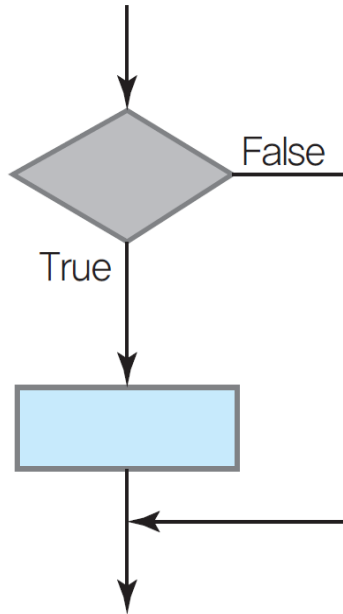


조건문 *(Condition Statement)*

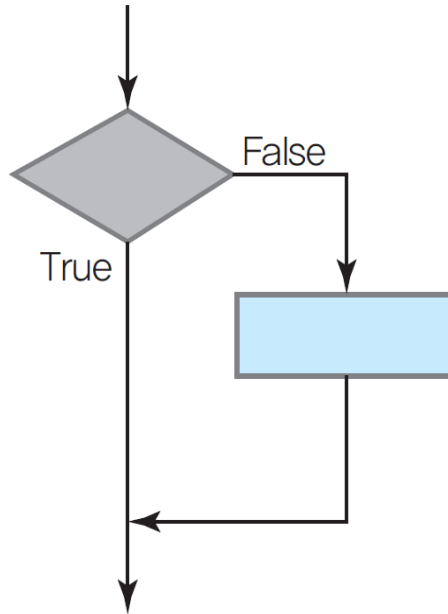
알고리즘의 구조: 조건 구조

- ❖ 조건에 따라 수행되는 작업들이 다른 경우
 - 조건이 맞으면 '참(True)'으로 표시된 방향의 작업 수행
 - 조건이 틀리면 '거짓(False)'으로 표시된 방향의 작업 수행

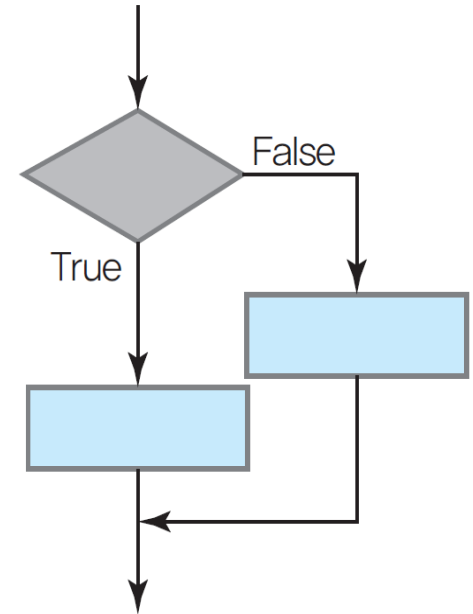
참의 경우에만
수행되는 구조



거짓의 경우에만
수행되는 구조



참과 거짓에 따라 다른
작업을 수행되는 구조



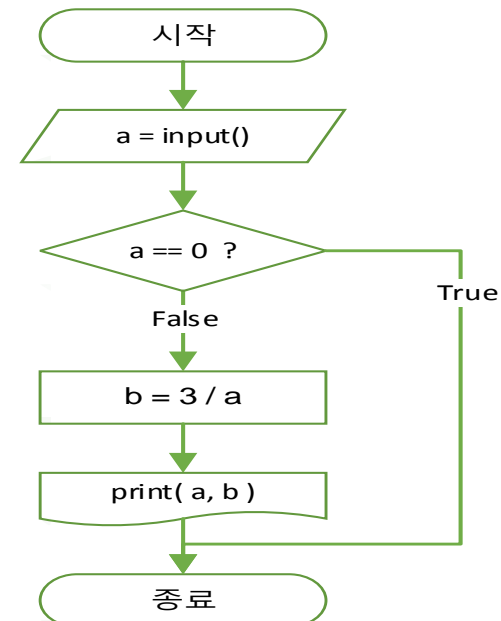
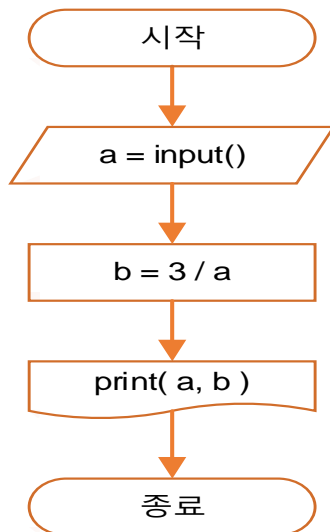
분기문

❖ 프로그램의 흐름을 가르는 문장 : 나눌 분(分), 갈림길 기(岐)

■ 프로그래밍에서 '흐름(*flow*)'은 컴퓨터에 내려지는 명령의 순서

1. a에 사용자로부터 입력받은 수를 저장하라.
2. b에 $3 \div a$ 의 결과를 저장하라.
3. a와 b를 출력하라.

