SELECTION

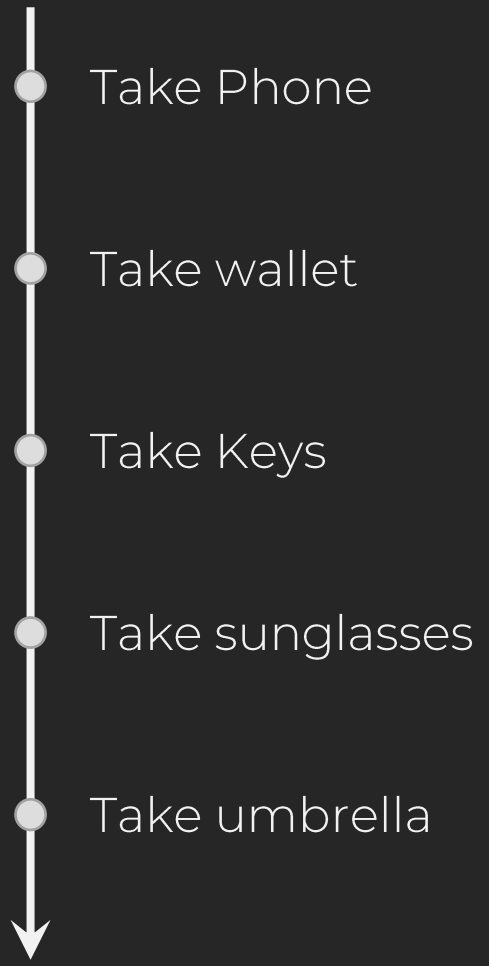
if-elif-else statement

by

Lilian Blot

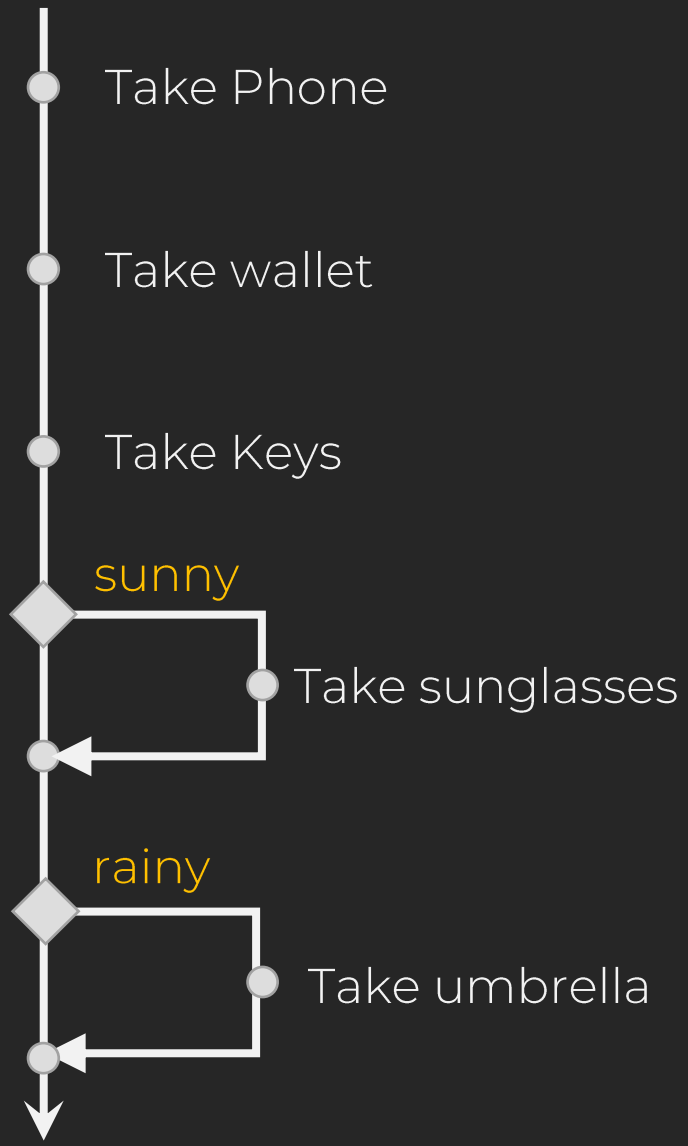
Sequencing, that is executing a set a statements in sequence, from top to bottom, allows us to write only very simple and basic programs.

Sequencing, that is executing a set a statements in sequence, from top to bottom, allows us to write only very simple and basic programs.



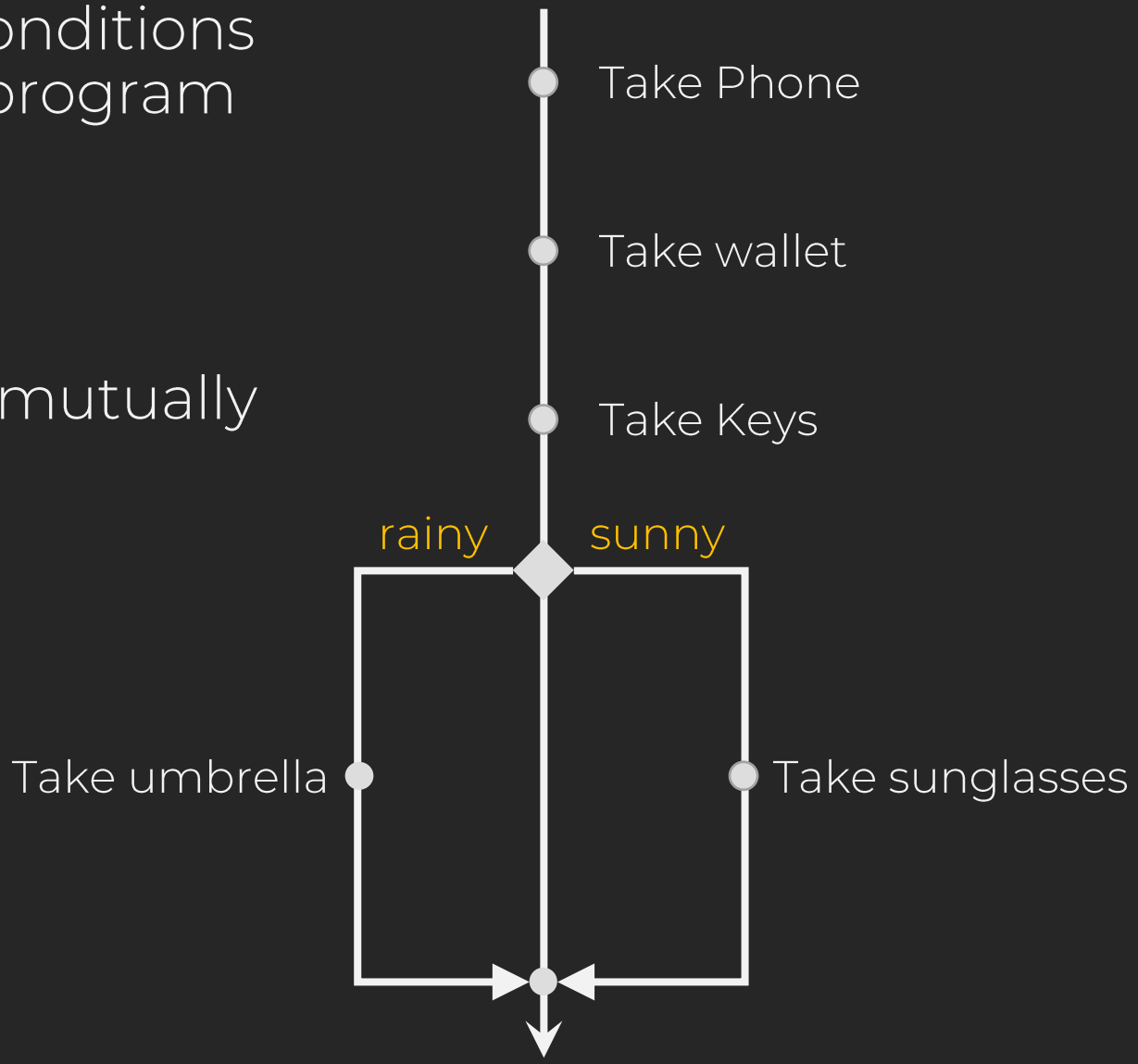
In order to write useful programs, we almost always need the ability to check conditions and change the behaviour of the program accordingly.

In order to write useful programs, we almost always need the ability to check conditions and change the behaviour of the program accordingly. It is sometimes called branching.

