

Come to HUFS Meet the World!

객체지향프로그래밍

24.9.13 실습

- 담당 교수 : 전 병 환 교수님
- 조교 : 윤 종 업
- 조교 메일 : juyoon@hufs.ac.kr



객체지향 프로그래밍 실습

- 주차 별 이론(2시간)을 기반으로 실습(1시간) 진행
 - 특정 요일 자정까지 실습과제 제출(추석연휴 이후 실습 일자 공지)
 - 지각제출 허용 (단, 지각 제출 하루 당 **10% 감점**)

ex. 13일 금요일 실습진행

→ 17일 화요일 자정까지 실습과제 제출, 19일 목요일 자정까지 지각제출 허용

최종 제출시간 기준으로 점수가 채점됨

→ 기한 내에 제출했더라도 지각제출허용기간에 다시제출하면 점수 감점

객체지향 프로그래밍 실습

- 실습시간에 개인 노트북 필수 지참

1. 실습코드 캡처

2. 코드관련 설명

3. 결과 출력 화면 캡처

4. 과제제출양식

- PDF파일로 제출
- [파일명] 이름_학번_학과.pdf

자바 프로그래밍의 구조

```
1  /*
2  * 소스 파일 : Hello.java
3  */
4  public class Hello {
5
6      public static int sum(int n, int m) {
7          return n + m;
8      }
9
10     // main() 메소드에서 실행 시작
11     public static void main(String[] args) {
12         int i = 20;
13         int s;
14         char a;
15
16         s = sum(i, 10);           // 메소드 호출
17         a = '?';
18         System.out.println(a);   // 문자 '?' 출력
19         System.out.println("Hello"); // "Hello" 문자열 출력
20         System.out.println(s);   // 정수 s 값 30 출력
21     }
22 }
```

메소드

클래스

메소드

실습 1: 자바 프로그래밍의 구조

- 예제 2-1에서 주어진 코드를 활용하여 계산기 코드를 작성하여라
 - 변수와 메소드는 모두 float로 정의한다.
 - 덧셈 메소드를 sum, 뺄셈 메소드를 sub, 곱셈 메소드를 mul, 나눗셈 메소드를 div로 정의한다.

– 출력 예시

```
2.0 + 4.0 = 6.0
2.0 - 4.0 = -2.0
2.0 * 4.0 = 8.0
2.0 / 4.0 = 0.5
```

```
public class Calculator {
    public static float sum(float n, float m) {
    }
    public static float sub(float n, float m) {
    }
    public static float mul(float n, float m) {
    }
    public static float div(float n, float m) {
    }
}
```

조건 연산자 ?:

- Condition ? Opr2 : Opr3
 - 세 개의 피연산자로 구성된 삼항 연산자
 - Condition이 true이면, 연산식의 결과는 opr2, false이면 opr3
 - if-else를 간결하게 표현할 수 있음

다음은 조건 연산자의 사례이다. 실행 결과는 무엇인가?

```
1 public class TernaryOperator {  
2     public static void main (String[] args) {  
3         int a = 3, b = 5;  
4  
5         System.out.println("두 수의 차는 " + ((a>b)?(a-b):(b-a)));  
6     }  
7 }
```

→ 실행 결과

두 수의 차는 2

실습 2: 조건 연산자 ?:

- 숫자 3개를 입력 받아 가장 큰 숫자를 출력하는 코드를 작성하라
 - 변수와 메소드는 모두 int로 정의한다.
 - 숫자 3개를 비교하는 메소드를 compare()로 정의한다

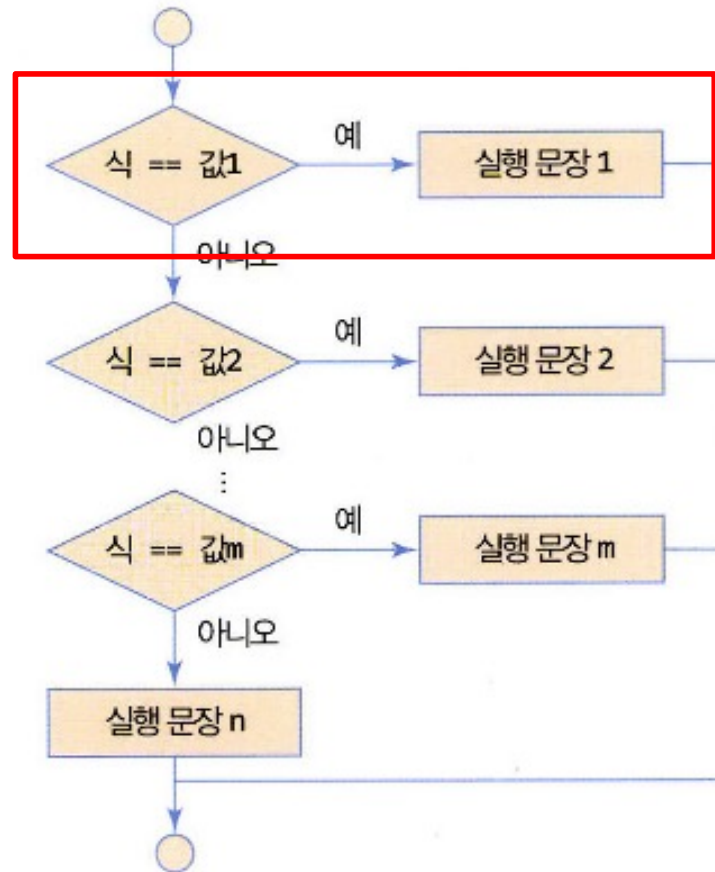
```
public class Comparison {  
    public static int compare(int num1, int num2, int num3) {  
        (여기에 코드 작성)  
    }  
}
```

