

실습과제 1

[5~8] 다음은 2차원 상의 한 점을 표현하는 Point 클래스이다.

```
class Point {  
    private int x, y;  
    public Point(int x, int y) { this.x = x; this.y = y; }  
    public int getX() { return x; }  
    public int getY() { return y; }  
    protected void move(int x, int y) { this.x = x; this.y = y; }  
}
```

🔗 서브 클래스 생성자 및 메소드 작성, super 활용

8. Point를 상속받아 양수의 공간에서만 점을 나타내는 PositivePoint 클래스를 작성하라. 다음 main() 메소드를 포함하고 실행 결과와 같이 출력되게 하라. 난이도 5

```
public static void main(String[] args) {  
    PositivePoint p = new PositivePoint();  
    p.move(10, 10);  
    System.out.println(p.toString() + "입니다.");  
  
    p.move(-5, 5); // 객체 p는 음수 공간으로 이동되지 않음  
    System.out.println(p.toString() + "입니다.");  
  
    PositivePoint p2 = new PositivePoint(-10, -10);  
    System.out.println(p2.toString() + "입니다.");  
}
```

(10,10)의 점입니다.
(10,10)의 점입니다.
(0,0)의 점입니다.

힌트

Point 클래스의 move()를 PositivePoint 클래스에서 오버라이딩하여 재작성하고 적절히 super.move()를 호출해야 한다. PositivePoint의 2 개의 생성자에서도 적절히 super() 생성자와 super.move()를 호출해야 한다.

실습과제2

9. 다음 Stack 인터페이스를 상속받아 실수를 저장하는 StringStack 클래스를 구현하라.

```
interface Stack {  
    int length(); // 현재 스택에 저장된 개수 리턴  
    int capacity(); // 스택의 전체 저장 가능한 개수 리턴  
    String pop(); // 스택의 톱(top)에 실수 저장  
    boolean push(String val); // 스택의 톱(top)에 저장된 실수 리턴  
}
```

☞ 인터페이스에 대한 이해 및 클래스 구현 활용

그리고 다음 실행 사례와 같이 작동하도록 StackApp 클래스에 main() 메소드를 작성하라. **난이도 6**

```
총 스택 저장 공간의 크기 입력 >> 3  
문자열 입력 >> hello  
문자열 입력 >> sunny  
문자열 입력 >> smile  
문자열 입력 >> happy  
스택이 꽉 차서 푸시 불가!  
문자열 입력 >> 그만 "그만"을 입력하면 프로그램 종료  
스택에 저장된 모든 문자열 팝 : smile sunny hello
```


실습과제 3

10. 다음은 키와 값을 하나의 아이템으로 저장하고 검색 수정이 가능한 추상 클래스가 있다.

```
abstract class PairMap {
    protected String keyArray []; // key 들을 저장하는 배열
    protected String valueArray []; // value 들을 저장하는 배열
    abstract String get(String key); // key 값을 가진 value 리턴. 없으면 null 리턴
    abstract void put(String key, String value); // key와 value를 쌍으로 저장. 기존에
                                                // key가 있으면, 값을 value로 수정
    abstract String delete(String key); // key 값을 가진 아이템(value와 함께) 삭제.
                                                // 삭제된 value 값 리턴
    abstract int length(); // 현재 저장된 아이템의 개수 리턴
}
```

목표 추상 클래스의 구현과 활용 연습

PairMap을 상속받는 Dictionary 클래스를 구현하고, 이를 다음과 같이 활용하는 main() 메소드를 가진 클래스 DictionaryApp도 작성하라. **난이도 7**

320

JAVA

자바 프로그래밍

JAVA PROGRAMMING

```
public static void main(String[] args) {
    Dictionary dic = new Dictionary(10);
    dic.put("황기태", "자바");
    dic.put("이재문", "파이선");
    dic.put("이재문", "C++"); // 이재문의 값을 C++로 수정
    System.out.println("이재문의 값은 " + dic.get("이재문"));
    System.out.println("황기태의 값은 " + dic.get("황기태"));
    dic.delete("황기태"); // 황기태 아이템 삭제
    System.out.println("황기태의 값은 " + dic.get("황기태")); // 삭제된 아이템 접근
}
```

이재문 아이템 출력

황기태 아이템 출력

이재문의 값은 C++
황기태의 값은 자바
황기태의 값은 null

목표 추상 클래스, 오버라이딩, 동적바인딩

11. 철수 학생은 다음 3개의 필드와 메소드를 가진 4개의 클래스 Add, Sub, Mul, Div를 작성하려고 한다.