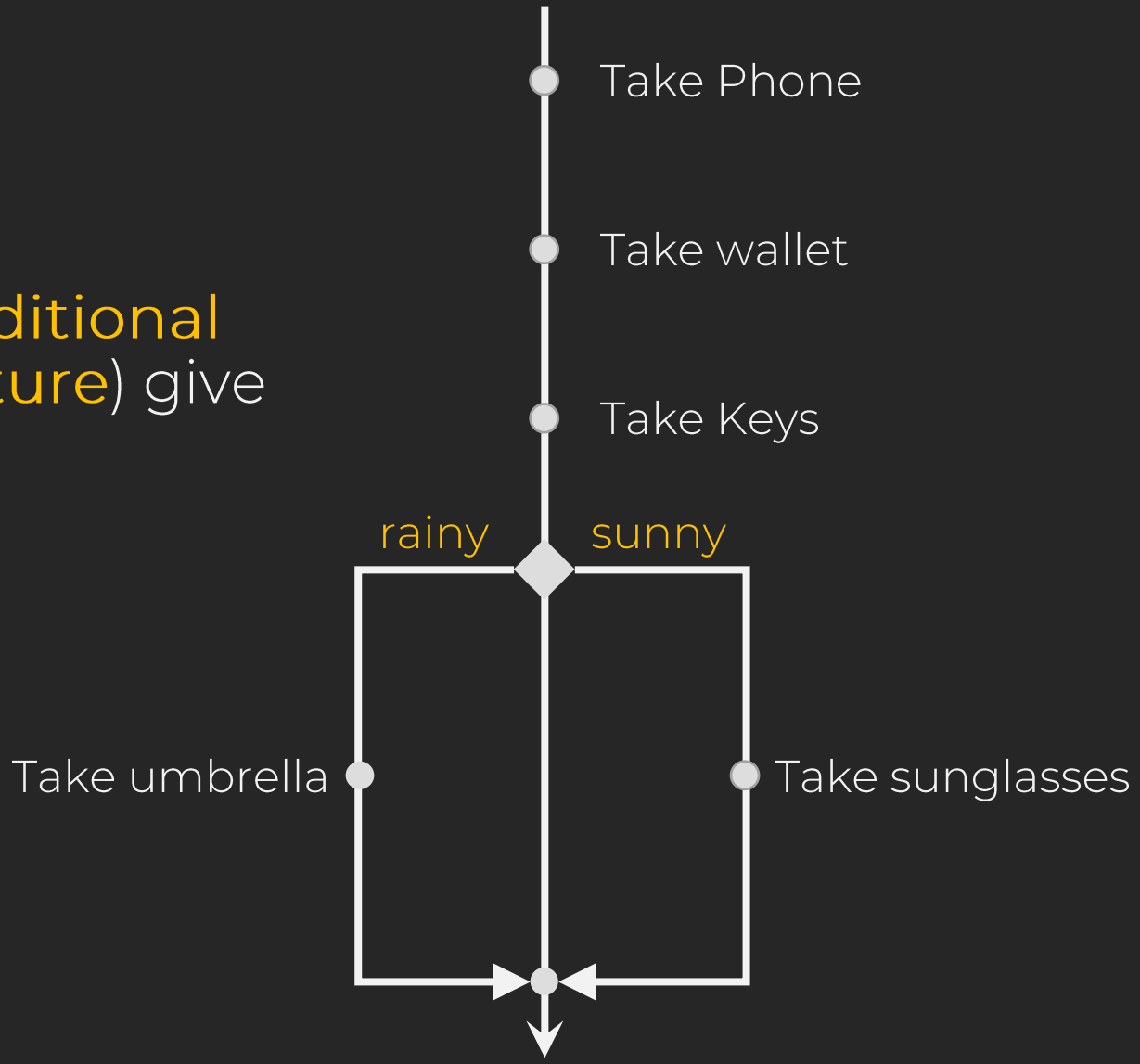


In order to write useful programs, we almost always need the ability to check conditions and change the behaviour of the program accordingly. It is sometimes called branching.

Ideally, we should be able to have mutually exclusive branches.

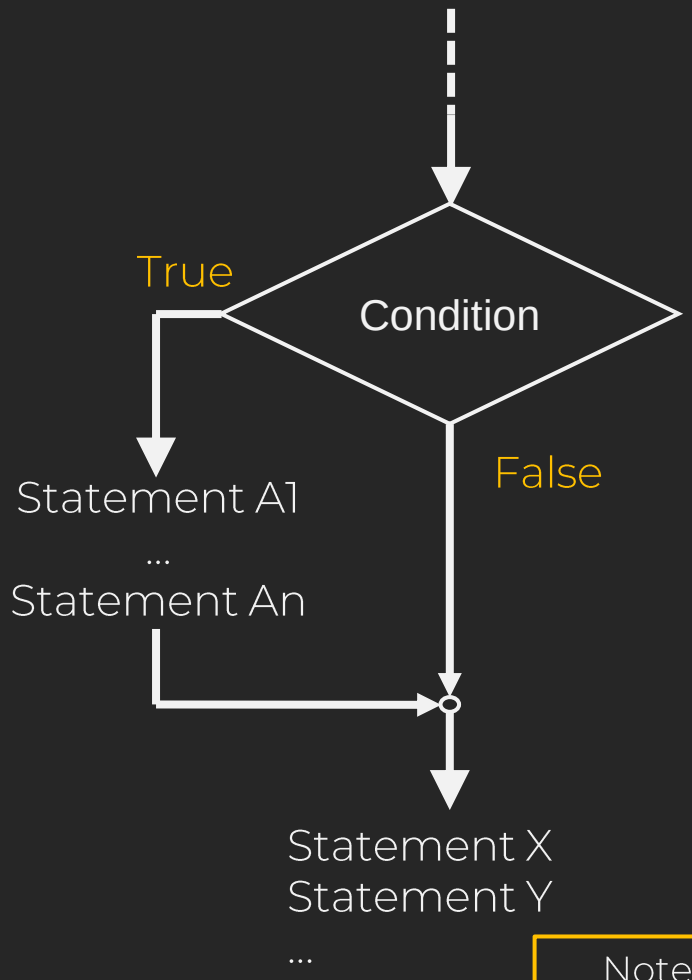


In a programming language, Conditional statements (a.k.a. Selection structure) give us this ability.



A simple `if` statement

Schema

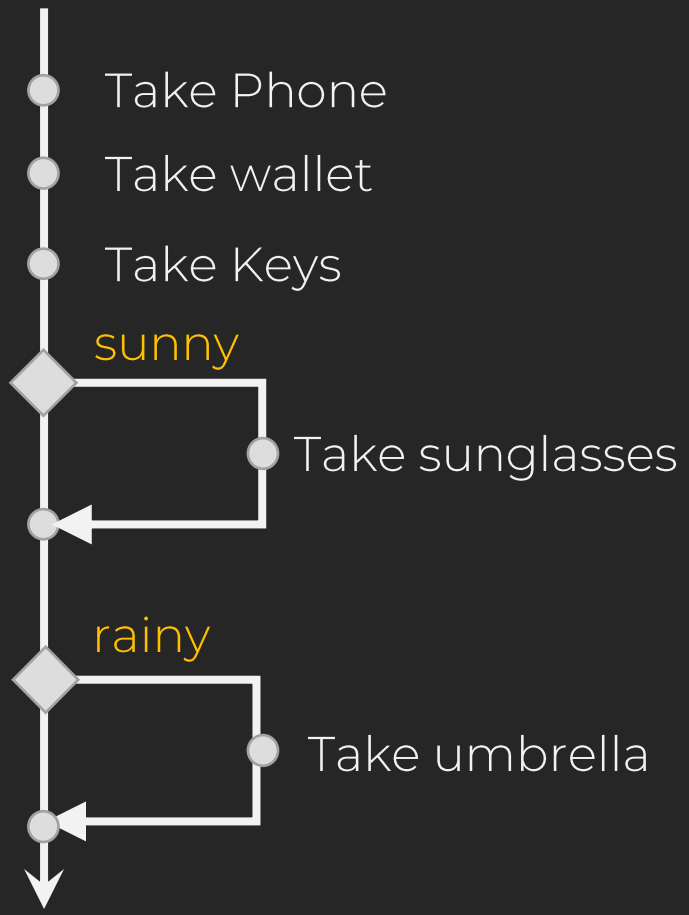


Python Code

```
1  ...
2  if condition : # a Boolean expression
3      statement A1
4      ...
5      statement An
6
7  statement X
8  statement Y
9  ...
```

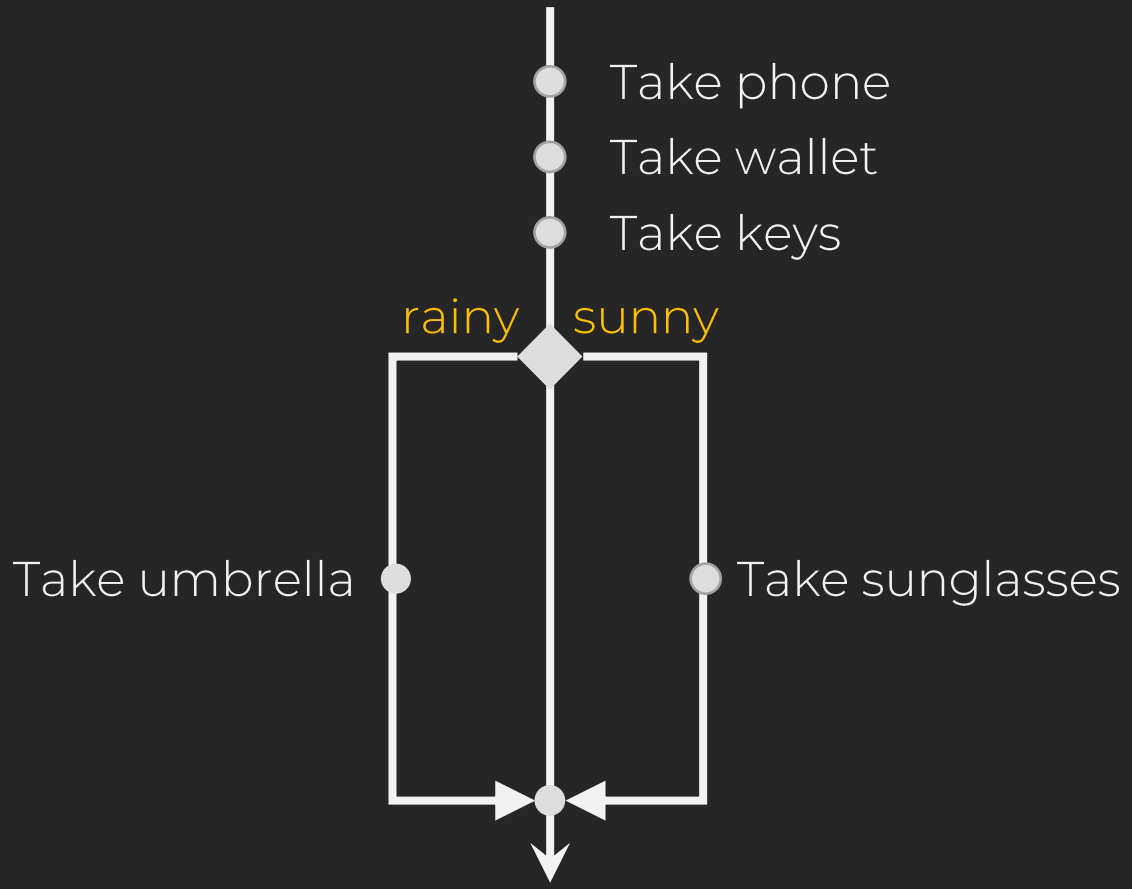
Note the change of indentation

A simple `if` statement




```
1  print('take phone')
2  print('take wallet')
3  print('take keys')
4  if weather == 'sunny' :
5      print('Take sunglasses')
6
7  if weather == 'rainy' :
8      print('Take umbrella')
9
```

