

# 제 5 장

## 프로그램 흐름제어

# 프로그램 흐름제어분류

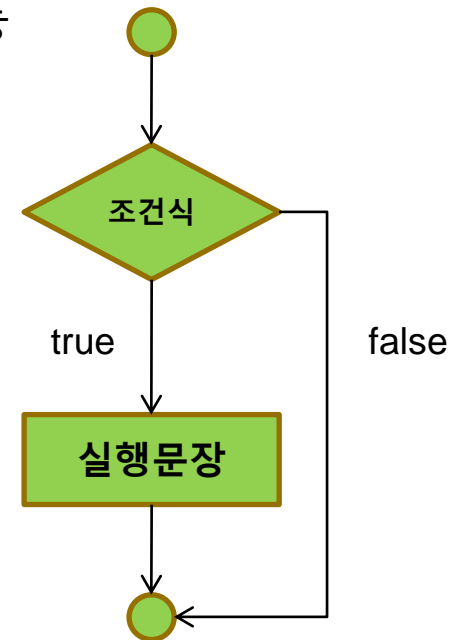
- 조건 (흐름제어) 명령문
  - 특정 조건을 테스트하여 프로그램 흐름을 결정
  - if, if-else, switch – nested if
- 반복 (흐름제어) 명령문
  - 특정 부분을 반복하여 실행
  - while, do-while, for – nested while, nested for, ..
- 분기 (흐름제어) 명령문
  - 특정 부분으로 분기하여 명령문 실행
  - break, continue, return, label:
- 예외 발생의 경우 (흐름 제어) 명령문
  - 예외처리 명령문 : try-catch-finally, throw(s)
  - 분기 흐름제어에 포함할 수 있다

# 조건 명령문 - if

## ■ 단순 if 문

- if 다음의 괄호 안에는 조건식(논리형 변수나 논리 연산)
- 조건식의 값
  - true인 경우, if문을 벗어나 다음 문장이 실행된다.
  - false의 경우에는 if 다음의 문장이 실행되지 않고 if 문을 빠져 나온다.
- 실행문장이 단일 문장인 경우 중괄호 { } 생략 가능

```
if (조건식) {  
    실행문장  
}
```



## 단순 if문 예

시험 점수가 80점이 이상이면 합격을 판정 하는 if문 프로그램

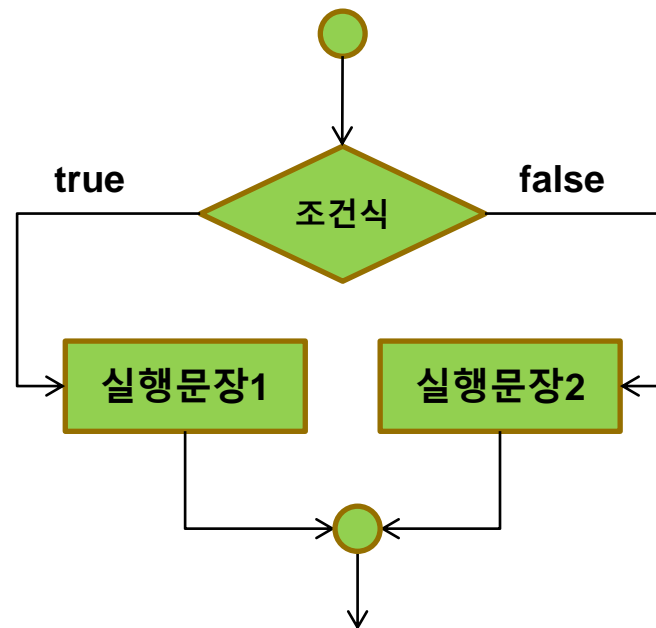
```
import java.util.Scanner;
public class SuccessOrFail {
    public static void main (String[] args) {
        Scanner in = new Scanner(System.in);
        System.out.print("점수를 입력하시오: ");
        int score = in.nextInt();
        if (score >= 80)
            System.out.println("축하합니다! 합격입니다.");
    }
}
```

점수를 입력하시오: 95  
축하합니다! 합격입니다.

