

## 22-겨울계절 자바프로그래밍 박숙영 교수님

.java

바이트코드

1. 자바 소스 파일 확장자와 자바 소스 파일 확장자 컴파일 후 생기는 파일의 이름
2. 유효하지 않은 식별자 찾기 (4 지선다, 객관식) (안타깝, \_로 시작, 특수문자, \$외 불가 / 숫자사용가능)
3. 이름이 `studentNumbers` 이고 배열의 크기가 30 인 정수형 배열을 선언하는 코드 짜기  
`Integer[] studentNumbers = new Integer[30];`
4. 코드 계산 후 출력되는 값  
`int a = 27;`  
`int b = a++ % 5;` → b는 2, a는 28  
`string c = "7"; System.out.println(a + b + c);` 307  
+ 양옆이 숫자면 연산  
하나라도 문자면이면 문자열연산  
동일한 우선순위가면 왼쪽부터  
배열은 () 없어도 됨!
5. 배열의 각 원소를 100 으로 초기화하는 코드 짜기 → `for(int i=0; i < s.length(); i++)`  
`s[i] = 0;`
6. 주어진 코드에서 오류를 찾고 고치는 문제
7. 주어진 코드에서 오류를 찾고 고치는 문제
8. 추상클래스와 인터페이스를 정의하고 구현하는 방식 기술하기, 유사점과 차이점도 기술하기

예) Animal 클래스는  
상속받는 두 클래스  
Dog와 Cat의  
break 메서드를  
비교하는 식별자  
를 적어야 하는 것임

9. 다형성이란 무엇인지 설명하고, 자바에서 다형성을 실현하기 위한 방법을 예를 들어 설명하시오  
다형성: 동일한 이름의 동작이 객체의 식제 타입에 따라 달라짐 → 인버전 (정라와 사냥개끼리 다형성), 오버라이딩 (식제타입에서의 다형성)
10. 오버로딩과 오버라이딩을 비교 서술하시오  
오버로딩: 매개변수가 다르다면 이름이 같아도 여러개의 메서드 작성 가능  
오버라이딩: 부모 클래스의 메서드를 자식 클래스가 재정의함  
공통점: 이름 재사용
11. 13~26 사이의 값을 난수 생성하는 코드 짜기  
`int num = (int)(Math.random() * 1007 - 1.14 + 13);`  
↳ 0~13
12. public/private/protected/생략 각 접근지정자에 대한 간단한 설명쓰기
13. public: 누구나 접근 가능  
private: 클래스 안에서만 접근 가능  
protected: 자식 클래스 안에서만 접근 가능  
생략: 동일 패키지 안에서만 접근 가능
14. String s = "Happy New Year" 이라는 문장에서 "Happy"와 "Year"만 뽑아내서 이어붙이는 코드 짜기  
`s.substring(0,5) + s.substring(10)`  
↳ [0,5)    ↳ 10부터 끝까지
15. 주어진 코드에서 오류가 생성되는 이유 찾기(배열의 범위를 벗어나는 오류였음) / try catch 블럭을 사용해서 오류 수정
16. 주어진 형식에 맞게 출력되는 Circle 클래스 생성하는 코드 짜기 - 클래스의 멤버변수 선언하기, 생성자 선언하기, 메소드 선언하기

총 16 문제 중 15 문제 기술하였습니다.

Lab0630 문제

```
for(int i=0; i < s.length(); i++) {
    try {
        count[s.charAt(i) - 'A']++;
    }
    catch (ArrayIndexOutOfBoundsException e) {
        System.out.println("ArrayIndexOutOfBoundsException 발생!");
    }
}
```

Lab0630 문제

```
class Circle {
    private int x, y, radius;
    public Circle(int x, int y, int radius) {
        super();
        this.x = x;
        this.y = y;
        this.radius = radius;
    }
    public int getX() { return x; }
    public void setX(int x) { this.x = x; }
    :
    public String toString() {
        return " ";
    }
}
```