Mutually Exclusive `if-elif` statement



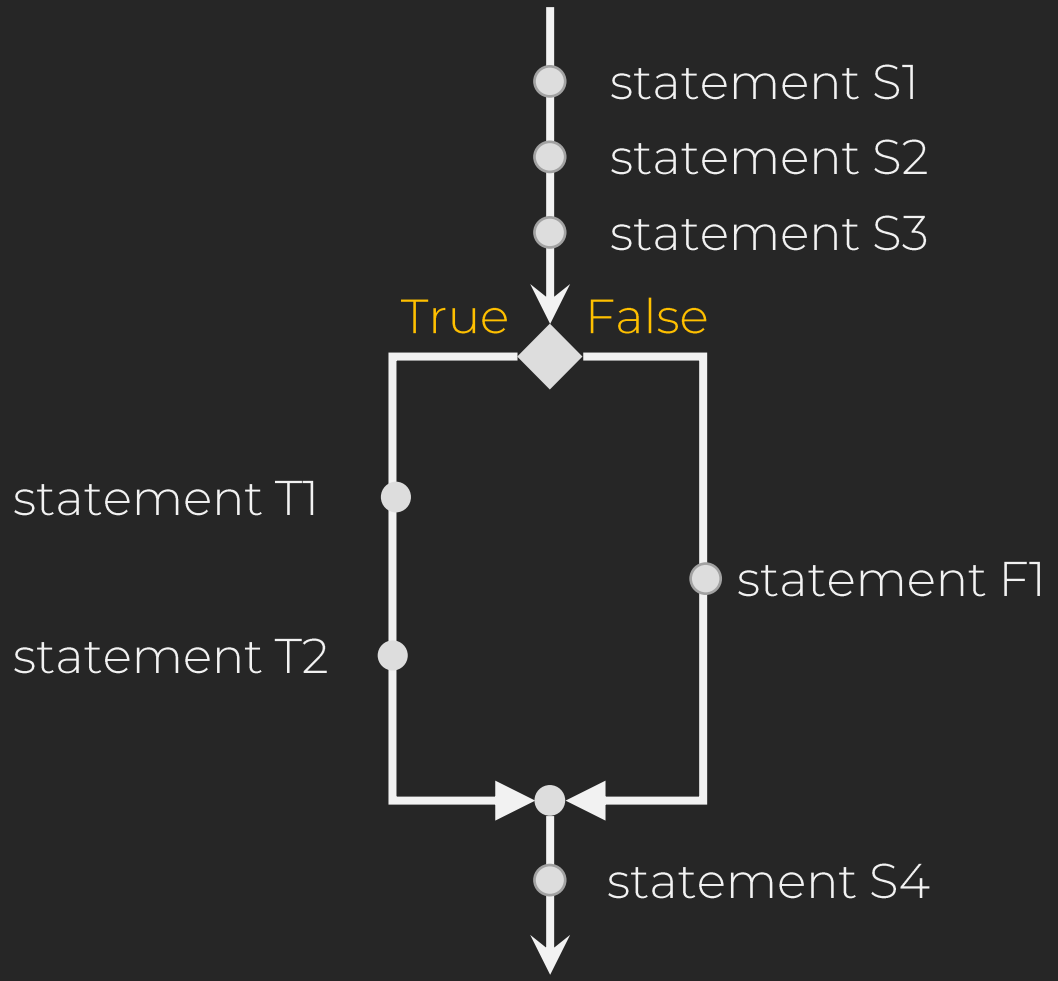

```
1  print('take phone')
2  print('take wallet')
3  print('take keys')
4  if weather == 'sunny' :
5      print('Take sunglasses')
6  elif weather == 'rainy' :
7      print('Take umbrella')
8
```

`if` vs `if-elif` statements

```
1  print('take phone')
2  print('take wallet')
3  print('take keys')
4  if weather == 'sunny' :
5      print('Take sunglasses')
6
7  if temperature == 'cold' :
8      print('Take coat')
9
```

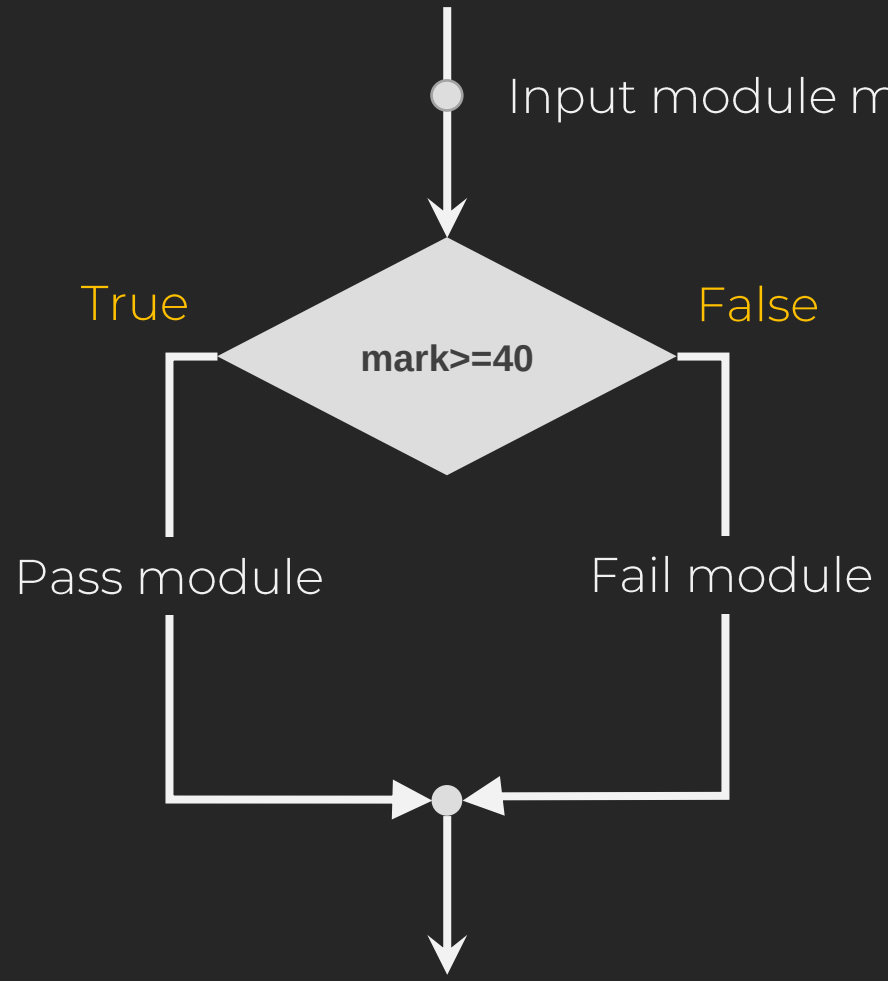
```
1  print('take phone')
2  print('take wallet')
3  print('take keys')
4  if weather == 'sunny' :
5      print('Take sunglasses')
6  elif temperature == 'cold' :
7    print('Take coat')
8
```

Catch all `if-else` statement



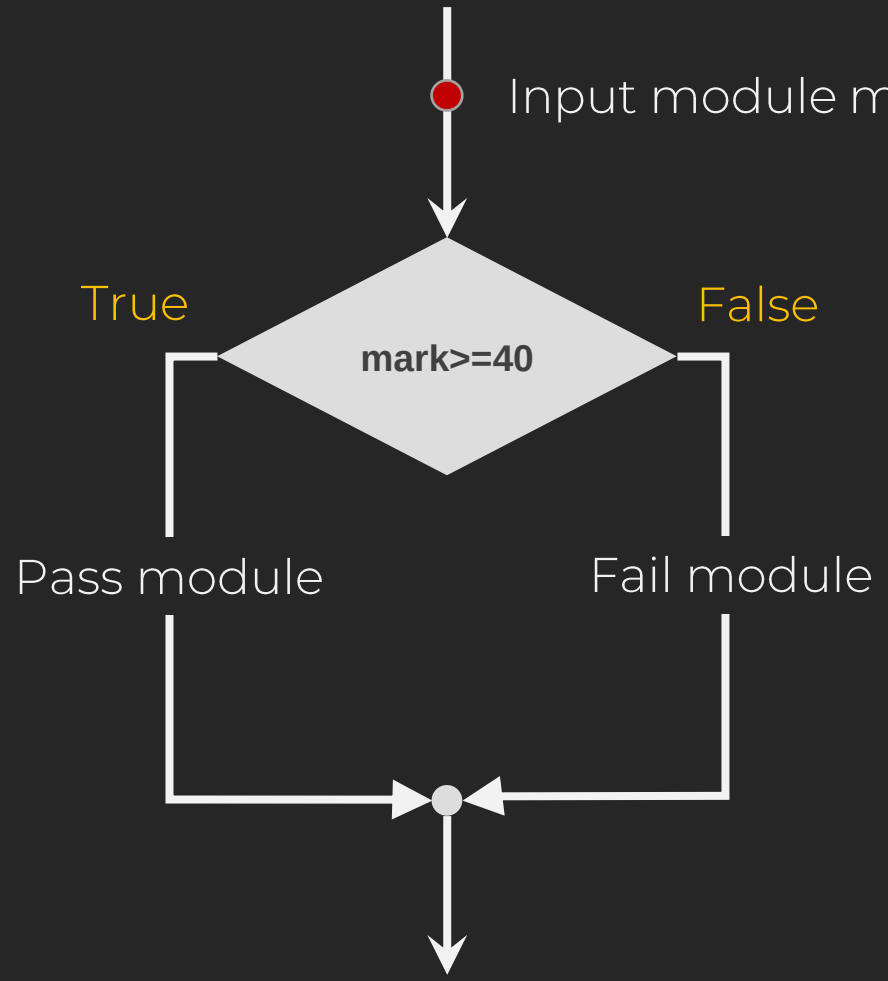

```
1   statement S1
2   statement S2
3   statement S3
4   if condition : #condition is True
5       statement T1
6       statement T2
7   else : #condition is False
8       statement F1
9
10  statement S4
```