## ■ if-else 문

- ❑ 조건식이 true면 실행문장1 실행 후 if-else문을 벗어남
- ❑ false인 경우에 실행문장2 실행후, if-else문을 벗어남

```
if (조건식) {  
    실행문장1  
}  
else  
{  
    실행문장2  
}
```



## if-else 문 예

입력된 수가 3의 배수인지 판별하는 프로그램

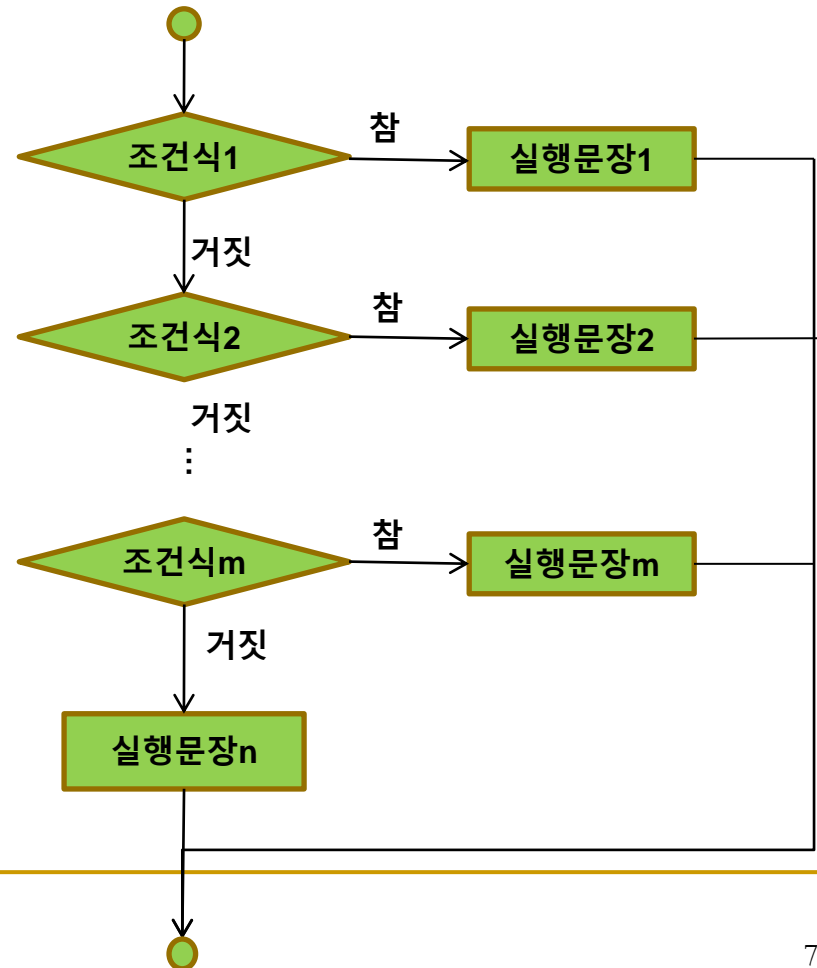
```
import java.util.Scanner;
public class MultipleOfThree {
    public static void main (String[] args) {
        Scanner in = new Scanner(System.in);
        System.out.print("수를 입력하시오: ");
        int number = in.nextInt();
        if (number % 3 == 0)
            System.out.println("3의 배수입니다.");
        else
            System.out.println("3의 배수가 아닙니다.");
    }
}
```

수를 입력하시오: 129  
3의 배수입니다.

## ■ 다중 if문(if-else ... f-else)

- ❑ 실행문장이 다시 if문 또는 if-else문을 포함
- ❑ else 문은 바로 전의 if문과 짝을 이룬다.
- ❑ 조건문이 너무 많은 경우에는 switch 문을 사용하는 것이 좋다

```
if(조건식1) {  
    실행문장1; // 조건식1이 참인 경우.  
}  
else if(조건식2) {  
    실행문장2; // 조건식2가 참인 경우.  
}  
else if(조건식 m) {  
    실행문장m; // 조건식m이 참인  
    경우.  
}  
else {  
    실행문장n; // 앞의 모든 조건이 거짓  
    인 경우.  
}
```



# 다중 if-else를 사용하는 학점 부여 예

다중 if-else문을 이용하여 키보드 입력된 성적에 대해 학점을 부여하는 프로그램

```
import java.util.Scanner;
public class Grading {
    public static void main (String[] args) {
        char grade;
        Scanner a = new Scanner(System.in);
        while (a.hasNext()) {
            int score = a.nextInt();
            if(score >= 90.0) // score가 90.0 이상인 경우
                grade = 'A';
            else if(score >= 80.0) // score가 80.0 이상이면서 90.0 미만인 경우
                grade = 'B';
            else if(score >= 70.0) // score가 70.0 이상이면서 80.0 미만인 경우
                grade = 'C';
            else if(score >= 60.0) // score가 60.0 이상이면서 70.0 미만인 경우
                grade = 'D';
            else // score가 60.0 미만인 경우
                grade = 'F';
            System.out.println("학점은 "+grade+"입니다");
        }
    }
}
```