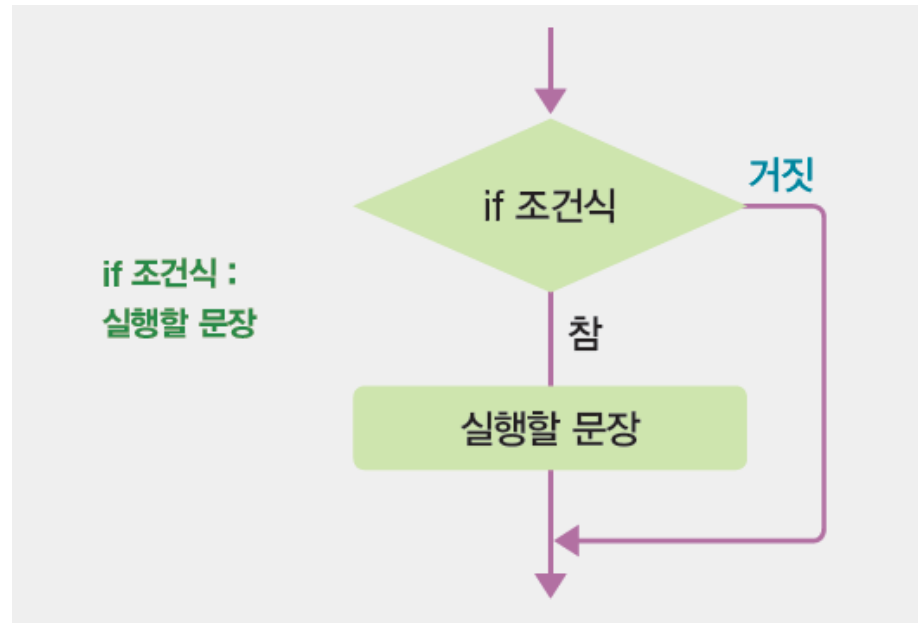
1. a에 사용자로부터 입력받은 수를 저장하라.
2. a가 0**이라면** 프로그램을 종료하고, **그렇지 않으면** 다음 명령을 수행하라.
3. b에 $3 \div a$ 의 결과를 저장하라.
4. a와 b를 출력하라.



if 문 (1)

❖ if 조건식 :

- 조건식이 참이면 실행할 문장이 처리되고, 거짓이면 아무것도 실행하지 않고 프로그램을 종료



if 문 (2)

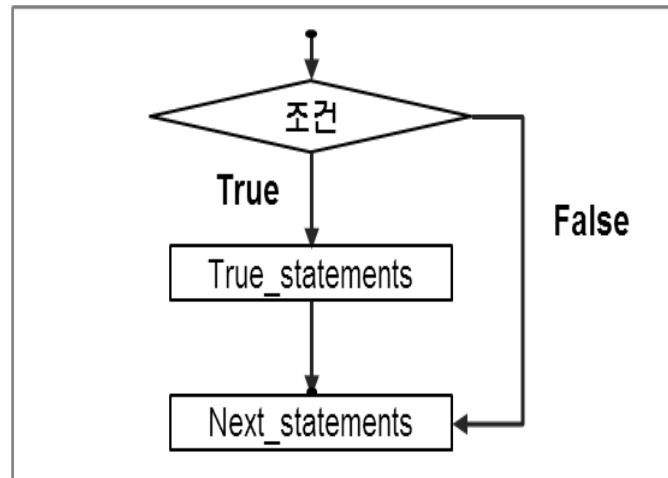
if **조건** :

 True_statements

Next_statements

"조건"이 True 이면 True_statements를 실행하고 Next_statement를 실행

"조건"이 False 이면 바로 Next_statement를 실행



코드블록과 들여쓰기

❖ 코드블록(*Code Block*)

- 여러 줄의 코드가 이루는 일정한 구역
- 프로그래밍 언어들마다 이것을 표현하는 방법이 다름

❖ 파이썬에서는 들여쓰기로 구역을 나눔

- 들여쓰기는 스페이스(*Space*)나 탭(*Tab*) 둘 다 사용 가능
- 파이썬을 만든 귀도는 *PEP(Python Enhancement Proposals)*에서 스페이스 4칸을 사용할 것을 권장

if (조건):

