에 적절한 한줄의 코드를 삽입하라.

```
class LCD{
    private int size;
    public LCD(int size){
        this.size=size;
    }
}
class ColorLCD extends LCD{
    int colorSize;
    public ColorLCD(int colorSize, int size){
        super(size);
        this.colorSize=colorSize;
    }
}
```

12. 업캐스팅에 관한 문제이다. 다음 코드에는 클래스 A, B와 객체 생성 코드가 있다.

잘못 선언된 것은? 3

```
class A{
    protected int x;
}
class B extends A{
    private int y;
}
.....
A a=new A();
B b=new B();
```

- ① a=new A();
- ② a=b;
- ③ **b=a;**
- ④ b=new B();

13. 다음을 실행 하였을 때, 출력 결과물을 작성하세요

```
class A{
    public A(){
        System.out.println("안녕하세요.");
    }
}
class B extends A{
    public B(){
        super();
    }
}
public class ABC {
    public static void main(String[] args){
        B b=new B();
    }
}
```

[결과] 안녕하세요.

14. 다음의 클래스 정의 중에서 추상 클래스를 옳게 정의한 것은 어느 것인가? 1

① **abstract class A{**

abstract void sub();

}

② **abstract class A{**

abstract void sub(){ 함수내용 생략 }

}

③ **class A{**

abstract void sub();

}

④ **class A{**

abstract void sub(){ 함수내용 생략 }

}

15. 오버라이딩(overriding) 조건 3가지를 작성하세요

조상클래스의 메소드와 이름이 같아야한다. 매개변수가 같아야한다. 리턴이 같아야한다

16. 다음 상속에 대해서 작성된 다음 프로그램에서 잘못된 곳을 찾으세요

```
1    class Super{
2        private int a, b;
3        public void disp(){
4            System.out.println(a + "," + b);
5        }
6    }
7    class Sub extends Super{
8        public void disp()
```

```
9          System.out.println(a + "," + b);
10      }
11  }
```

17. 다음 프로그램의 결과값은? **서브 클래스 입니다**

```
class Super{
    public void printMsg(){
        System.out.println("상위 클래스입니다.");
    }
}

class Sub extends Super{
    public void printMsg(){
        System.out.println("서브 클래스 입니다.");
    }
}

public class ABC {
    public static void main(String[] args){
        Super s=new Sub();
        s.printMsg();
    }
}
```

18. 다음 중 import 문을 명시적으로 사용하지 않아도 사용할 수 있는 패키지를 모두 고르세요.

- ① 현재 패키지
 - ② java.lang 패키지
 - ③ java.util 패키지
 - ④ 현재패키지의 상위 패키지

19. 다음 중 접근 제한자에 대한 설명으로 옳지 않은 것은? 2

- ① public은 접근 제한이 전혀 없는 접근 제한자 이다.
- ② 지역변수에도 접근 제한자를 사용할 수 있다.
- ③ protected가 붙으면, 같은 패키지 내에서도 접근이 가능하다.
- ④ protected가 붙으면, 다른 패키지에서 상속하면 자손클래스에선 접근이 가능하다

20. 다음과 같은 클래스를 선언해서 상속 하였다. 각각 해당 변수들의 값은?

```
class AA{    protected int x=10;    }
class BB extends AA{
    public void disp(){
        System.out.println(x + "," + this.x + "Wt" + super.x);
    }
}
```

[결과값] x:10 this.x:10 super.x:10

21. 다음 추상클래스 Study로 부터 상속받아 Study_Sub 클래스를 정의 하세요

```
abstract class Study{
    public abstract void printMsg();
}
// 프로그램 작성
class Study_Sub extends Study{
    public void printMsg(){
        System.out.println("Java는 참 재미있습니다.");
    }
}
public class ABC {
    public static void main(String[] args){
        Study s=new Study_Sub();
        s.printMsg();
    }
}
```

[결과 화면] Java는 참 재미있습니다.

