

상속

1.

5. Parent 클래스를 상속해서 Child 클래스를 다음과 같이 작성했는데, Child 클래스의 생성자에서 컴파일 에러가 발생합니다. 그 이유를 설명해보세요.

【 Parent.java 】

```
01 public class Parent {  
02     public String name;  
03  
04     public Parent(String name) {  
05         this.name = name;  
06     }  
07 }
```

【 Child.java 】

```
01 public class Child extends Parent {  
02     private int studentNo;  
03  
04     public Child(String name, int studentNo) {  
05         this.name = name;  
06         this.studentNo = studentNo;  
07     }  
08 }
```

2.

2 아래의 코드를 실행하면 그 결과는 무엇인가? (힌트 : <표 5-2>의 경우 2)

```
class A {  
    public A() {  
        System.out.println("생성자A");  
    }  
    public A(int x) {  
        System.out.println("매개변수생성자A" + x);  
    }  
}  
class B extends A {  
    public B() {  
        super(30);  
        System.out.println("생성자B");  
    }  
    public B(int x) {  
        System.out.println("매개변수생성자B" + x);  
    }  
}  
  
public class ConstructorEx5 {  
    public static void main(String[] args) {  
        B b;  
        b = new b();  
    }  
}
```

3 다음은 서브 클래스 B의 생성자 코드이다 잘못된 부분은?

```
public B() {  
    System.out.println("생성자B");  
    super(10);  
}
```

3.

다음은 상속에 있어서 생성자 간의 관계를 묻기 위한 코드이다. 이 코드를 수행했을 때 출력되는 내용은 무엇인가?

```
class A {  
    public A() {  
        System.out.println("생성자A");  
    }  
    public A(int x) {  
        System.out.println("생성자A: " + x);  
    }  
}  
class B extends A {  
    public B(int x) {  
  
        System.out.println("생성자B: " + x);  
    }  
}  
public class Example {  
    public static void main(String[] args) {  
        A a = new A(10);  
        B b = new B(11);  
    }  
}
```

4.

7. Tire 클래스를 상속받아 SnowTire 클래스를 다음과 같이 작성했습니다. SnowTireExample 클래스를 실행했을 때 출력 결과는 무엇일까요?

【 Tire.java 】

```
01 public class Tire {
02     public void run() {
03         System.out.println("일반 타이어가 굴러갑니다.");
04     }
05 }
```

【 SnowTire.java 】

```
01 public class SnowTire extends Tire {
02     @Override
03     public void run() {
04         System.out.println("스노우 타이어가 굴러갑니다.");
05     }
06 }
```

【 SnowTireExample.java 】

```
01 public class SnowTireExample {
02     public static void main(String[] args) {
03         SnowTire snowTire = new SnowTire();
04         Tire tire = snowTire;
05
06         snowTire.run();
07         tire.run();
08     }
09 }
```

4.

다음 소스에는 오류가 포함되어 있다. 오류를 찾아내어 이유를 설명하고, 오류를 수정하라.

```
class A {
    public int i;
    private int j;
    public void setJ(int n) {
        j = n;
    }
    public int getJ() {
        return j;
    }
}

class B extends A {
    private int n;
    public void setN(int i) {
        n = i;
    }
```

```
    public int getN() {
        return n;
    }
    public int getSum() {
        return n + i + j;
    }
}
```

5.

다음 코드에는 오류가 포함되어 있다. 오류를 찾아내어 이유를 설명하고, 오류를 수정하라.

```
class A {
    private int a;
    public A(int i) {
        a = i;
    }
}

class B extends A {
    private int b;
    public B() {
        b = 0;
    }
}
```

6.

다음 빈칸에 적절한 한 줄의 코드를 삽입하라.

```
class LCD {  
    int size;  
    public LCD(int n) { size = n;}  
}  
class ColorLCD extends LCD {  
    int colorSize;  
    public B(int colorSize, int size) {  
        _____  
    }  
}
```

7. 다음 클래스들을 상속을 이용해 간결하게 구성해보세요.

```
class A {
    private int n;
    public int getN() {
        return n;
    }
    public void setN(int i) {
        n = i;
    }
}

class B {
    public String s;
    private int n;
    public int m;
    private char c;
    public void setN(int i) {
        n = i;
    }
    public void setC(char ch) {
        c = ch;
    }
    public char getC() {
        return c;
    }
    public int getN() {
        return n;
    }
}

class C {
    public String s;
    private char c;
    private int n;
    public int m;
    public double d;
    public void setN(int i) {
        n = i;
    }
}

    public void setC(char ch) {
        c = ch;
    }
    public char getC() {
        return c;
    }
    public int getN() {
        return n;
    }
}
```

8.

과제 1 다음 클래스를 기반으로 main() 메서드가 실행될 수 있도록 프로그램을 완성해 보자.