80  
학점은 B입니다

90  
학점은 A입니다

76  
학점은 C입니다



## if문과 3항 연산자 ?:

- 3항 연산자 ?: 는 if-else로 바꿀 수 있다

```
i = a > b ? a - b : b - a;
```



```
if (a > b)
    i = a - b;
else
    i = b - a;
```

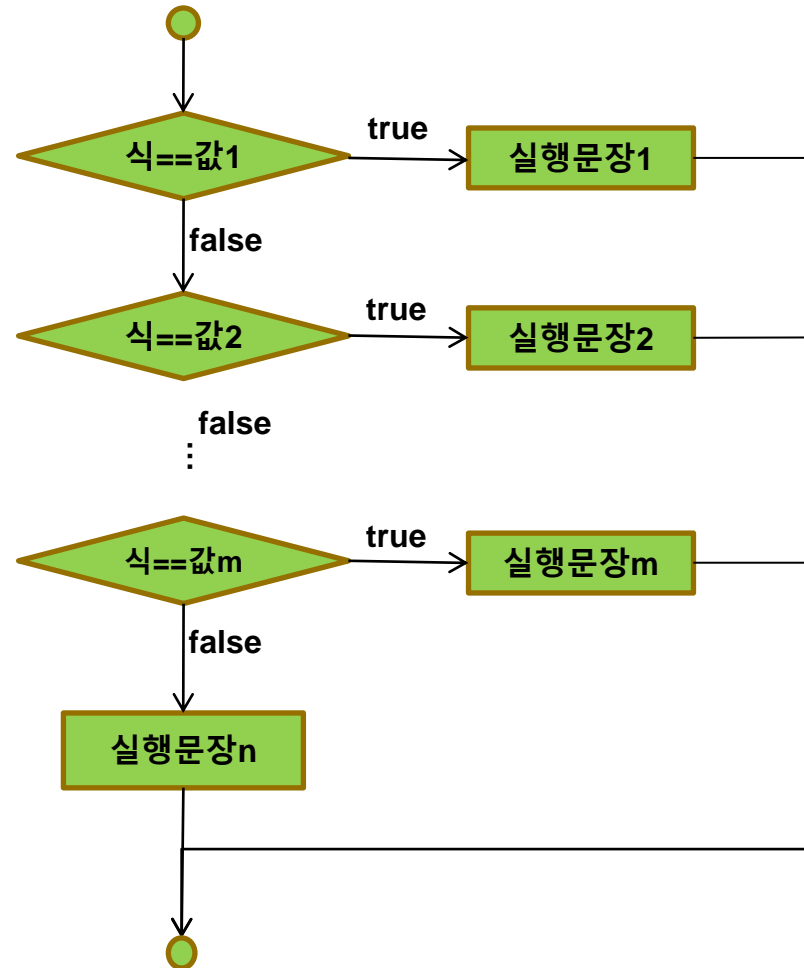
## switch문

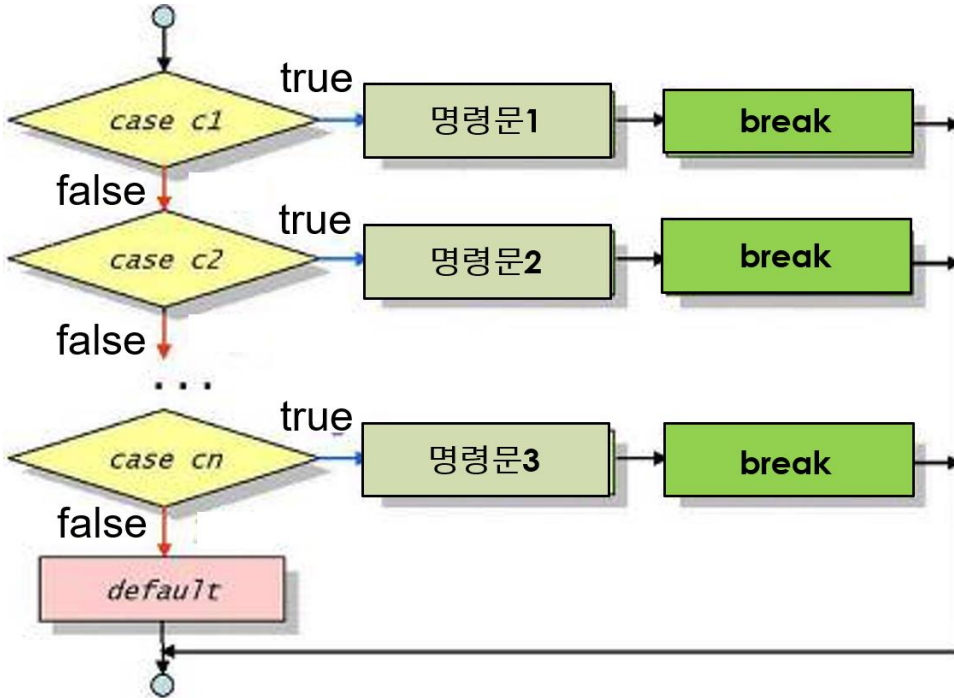
- 가능한 실행경로가 여럿인 경우, 다중 if-else보다 switch 문을 사용하는 것이 편리
- switch문은 **하나의 식**에 대해 결과값만 비교할 수 있다.
  - case문의 비교 값은 기본데이터 값(주로 정수형) 또는 enum이 될 수 있다
  - 자바 8.0부터는 String 타입 및 기본타입을 객체화시킨 몇 개의 객체 타입도 비교 가능
- 비교값과 일치하면 해당 case문을 수행하고 break를 만나면 switch문을 벗어남
- 일치하는 비교값이 없으면 default 문 실행.
- default문은 생략 가능

```

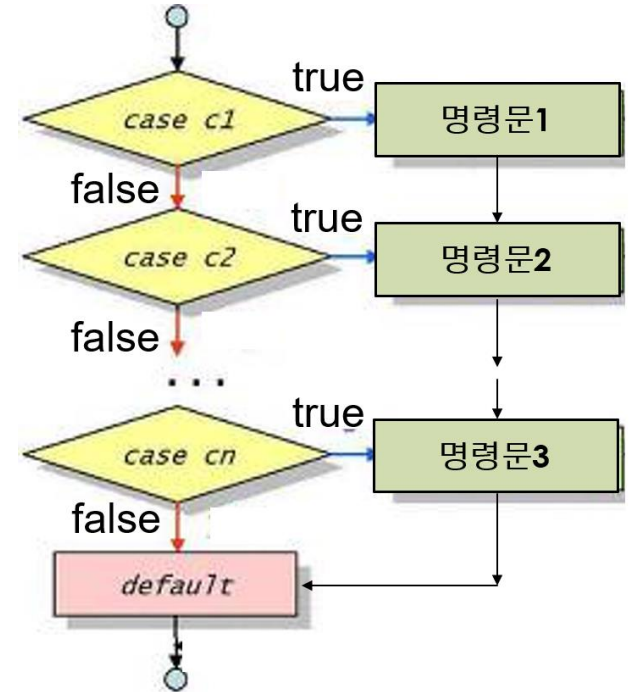
switch (식) {
  case 값1:
    실행문장1
    break;
  case 값2:
    실행문장1
    break;
  ...
  case 값m:
    실행문장m
    break;
  default:
    실행문장n
}

```





일반적인 “switch” 조건 명령문의 실행 흐름 제어




“break” 명령문을 사용하지 않는  
경우 “switch” 문의 실행 흐름 제어

## ■ switch문 내의 break문

- break 명령문을 만나면 switch문 탈출
- 만일 case 문에 break문이 없다면 다음 case문의 실행문장으로 실행을 계속하게 되며 break를 만날 때까지 계속 내려감

```
char grade='A';  
switch (grade) {  
    case 'A':  
        System.out.println("90 ~ 100점입니다.");  
        break;  
    case 'B':  
        System.out.println("80 ~ 89점입니다.");  
        break;  
    case 'C':  
        System.out.println("70 ~ 79점입니다.");  
        break;  
}
```