- 출력 예시

숫자 1를 입력하세요: 10
숫자2를 입력하세요: 20
숫자3를 입력하세요: 30
가장 큰 숫자는 30입니다.

Switch 문

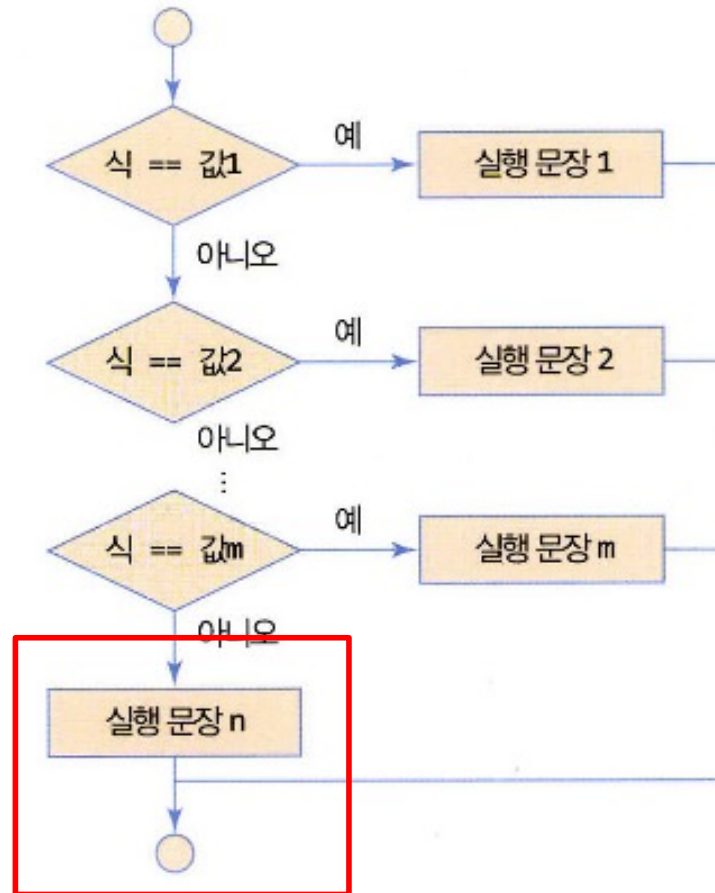
```
switch (식) {  
  case 값1:  
    실행 문장 1;  
    break;  
  case 값2:  
    실행 문장 2;  
    break;  
  ...  
  case 값m:  
    실행 문장 m;  
    break;  
  default:  
    실행 문장 n;  
}
```



- switch문은 식과 case 문의 값과 비교
 - case의 비교 값과 일치하면 해당 case의 실행문장 수행
 - break를 만나면 switch문을 벗어남

Switch 문

```
switch (식) {  
  case 값1:  
    실행 문장 1;  
    break;  
  case 값2:  
    실행 문장 2;  
    break;  
  ...  
  case 값m:  
    실행 문장 m;  
    break;  
  default:  
    실행 문장 n;  
}
```



- switch문은 식과 case 문의 값과 비교
 - case의 비교 값과 일치하는 것이 없으면 default 문 실행
 - default문은 생략 가능

실습 3: Switch 문

- Switch문을 사용하여 현재 월을 입력 받아 입력 받은 월에 해당하는 계절을 출력하는 프로그램을 작성하라.

- 3~5월 “현재 계절은 봄 입니다.” / 6~8월 “현재 계절은 여름 입니다.”
9~11월 “현재 계절은 가을 입니다.” / 12~2월 “현재 계절은 겨울 입니다.”

- 입력 월이 1~12월이 아닐 때는 “이러가이 자모 디어스입니다”를 출력

- 출력 예시

현재 월을 입력하세요. : 5
현재 계절은 봄입니다.

```
public class Season {  
  
    public static void main(String[] args) {  
        // TODO Auto-generated method stub  
        Scanner scanner = new Scanner(System.in);  
  
        System.out.print("현재 월을 입력하세요. : ");  
        int month = scanner.nextInt();  
  
        String season = "겨울";  
        String a = "현재 계절은 %s입니다.";
```

(여기에 코드 작성)