Catch all `if-else` statement



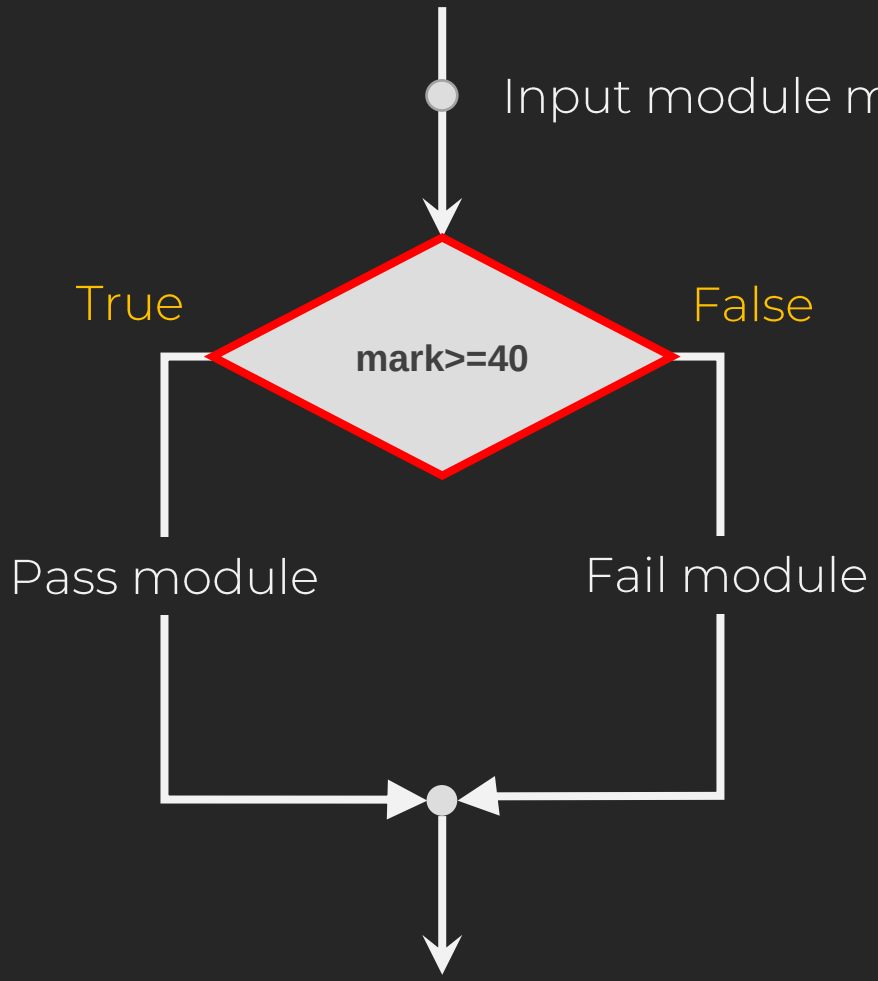

```
1  mark = int(input('Enter mark:'))
2  if mark >= 40 : #condition is True
3      print('Pass module.')
4  else : # means mark < 40
5      print('Fail module.')
6
```

Catch all `if-else` statement



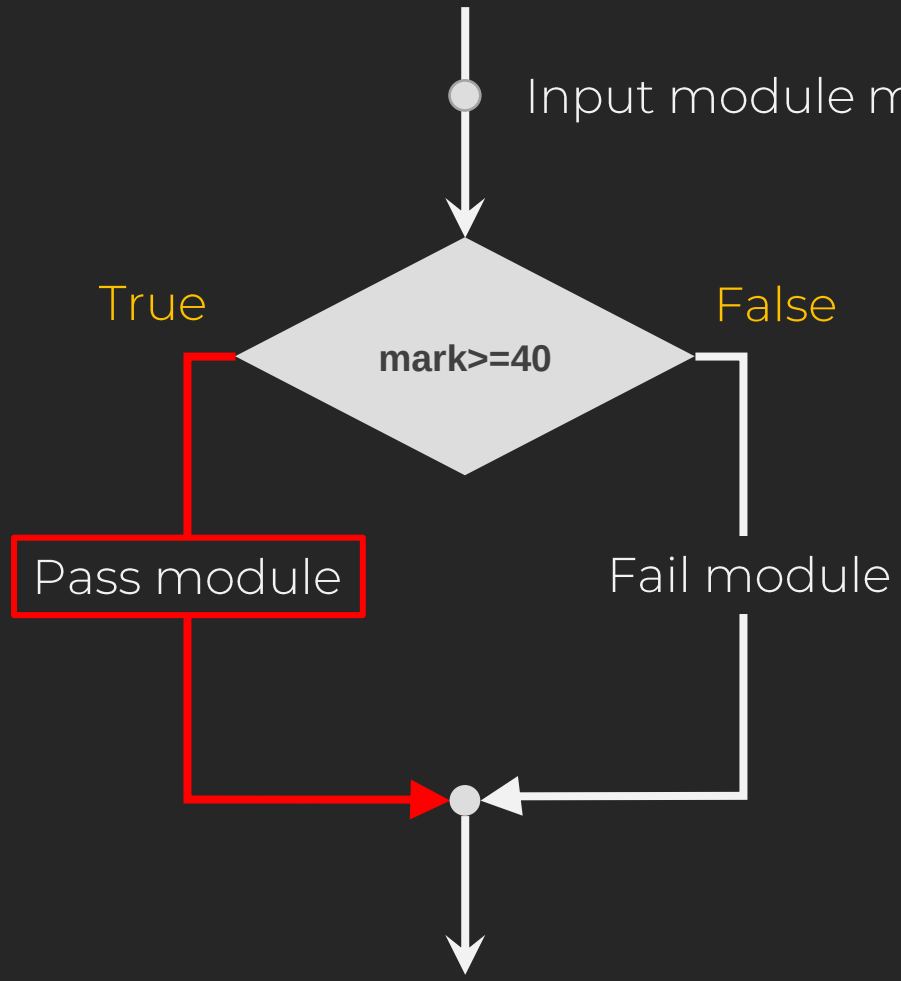

```
1  mark = int(input('Enter mark:'))
2  if mark >= 40 : #condition is True
3      print('Pass module.')
4  else : # means mark < 40
5      print('Fail module.')
6
```

Catch all `if-else` statement



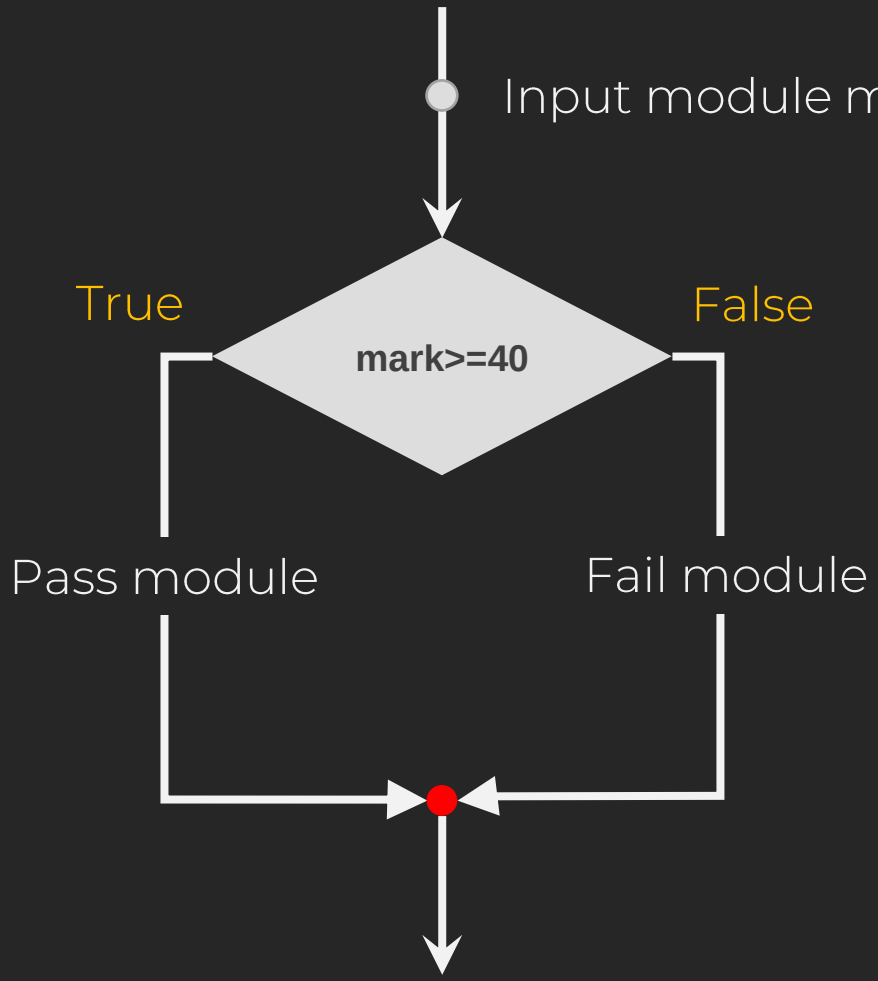

```
1  mark = int(input('Enter mark:'))
2  if mark >= 40 : #condition is True
3      print('Pass module.')
4  else : # means mark < 40
5      print('Fail module.')
6
```

Catch all `if-else` statement



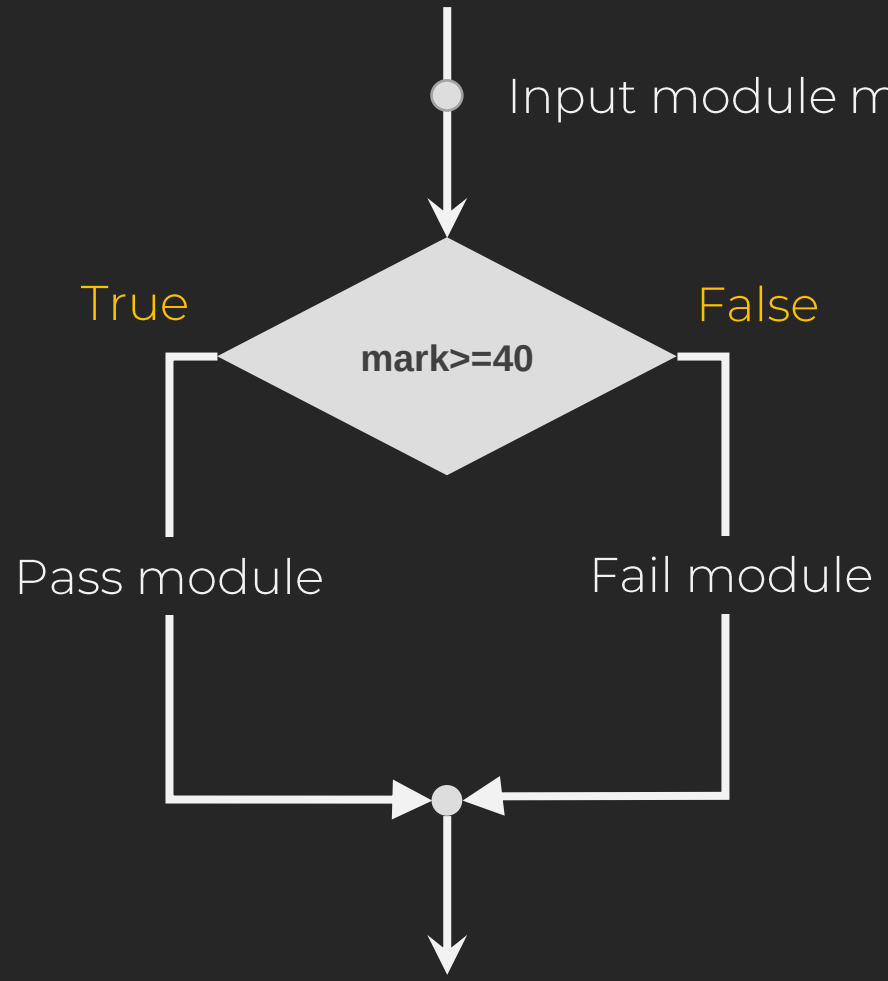

```
1  mark = int(input('Enter mark:'))
2  if mark >= 40 : #condition is True
3      print('Pass module.')
4  else : # means mark < 40
5      print('Fail module.')
6
```