22. 다음을 프로그램을 보시고 Son 생성자를 작성하세요

```
class Parents{
    protected int x, y;
    public Parents(int x, int y){
        this.x=x;
        this.y=y;
    }
}
class Son extends Parents{
    private int z;
    // 생성자 작성
    public Son(int x, int y, int z){
        super(x, y);
        this.z=z;
    }
    public void disp(){
        System.out.println(x + "," + y + "," + z);
    }
}
```

23. 다음 클래스를 보시고 오버라이딩 하세요

```
class Employee{
    protected String name, dept, grade;
```

```

        protected int number, age;
        public Employee(String name, String dept, String grade, int number, int age){
            this.name=name;
            this.dept=dept;
            this.grade=grade;
            this.number=number;
            this.age=age;
        }
        public void disp(){
            System.out.println("한사람의 이름은" + name + "이고 나이는" + age);
        }
    }
    class Manager extends Employee{
        // 프로그램 작성
        public Manager(String name, String dept, String grade, int number, int age){
            super(name, dept, grade, number, age);
        }
        public void disp(){
            System.out.println("한사람의 이름은" + name + "이고 부서는" + dept);
        }
    }
    public class Quiz{
        public static void main(String[] args) {
            Manager a=new Manager("홍길동", "개발부서", "대리", 1234, 27);
            a.disp();
        }
    }
}

```

[결과화면] 한사람의 이름은홍길동이고 부서는개발부서

24. 다음의 그림을 보시고 추상 클래스를 선언 하세요 (send는 추상 메서드)

Message
#title: String #name: String
+send(title: String, name: String)

```

abstract class Message{
    protected String title;
    protected String name;
    public abstract void send(String title, String name)
}

```

25. Parent 클래스를 상속해서 Child 클래스를 다음과 같이 작성했는데, Child 클래스의 생성자에서 컴파일 에러가 발생 했습니다. 그 이유를 설명해보세요.

```

class Parent{
    public String name;
    public Parent(String name){
        this.name=name;
    }
}
class Child extends Parent{
    private int studentNo;
    public Child(String name, int studentNo){
        super(name);
        this.name=name;
        this.studentNo=studentNo;
    }
}

```

26. Parent 클래스를 상속받아 Child 클래스를 다음과 같이 작성했습니다. ChildExample 클래스를 실행했을 때 호출되는 각 클래스의 생성자의 순서를 생각하면서 출력 결과를 작성해 보세요.

```

class Parent{
    public String nation;
    public Parent(){
        this("대한민국");
        System.out.println("Parent() call");  -----> ②
    }
    public Parent(String nation){
        this.nation=nation;
        System.out.println("Parent(String nation) call");  -----> ①
    }
}
class Child extends Parent{
    private String name;
    public Child(){
        // super();  error
        this("홍길동");  -----> ④
        System.out.println("child() call");
    }
    public Child(String name){
        this.name=name;
        System.out.println("child(String name) call");  -----> ③
    }
}
public class ChildExample {
    public static void main(String[] args){
        Child child=new Child();  }}

```

[결과값]

Parent(String nation) call

Parent() call

child(String name) call

child() call

27. Tire 클래스를 상속받아 SnowTire 클래스를 다음과 같이 작성 했습니다. SnowTireExample 클래스를 실행했을 때 출력 결과는 무엇일까요?

```
class Tire{
    public void run(){
        System.out.println("일반 타이어가 굴러갑니다.");
    }
}
class SnowTire extends Tire{
    public void run(){
        System.out.println("스노우 타이어가 굴러갑니다.");
    }
}
public class SnowTireExample {
    public static void main(String[] args){
        SnowTire snowTire=new SnowTire();
        Tire tire=snowTire;
        snowTire.run();
        tire.run();
    }
}
```

[결과값]

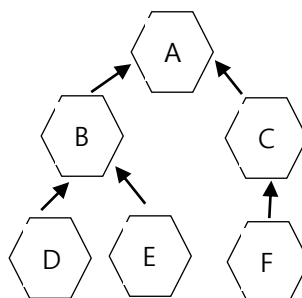
스노우 타이어가 굴러갑니다.

스노우 타이어가 굴러갑니다.