코드
블록

True_statements1

True_statements2

True_statements3

```
>>> if a>b:  
    print('True')  
    print('***')
```

SyntaxError: unexpected indent

Error

if 문 들여쓰기 예제

```
1 a=200
2
3 if a<100 :
4     print("100보다 작군요")
5 print("거짓이므로 이 문장은 안 보이겠죠?")
6
7 print("프로그램 끝")
```

거짓이므로 이 문장은 안 보이겠죠?
프로그램 끝

```
1 a=200
2
3 if a<100 :
4     print("100보다 작군요")
5     print("거짓이므로 이 문장은 안보이겠죠?")
6
7 print("프로그램 끝")
```

프로그램 끝

if 문 예제 (1)

하나에 1,000원하는 연필과 하나에 2,000원하는 펜이 있다.

구입 시 10,000원이 넘으면 10%를 할인해 준다고 할 때,

Python 소스 파일을 생성하여 다음 조건에 만족하는 프로그램을 작성해보자.

▶ 구매하고자 하는 연필과 펜의 개수를 사용자로부터 입력받는다.

▶ 조건문을 활용하여 할인율을 적용한다.

● 문제 해결 알고리즘

합계 = (1000 * 연필의 개수 + 2000 * 펜의 개수)

if 합계 > 10000 → 10% 할인 → 합계 = (1 - 0.1) * 합계

출력 합계

·합계가 10000원을 넘는 경우에만 합계 가격이 할인된 가격으로 다시 계산되어 출력된다.

·합계가 10000원을 넘지 않는 경우에는 원래 합계가 그대로 출력된다.

if 문 예제 (2)

- 프로그램

```
price_pencil = 1000
price_pen = 2000

num_pencil = int(input("pencil 개수 입력 : "))
num_pen = int(input("pen 개수 입력 : "))

total_price = price_pencil * num_pencil + price_pen * num_pen

if total_price > 10000:
    total_price = total_price * 0.9
    print("10% 할인되었습니다.")

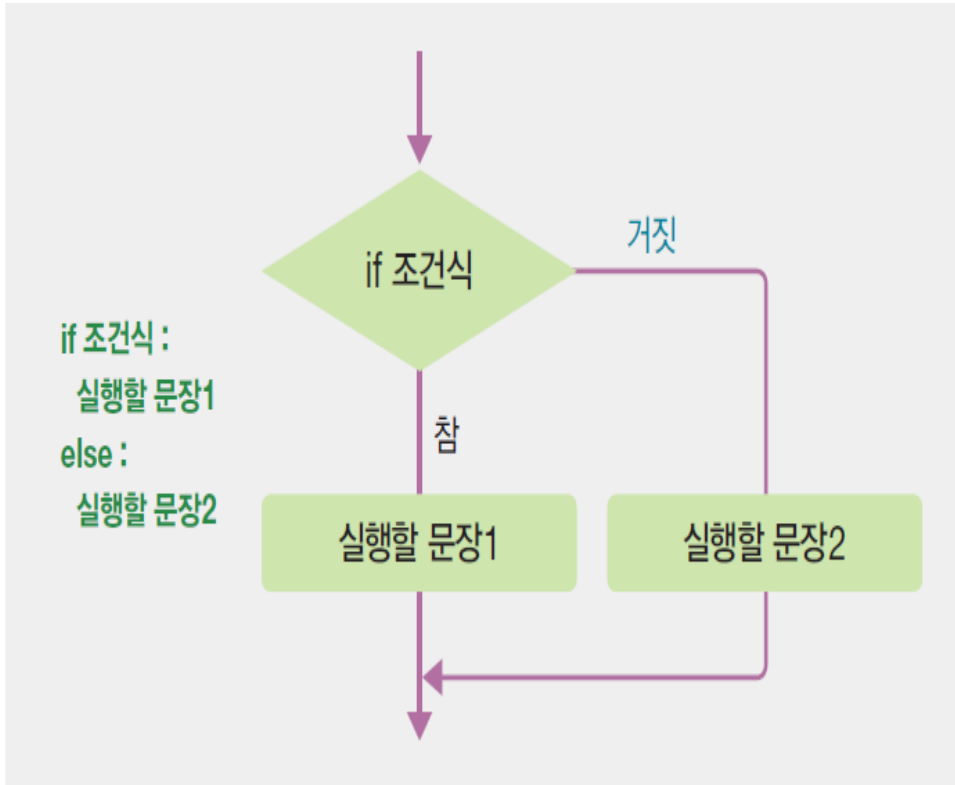
print("총합 : ", int(total_price), "원")
```