Catch all `if-else` statement



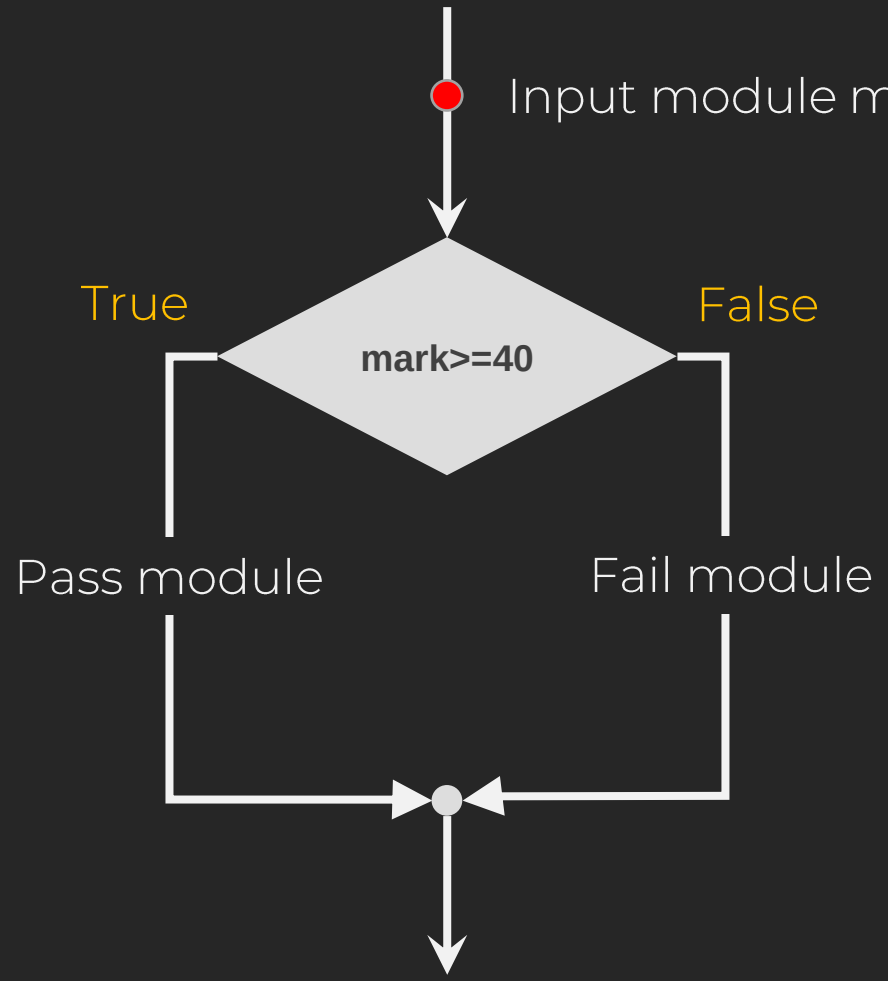

```
1  mark = int(input('Enter mark:'))
2  if mark >= 40 : #condition is True
3      print('Pass module.')
4  else : # means mark < 40
5      print('Fail module.')
6
```

Catch all `if-else` statement



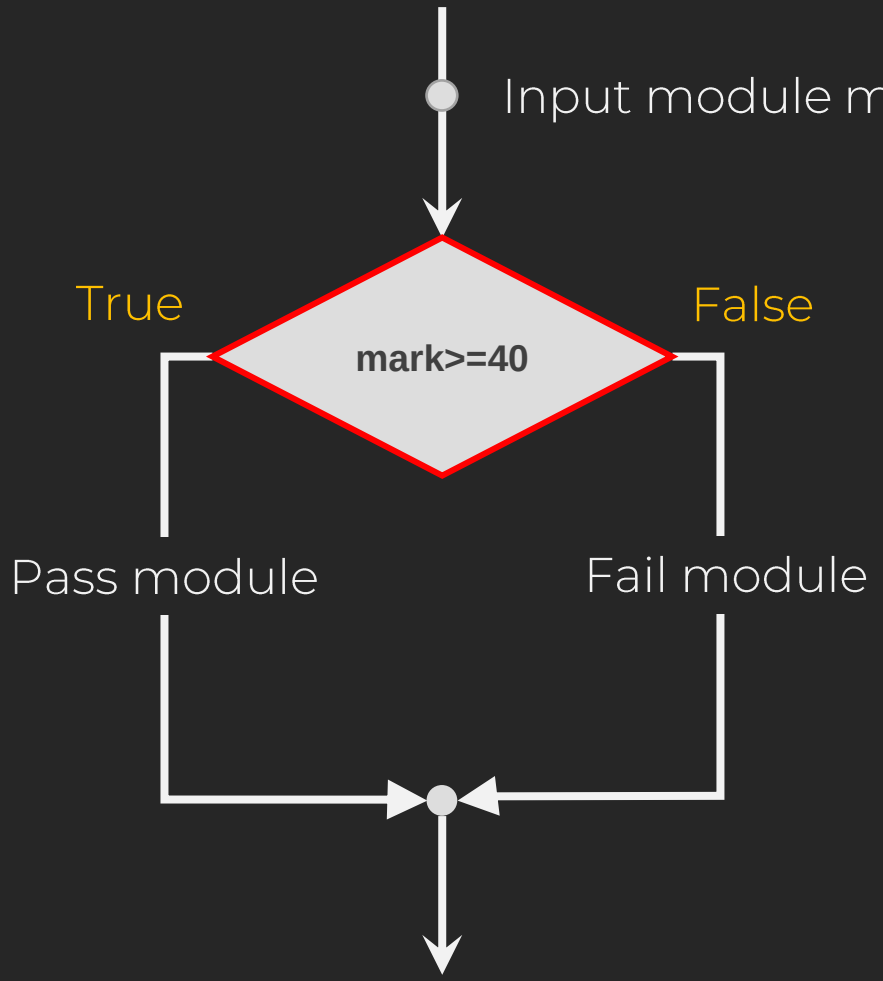

```
1  mark = int(input('Enter mark:'))
2  if mark >= 40 : #condition is True
3      print('Pass module.')
4  else : # means mark < 40
5      print('Fail module.')
6
```

Catch all `if-else` statement



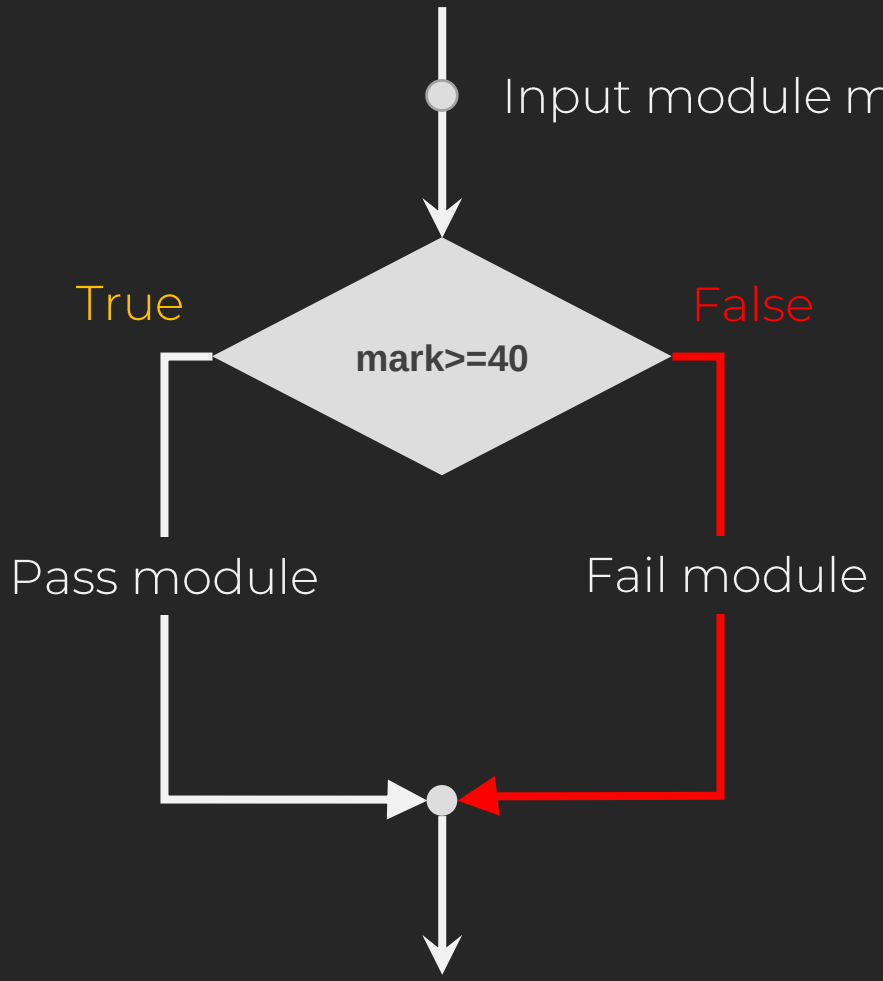

```
1  mark = int(input('Enter mark:'))
2  if mark >= 40 : #condition is True
3      print('Pass module.')
4  else : # means mark < 40
5      print('Fail module.')
6
```