28. A, B, C, D, E, F 클래스가 다음과 같이 상속 관계에 있을 때 다음 빈칸에 들어올 수 없는 코드는? 2

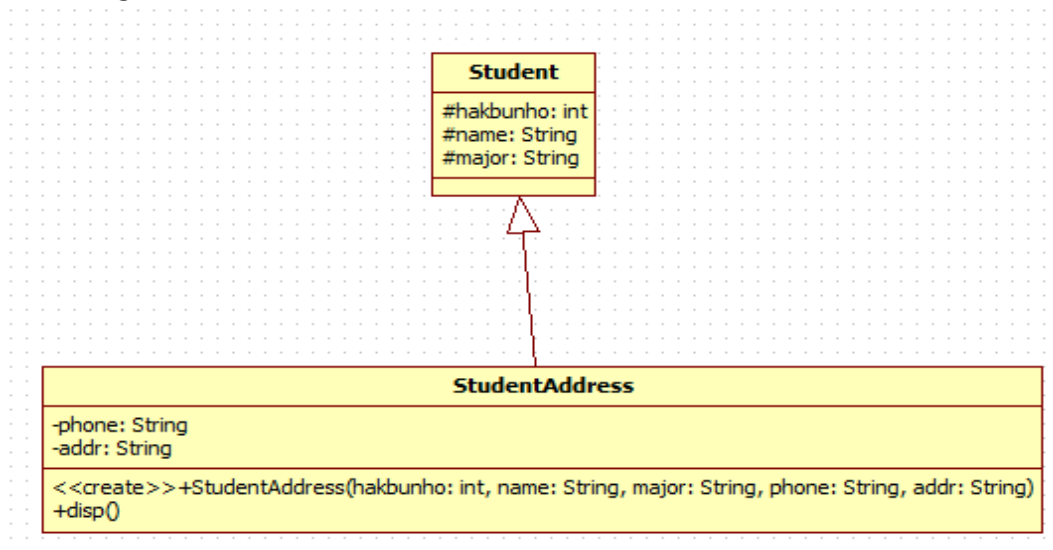
B b=();

- ① new B()
- ② (B) new A() // 강제타입변환
- ③ new D() // 자동타입변환
- ④ new E() // 자동타입변환



29. Class Diagram, 메인화면, 결과화면을 참조하여 프로그램을 작성하세요.

[Class Diagram]



[메인화면]

```

public class Quiz29 {
    public static void main(String[] args){
        StudentAddress s=new StudentAddress(123456, "홍길동", "소프트웨어", "010-123-1234", "서울시강남구");
        s.disp();
    }
}
    
```

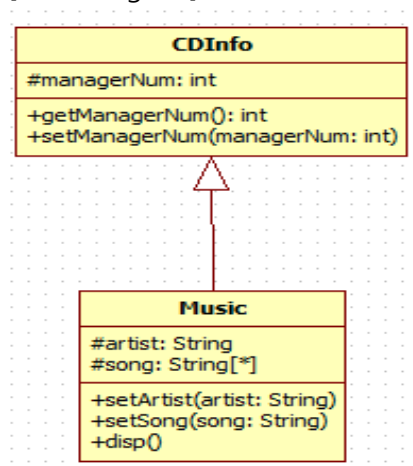
[결과화면]

```

Console
<terminated> Quiz01 (2) [Java App]
학번:123456
이름:홍길동
학과:소프트웨어
전화번호:010-123-1234
주소:서울시강남구
    
```

30. Class Diagram, 메인화면, 결과화면을 참조하여 프로그램을 작성하세요.

[Class Diagram]



[메인화면]

```

public class Quiz30 {
    public static void main(String[] args){
        Music m=new Music();
        m.setManagerNum(123456);
        m.setArtist("플라이 투 더 스카이");
        String[] song={"너를너를너를", "전화하지 말아요"};
        m.setSong(song);

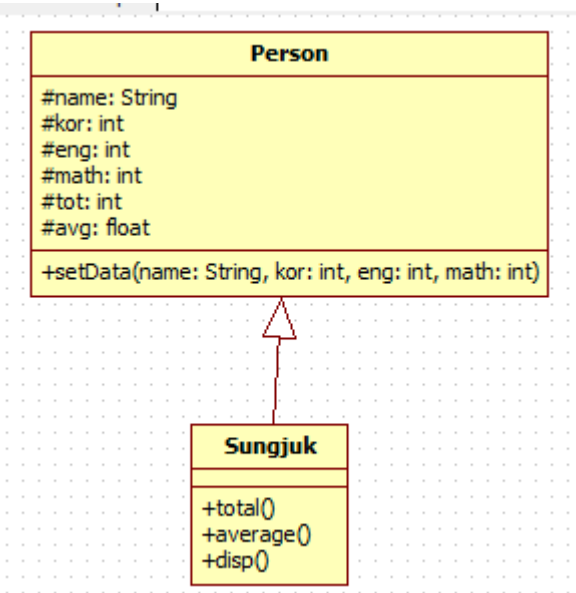
        m.disp();
    }
}
    
```