- 테스트

```
>>>
pencil 개수 입력: 5
pen 개수 입력: 3
10% 할인되었습니다.
총합 : 9900 원
>>>
pencil 개수 입력: 8
pen 개수 입력: 8
10% 할인되었습니다.
총합 : 21600 원
```

if ~ else 문 (1)

- ❖ 참일 때 실행하는 문장과 거짓일 때 실행하는 문장이 다를 때 사용

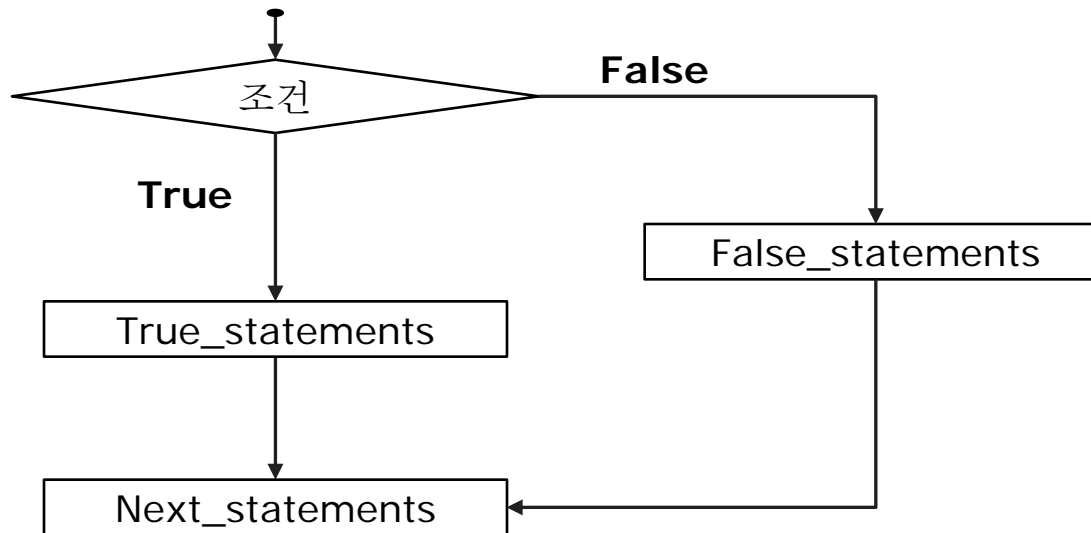


```
if(조건식):  
    True_statements1  
    True_statements2  
else:  
    False_statements1  
    False_statements2
```

if ~ else 문 (2)

```
if 조건 :  
    True_statements  
else :  
    False_statements  
Next_statements
```

‘조건’이 True 이면 True_statements를 실행하고
Next_statement로 이동,
False 이면 False_statements를 실행한 후
Next_statement로 이동



if ~ else 문 들여쓰기 예제

```
1  a=200
2
3  if a<100 :
4      print("100보다 작군요.")
5      print("참이면 이 문장도 보이겠죠?")
6  else :
7      print("100보다 크군요.")
8      print("거짓이면 이 문장도 보이겠죠?")
9
10 print("프로그램 끝!")
```

100보다 크군요.
거짓이면 이 문장도 보이겠죠?
프로그램 끝!

if ~ else 문 예제 (1)

❖ 백화점 할인을 적용하기

백화점 내의 의류 매장에서는 티셔츠(10,000원)와 스웨터(30,000원)를 구매 금액에 따라 할인 행사 중이다. 10만원 이하 구매 시 구매 금액의 5%를 할인해주고, 10만원 초과 구매 시 구입 금액의 15%를 할인해준다고 한다.

티셔츠와 스웨터 구매에 따른 총 구매 값이 얼마인지 계산해보자.

● 문제 해결 알고리즘

```
총 금액 = 10000 * 티셔츠의 개수 + 30000 * 스웨터의 개수
if 총 금액 <= 100000 → 5% 할인 → 총 금액 = ((1 - 0.05) * 총 금액)
else → 15% 할인 → 총 금액 = ((1 - 0.15) * 총 금액)
총 금액 출력
```

● 테스트

```
>>>
T-shirts 개수 입력 : 5
Sweater 개수 입력 : 20
합계 : 552500 원
>>>
```

if ~ else 문 예제 (2)

❖ 놀이기구의 입장 제한

놀이공원의 어린이용 놀이기구는 키는 155cm 이하, 몸무게는 50kg 이하의 제한 조건이 있다고 한다.
길동이의 키는 165cm, 몸무게는 60kg이다. 이 어린이용 놀이기구를 탈 수 있는지 확인해 보아라.

● 테스트

```
>>>  
키를 입력하세요 : 165  
몸무게를 입력하세요 : 60  
입장 불가능  
>>>
```