Catch all `if-else` statement



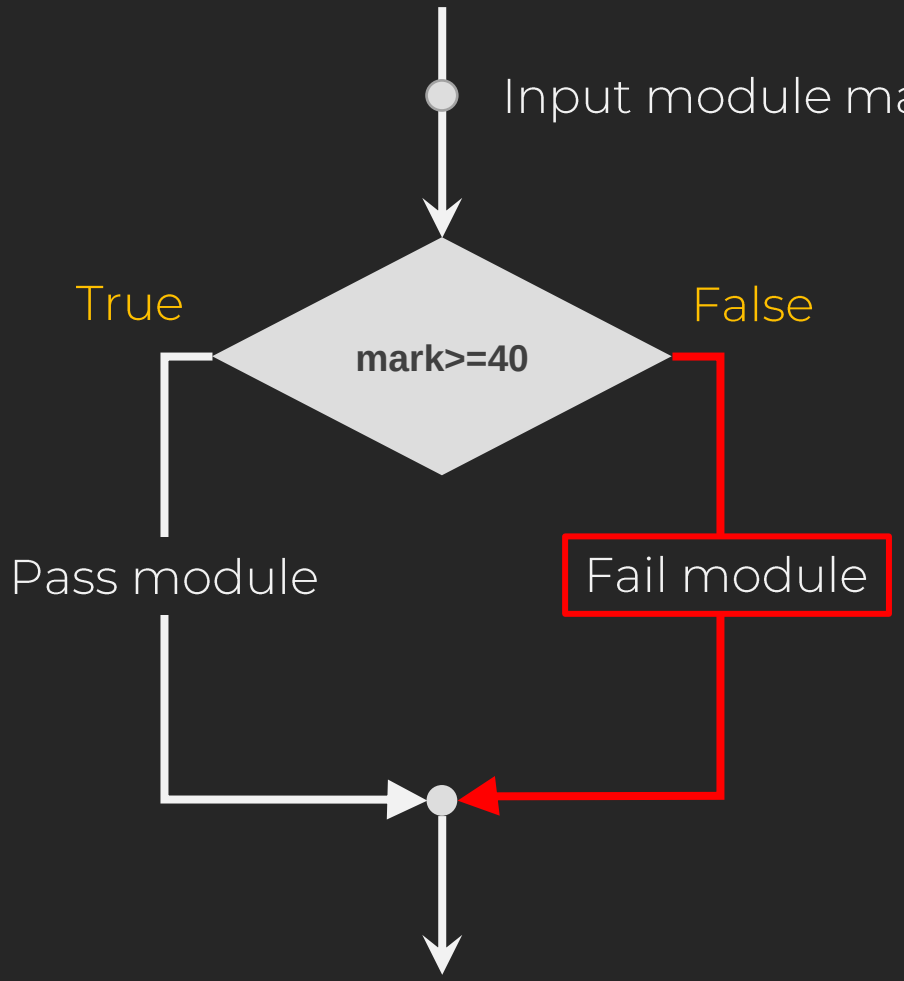

```
1  mark = int(input('Enter mark:'))
2  if mark >= 40 : #condition is True
3      print('Pass module.')
4  else : # means mark < 40
5      print('Fail module.')
6
```

Catch all `if-else` statement



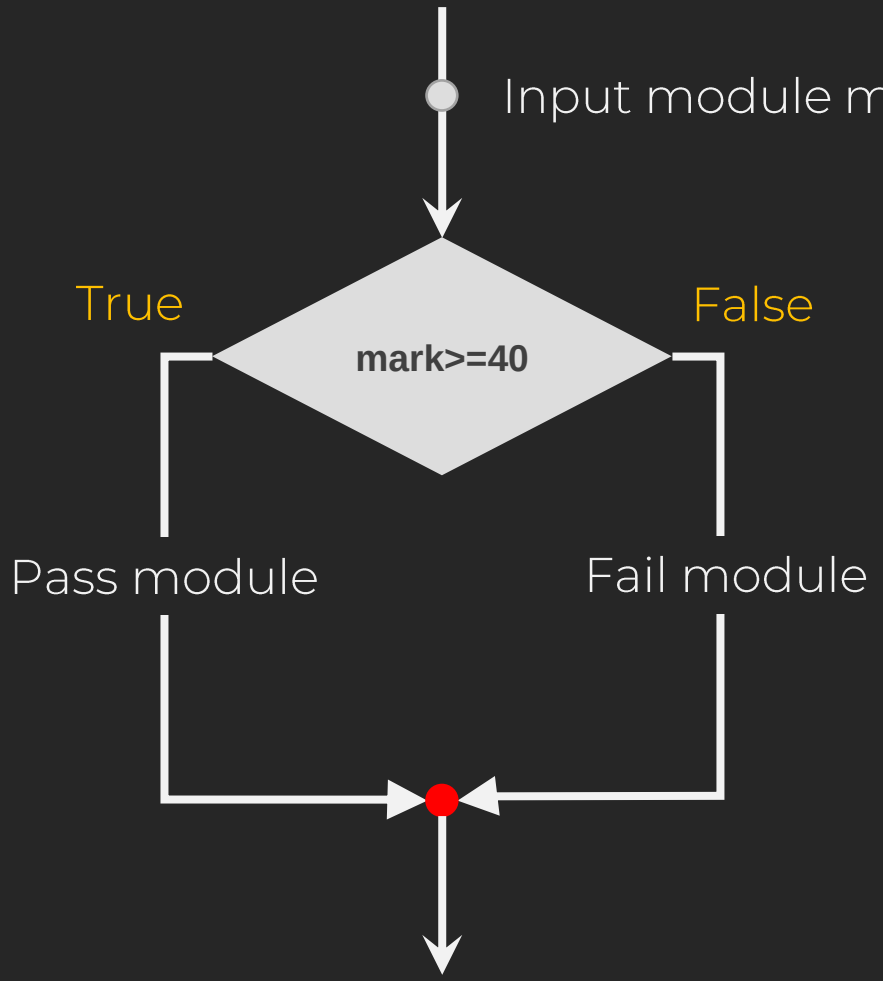

```
1   mark = int(input('Enter mark:'))
2   if mark >= 40 : #condition is True
3       print('Pass module.')
4   else : # means mark < 40
5       print('Fail module.')
6
```

Catch all `if-else` statement



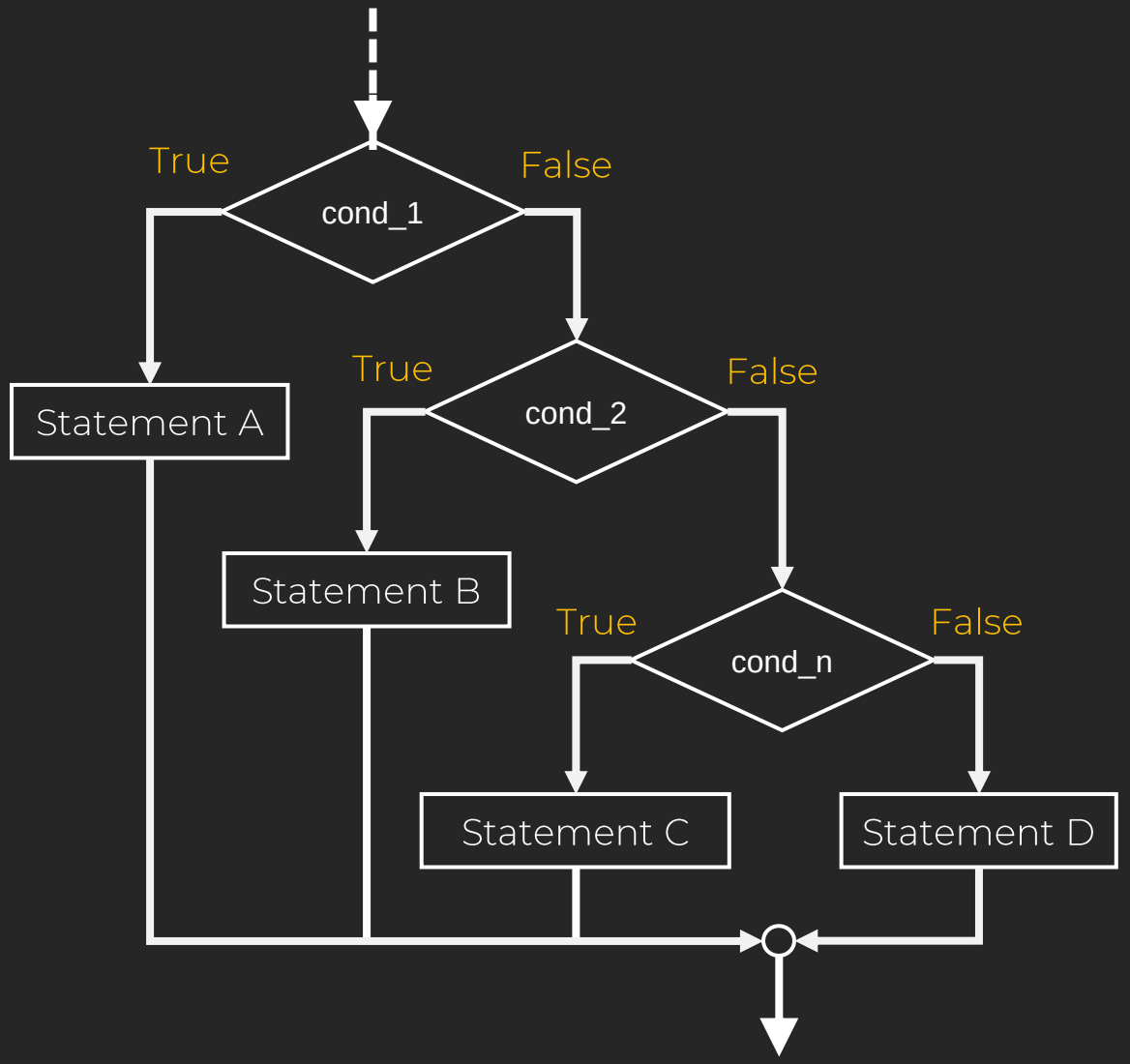

```
1  mark = int(input('Enter mark:'))
2  if mark >= 40 : #condition is True
3      print('Pass module.')
4  else : # means mark < 40
5      print('Fail module.')
6
```

