

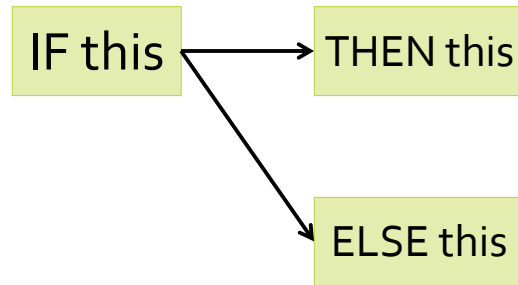
IF STATEMENTS

An Introduction to Computer Science



Let's learn about If Statements.

If Statements



If this is true, then do that.
Otherwise, don't do it.
That's the core idea of an IF statement.

Purpose

IF



THEN bring

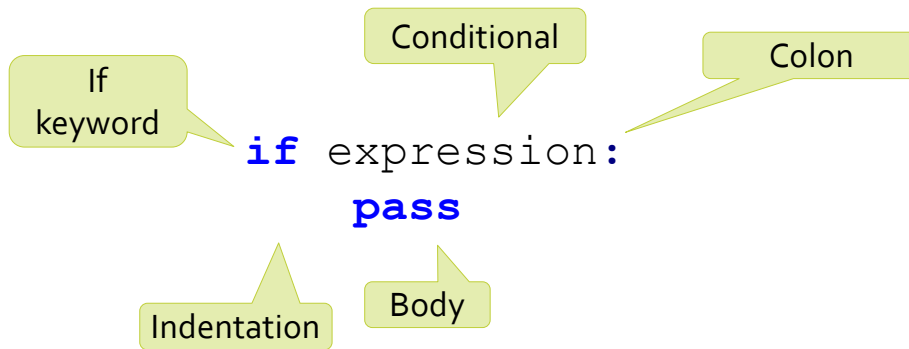


OTHERWISE do not.



If statements make code smarter.
They let you do different things in different situations.

Syntax



There are four parts to an IF statement.

First, you have the keyword "IF".

Then, you have the conditional.

The conditional is followed by a colon.

Finally, you have the body of the IF statement, indented with 4 characters.

Conditionals

```
if 4 < 5:
```

```
if age > 21:
```

```
if "cat" in animal_name:
```

```
if rent > 2000 and rooms < 2:
```

```
if not on_fire():
```



You already learned about writing conditionals, but as a reminder, conditionals are a way to ask questions.

Conditional expressions can be variables, comparisons, functions, combinations of these using AND, OR, and NOT.

Body

```
if income > 2000:  
    tax = .8  
    adjusted = income * tax  
    print("Income:", adjusted)
```

Body



The body of an IF statement is a sequence of one or more statements.
This body must be indented with 4 spaces.
Anything indented below the IF keyword is said to be "inside" the IF statement.
These indented statements will be executed if the conditional was evaluates to true.
Otherwise, Python will skip right over them.

The "Else" block

```
if precipitation > 0:
    print("It is raining")
else:
    print("It is NOT raining")
```

Else keyword

Colon

ELSE body

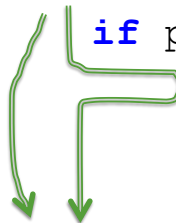
IF statements can optionally have an Else body.

If the conditional evaluates to false, then the ELSE will be executed instead of the IF body.

The IF body will literally be skipped over as if it didn't exist .

Only one of the two bodies will be executed!

Branching



```
if precipitation > 0:  
    print("It is raining")
```



Think about your program as a flowing stream.

Normally it will go from top to bottom.

An IF statement changes that flow, to optionally go around.

We call that behavior branching.

Every time you have another IF statement, you have another two branches.

Tracing Branches

```
1 age = 25
2 if age > 60:
3     discount = .8
4 else:
5     discount = 1
6 price = discount * 10
```

Step	Line	age	discount	price
1	1	25	X	X
2	2	25	X	X
3	5	25	1	X
4	6	25	1	10



We have previously shown how you can make a trace table that follows the execution of an IF statement.

We now have to add an extra column to this table, to differentiate between steps and lines.

With IF statements, it is possible that a step may skip a line of code.

Even though the program shown below has 6 lines of code, it only has 4 steps!

Notice that on lines with IF statements, we still have a step, but we simply repeat the values unchanged.