■ 기반 클래스

```
class A{
    private int x;
    private int y;
    .....
}
class B extends A{
    private int x;
    private int y;
}
public class Work{
    public static void main(String[] ar){
        B bp = new B();
        B bp1 = new B(10);
        B bp2 = new B(10, 20);
        B bp3 = new B(10, 20, 30);
        B bp4 = new B(10, 20, 30, 40);
        bp.disp();
        bp1.disp();
        bp2.disp();

        bp3.disp();
        bp4.disp();
    }
}
```

■ 실행 결과

```
x = 1, y = 1, x = 1, y = 1
x = 10, y = 1, x = 1, y = 1
x = 10, y = 20, x = 1, y = 1
x = 10, y = 20, x = 30, y = 1
x = 10, y = 20, x = 30, y = 40
```


9.

6. Parent 클래스를 상속받아 Child 클래스를 다음과 같이 작성했습니다. ChildExample 클래스를 실행했을 때 호출되는 각 클래스의 생성자의 순서를 생각하면서 출력 결과를 작성해보세요.

【Parent.java】

```
01 public class Parent {
02     public String nation;
03
04     public Parent() {
05         this("대한민국");
06         System.out.println("Parent() call");
07     }
```

확인문제

```
08
09     public Parent(String nation) {
10         this.nation = nation;
11         System.out.println("Parent(String nation) call");
12     }
13 }
```

【Child.java】

```
01 public class Child extends Parent {
02     private String name;
03
04     public Child() {
05         this("홍길동");
06         System.out.println("Child() call");
07     }
08
09     public Child(String name) {
10         this.name = name;
11         System.out.println("Child(String name) call");
12     }
13 }
```

【ChildExample.java】

```
01 public class ChildExample {
02     public static void main(String[] args) {
03         Child child = new Child();
04     }
05 }
```

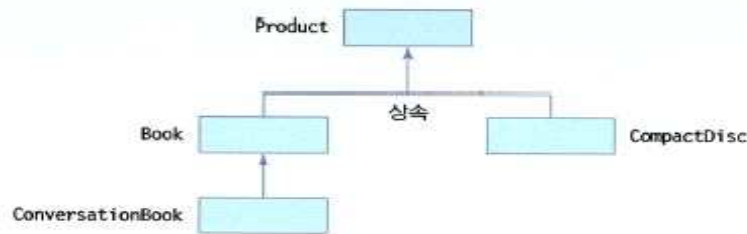
10.

다음 그림과 같은 상속 구조를 갖는 클래스를 설계하라. 모든 프린터는 모델명, 제조사, 인터페이스 종류(USB 또는 병렬 인터페이스), 인쇄 매수, 인쇄 종이 잔량을 나타내는 정보를 가지며 `print()` 라는 메소드를 갖는다. 잉크젯 프린터는 잉크 잔량이라는 추가 정보를 가지며 레이저 프린터는 토너 잔량이라는 추가 정보를 가진다. `print()` 메소드는 각 프린터 타입에 맞게 구현한다. 편의상 `print()` 메소드를 한 번 호출할 때마다 인쇄용지 1매를 사용하는 것으로 한다. 이러한 클래스를 설계하고 적절한 필드, 생성자, 메소드를 작성하고, 실행 프로그램을 작성하여 동작을 확인하라.



11.

다음과 같은 클래스 구조와 조건을 가진 자바 프로그램을 작성해보라.



Product 클래스는 각 상품의 고유한 식별자, 상품 설명, 생산자, 가격 정보를 포함하고 있다. Book 클래스는 ISBN 번호, 저자, 책 제목 정보를 포함한다. CompactDisc 클래스는 앨범 제목, 가수 이름 정보를 포함한다. ConversationBook은 회화책에서 다루는 언어명 정보를 포함한다. 객체 지향 개념에 부합하도록 적절한 접근 지정자, 필드, 메소드, 생성자 등을 작성하라. ProductInfo 클래스를 만들고 이곳에 main()을 둔다. main()에서는 최대 10개의 상품을 추가할 수 있으며 모든 상품의 정보를 조회할 수 있다. 모든 제품에 대한 정보를 출력할 때 Product 타입의 레퍼런스를 이용하라.

```

C:\Wtmp>java ProductInfo
상품 추가<1>, 모든 상품 조회<2>, 끝내기<3>>>1
상품 종류>>책<1>, 음악CD<2>, 회화책<3>>>2
상품 설명>>김가수 3집
생산자>>우리기획
가격>>12000
앨범 제목>>김가수 3집
가수>>김가수

상품 추가<1>, 모든 상품 조회<2>, 끝내기<3>>>1
상품 종류>>책<1>, 음악CD<2>, 회화책<3>>>3
상품 설명>>영어 회화책
생산자>>우리 영어사
가격>>20000
책 제목>>30분에 끝내는 영어
저자>>김영어
언어>>영어
ISBN>>000001111

상품 추가<1>, 모든 상품 조회<2>, 끝내기<3>>>2
상품ID>>0
상품 설명>>김가수 3집
생산자>>우리기획
가격>>12000
앨범 제목>>김가수 3집
가수>>김가수
상품ID>>1
상품 설명>>영어 회화책
생산자>>우리 영어사
가격>>20000
ISBN>>1111
책 제목>>30분에 끝내는 영어
저자>>김영어
언어>>영어

상품 추가<1>, 모든 상품 조회<2>, 끝내기<3>>>3
C:\Wtmp>
  
```