Catch all `if-else` statement



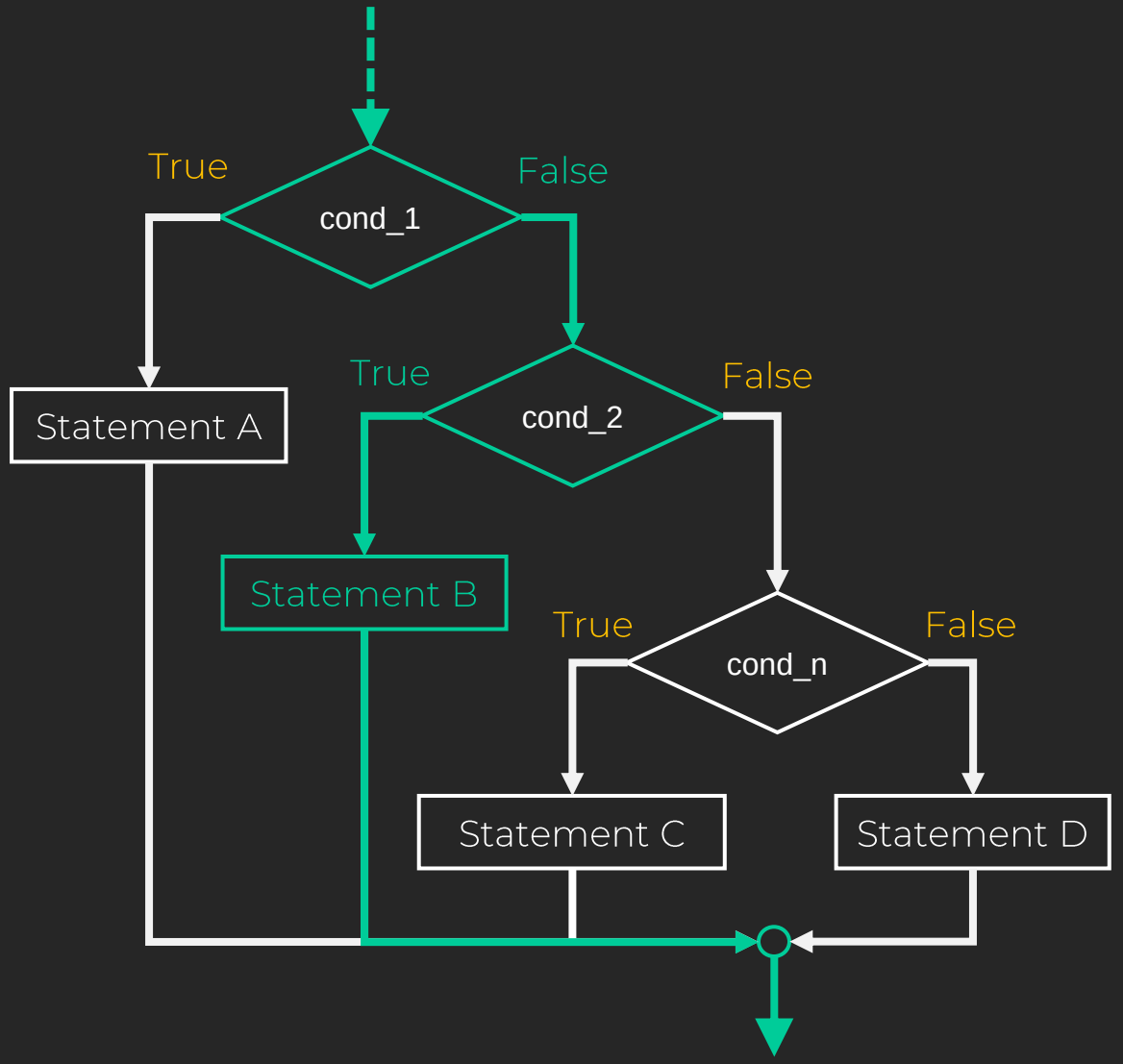

```
1  mark = int(input('Enter mark:'))
2  if mark >= 40 : #condition is True
3      print('Pass module.')
4  else : # means mark < 40
5      print('Fail module.')
6
```

`if-elif-else` statement



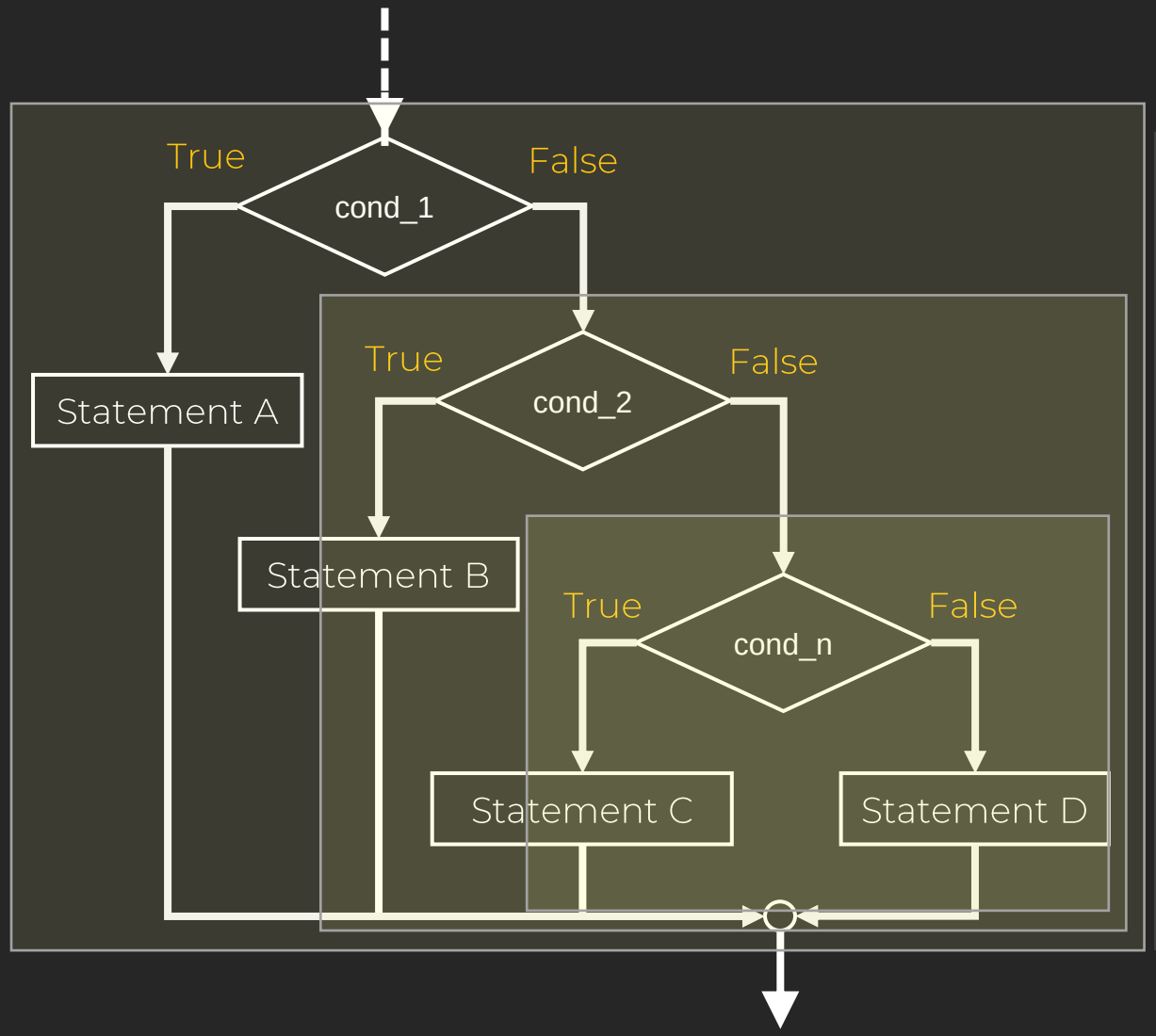

```
1    ...
2    if cond_1 :
3        statement A
4    elif cond_2 :
5        statement B
4    elif cond_n :
5        statement C
7    else : # all conditions are False
8        statement D
9    ...
```

`if-elif-else` statement




```
1    ...
2    if cond_1 :
3        statement A
4    elif cond_2 :
5        statement B
4    elif cond_n :
5        statement C
7    else : # all conditions are False
8        statement D
9    ...
```