[결과화면]

```
Console
<terminated> Quiz02 [Java Ap
관리자 번호:123456
가수:플라이 투 더 스카이
힌트송[1]:너를너를너를
힌트송[2]:전화하지 말아요
```

31. Class Diagram, 메인화면, 결과화면을 참조하여 프로그램을 작성하세요.

[Class Diagram]



[메인화면]

```
public class Quiz31 {
    public static void main(String[] args) {
        Sungjuk s=new Sungjuk();
        s.setData("홍길동", 100, 70, 80);
        s.total();
        s.average();
        s.disp();
    }
}
```

[결과화면]

```
Console
<terminated> Quiz03 [Java Application] C:\jdk1.8\bin
홍길동 100 70 80 250 83.333336
```

32. 다음 클래스를 보시고 오버라이딩 하세요

```
class Employee{
    protected String name, dept, grade;
    protected int number, age;
    public Employee(String name, String dept, String grade, int number, int age){
        this.name=name;
        this.dept=dept;
        this.grade=grade;
        this.number=number;
        this.age=age;
    }
}
```

```

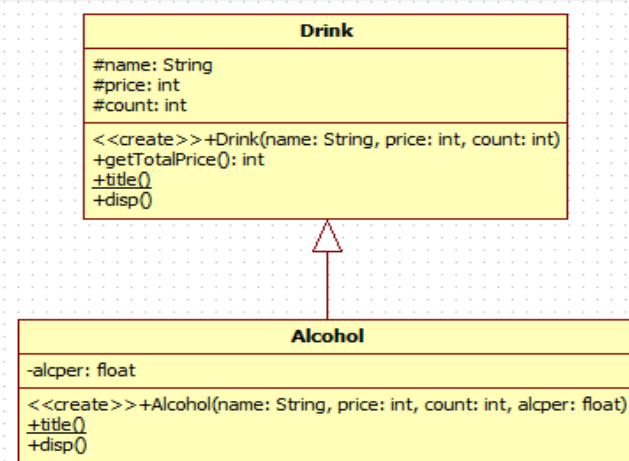
        public void disp(){
            System.out.println("한사람의 이름은" + name + "이고 나이는" + age);
        }
    }
    class Manager extends Employee{    // 여기 부분을 만드세요.    }
    public class Quiz31 {
        public static void main(String[] args) {
            Manager a=new Manager("홍길동", "개발부서", "대리", 1234, 27);
            a.disp();
        }
    }

```

[결과화면] 한사람의 이름은홍길동이고 부서는개발부서

33. Class Diagram, 메인화면, 결과화면을 참조하여 프로그램을 작성하세요.

[Class Diagram]



[결과화면]

```

public class Quiz34 {
    public static void main(String[] args) {
        Drink coffee=new Drink("커피", 200, 3);
        Drink tea=new Drink("녹차", 150, 2);

        Drink.title();
        coffee.disp();
        tea.disp();

        Alcohol wine=new Alcohol("와인", 300, 2, 15.0f);
        Alcohol.title();
        wine.disp();
    }
}

```

[출력화면]

```

Console
<terminated> Quiz32 [Java Application] C:\wjd
상품  단위  수량  금액
커피   200    3    600
녹차   150    2    300
상품명(도수[%])  단위  수량  금액
와인(15.0)      300    2    600

```

34. 두 개의 클래스(Circle, Rectangular)는 Figure 클래스를 상속하여 출력화면 같이 나올 수 있게 프로그램을 완성하세요.

```

abstract class Figure{
    public abstract void area();
    public abstract void around();
}

```

[메인화면]

```
public class Quiz35 {  
    public static void main(String[] args){  
        Figure f=new Rectangular(2.3, 3.7);  
        f.area();  
        f.around();  
  
        Figure i=new Circle(3.6);  
        i.area();  
        i.around();  
    }  
}
```

[출력화면]

```
Console  
<terminated> Quiz33 [Java Applicator  
사각형의 면적:8.51  
사각형의 둘레:12.0  
원의 면적:40.715032320000006  
원의 둘레:22.619462400000003
```