90 ~ 100점입니다.  
80 ~ 89점입니다.

```
public class SwitchTest {  
    public static void main(String[] args) {  
        int month = 8;  
        String monthString; // 참조 타입의 변수  
  
        switch (month) {  
            case 1: monthString = "January";  
                System.out.println(monthString);  
            case 2: monthString = "February";  
                System.out.println(monthString);  
            case 3: monthString = "March";  
                System.out.println(monthString);  
            case 4: monthString = "April";  
                System.out.println(monthString);  
            case 5: monthString = "May";  
                System.out.println(monthString);  
            case 6: monthString = "June";  
                System.out.println(monthString);  
            case 7: monthString = "July";  
                System.out.println(monthString);  
            case 8: monthString = "August";  
                System.out.println(monthString);  
            case 9: monthString = "September";  
                System.out.println(monthString);  
            case 10: monthString = "October";  
                System.out.println(monthString);  
            case 11: monthString = "November";  
                System.out.println(monthString);  
            case 12: monthString = "December";  
                System.out.println(monthString);  
            default: monthString = "Invalid month";  
                System.out.println(monthString);  
        }  
        System.out.println(monthString);  
    }  
}
```

August

September

October

November

December

Invalid month

Invalid month

```

class SwitchTest2 {
    public static void main(String[] args) {
        int month = 2;
        int year = 2000;
        int numDays = 0;
        switch (month) {
            case 1:
            case 3:
            case 5:
            case 7:
            case 8:
            case 10:
            case 12:
                numDays = 31;
                break;
        }
    }
}

```

**Number of Days = 29**

```

        case 4:
        case 6:
        case 9:
        case 11:
            numDays = 30;
            break;
        case 2:
            if (((year % 4 == 0) && !(year % 100 == 0)) ||
                (year % 400 == 0))
                numDays = 29;
            else
                numDays = 28;
            break;
        default:
            System.out.println("Invalid month.");
            break;
    }
    System.out.println("Number of Days = " + numDays);
}

```

학점이 A, B 인 학생에게는 "참 잘하셨습니다.", 학점이 C, D인 학생에게는 "좀 더 노력하세요.", 학점이 F인 학생에게는 "다음 학기에 다시 수강하세요."를 출력하는 프로그램을 switch문의 break를 잘 활용하여 작성하여라.

```
public class GradeSwitch {  
    public static void main(String[] args) {  
        char grade='C';  
        switch (grade) {  
            case 'A':  
            case 'B':  
                System.out.println("참 잘하셨습니다.");  
                break;  
            case 'C':  
            case 'D':  
                System.out.println("좀 더 노력하세요.");  
                break;  
            case 'F':  
                System.out.println("다음 학기에 다시 수강하세요.");  
                break;  
            default:  
                System.out.println("잘못된 학점입니다.");  
        }  
    }  
}
```

좀 더 노력하세요.



- case문의 값의 특징
  - switch문은 식의 결과 값을 case 문과 비교
  - case문에 지정하는 비교 값은 보통 정수 타입 리터럴
  - 자바 8.0부터는 String 타입 및 기본타입을 객체화시킨 몇 개의 객체 타입도 비교 가능

```
int a = 0;
int b = 1;
int c = 25;
switch(c%2) {
    case a : // 오류, 변수를 사용할 수 없음
        ...;
        break;
    case 1 : // 정상
        ...;
        break;
}
```

```
switch(a) {
    case a > 3 : // 오류
    case a == 1 : // 오류
}
```

앞의 다중 if문을 이용한  
성적 분류 프로그램을  
switch문으로 바꾸시오.

100  
학점은 A입니다  
55  
학점은 F입니다  
76  
학점은 C입니다

```
import java.util.Scanner;
public class Grading2 {
    public static void main (String[] args) {
        char grade;
        Scanner a = new Scanner(System.in);
        while (a.hasNext()) {
            int score = a.nextInt();
            switch (score/10) {
                case 10:
                case 9:
                    grade = 'A';
                    break;
                case 8:
                    grade = 'B';
                    break;
                case 7:
                    grade = 'C';
                    break;
                case 6:
                    grade = 'D';
                    break;
                default:
                    grade = 'F';
            }
            System.out.println("학점은 "+grade+"입니다");
        }
    }
}
```

## ■ if-else 문 vs. switch 문

- 코드의 가독성
- switch문의 조건테스트 방식
- if-else 문의 조건을 나타내는 수식은 일반적으로 **범위를 가진 값** 혹은 **boolean 조건을 테스트**
- switch문은 **하나의 정수, enum 값** 혹은 **String 객체**일 경우 사용한다.