Nested `if-else` statements

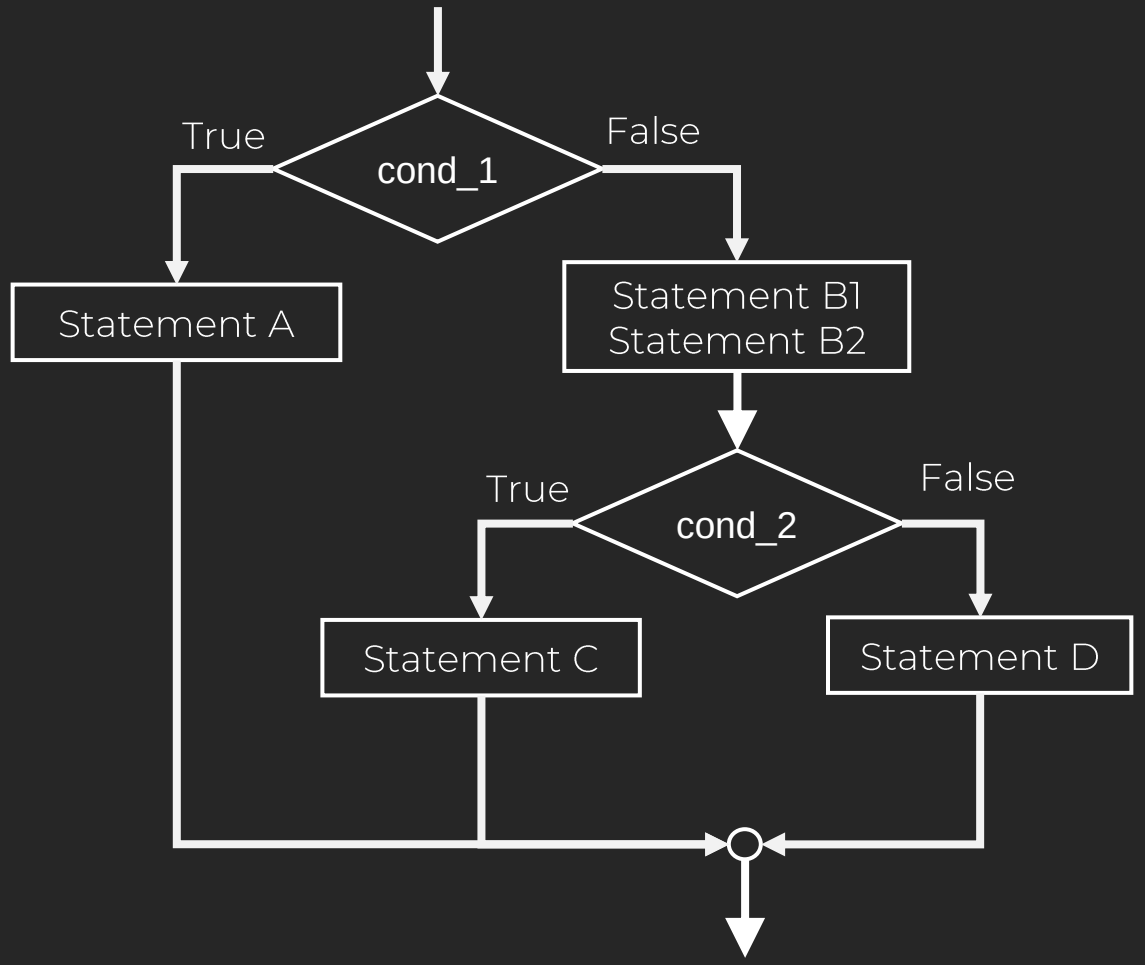



```
1    ...
2    if cond_1 :
3        statement A
4    else :
5        if cond_2 :
6            statement B
7        else :
8            if cond_n :
9                statement C
10           else :
11               statement D
12   ...
```

“There are many ways to code the same program.”

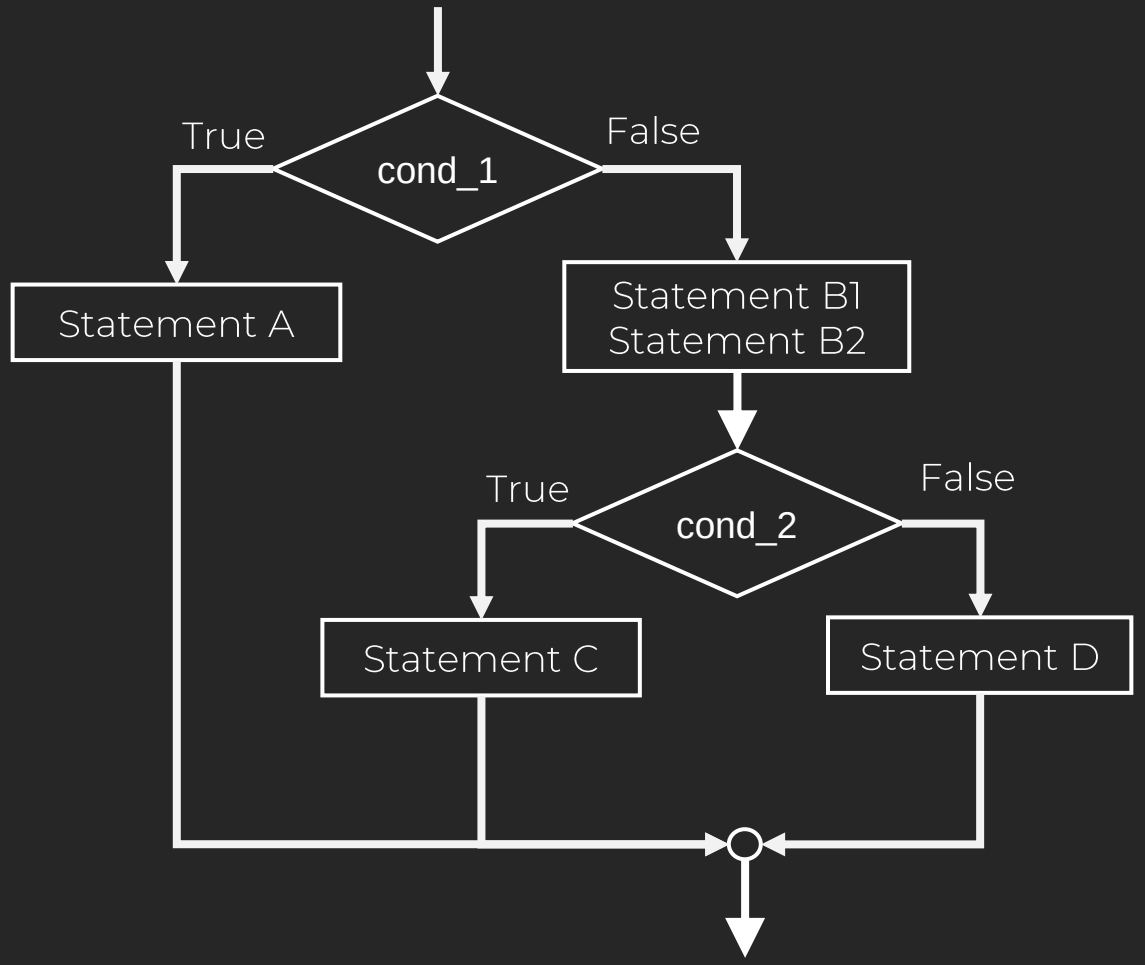
The challenge is to choose a good one.

if-elif-else vs nested if-else statements



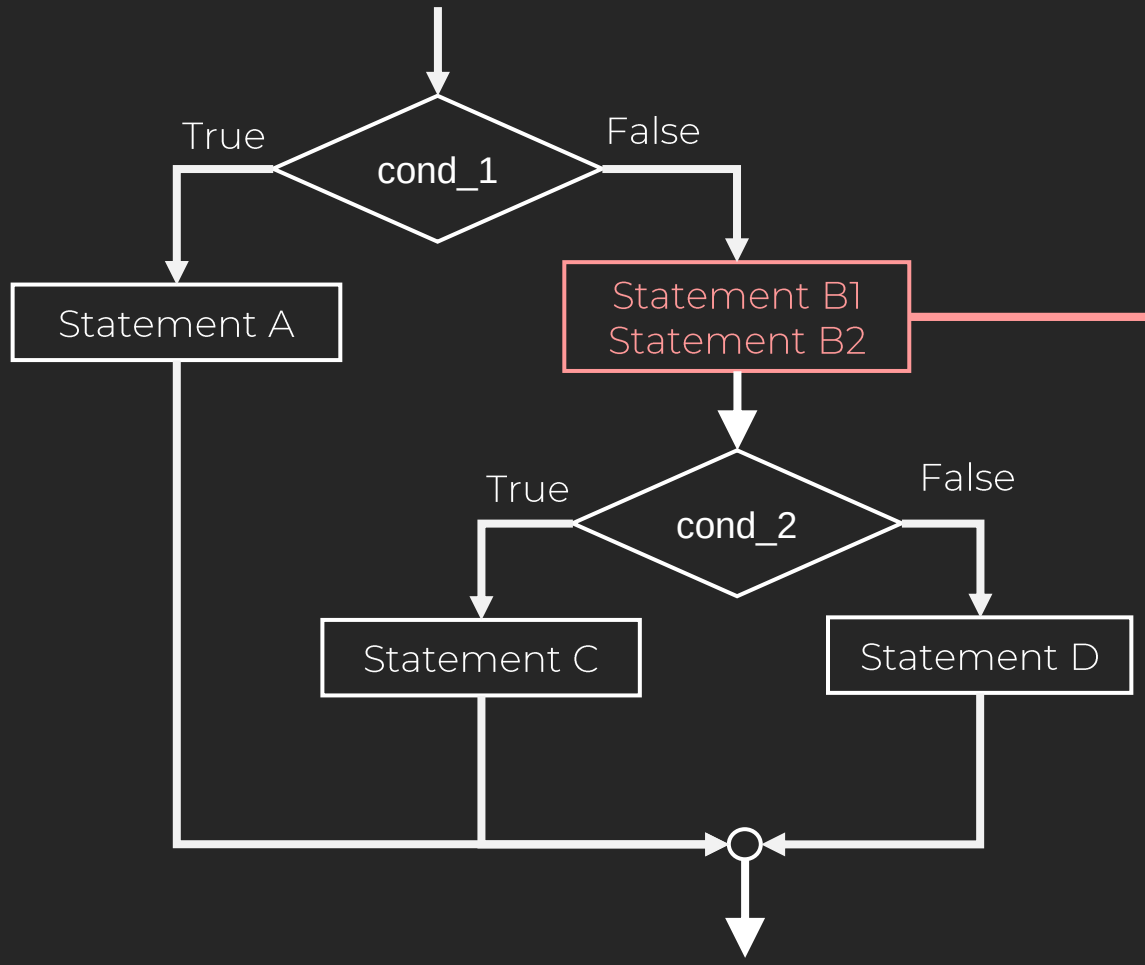

```
1    ...
2    if cond_1 :
3        statement A
4    else :
5        statement B1
6        statement B2
7        if cond_2 :
8            statement C
9        else :
10           statement D
11   ...
```

if-elif-else vs nested if-else statements




```
1    ...
2    if cond_1 :
3        statement A
4    elif cond_2 :
5        statement B1
6        statement B2
7        statement c
8    else :
9        statement B1
10       statement B2
11       statement D
12   ...
```

if-elif-else vs nested if-else statements




```
1    ...
2    if cond_1 :
3        statement A
4    elif cond_2 :
5        statement B1
6        statement B2
7        statement c
8    else :
9        statement B1
10       statement B2
11       statement D
12   ...
```

> You must avoid code duplication, that is writing exactly the same code multiple times.

You have seen how branching works, and how a program can execute some statements and skip others depending on certain conditions. This allows you to build more complex program.

You have seen how branching works, and how a program can execute some statements and skip others depending on certain conditions. This allows you to build more complex program.

In the next video, we will be looking at another flow-control structure, repetitions.