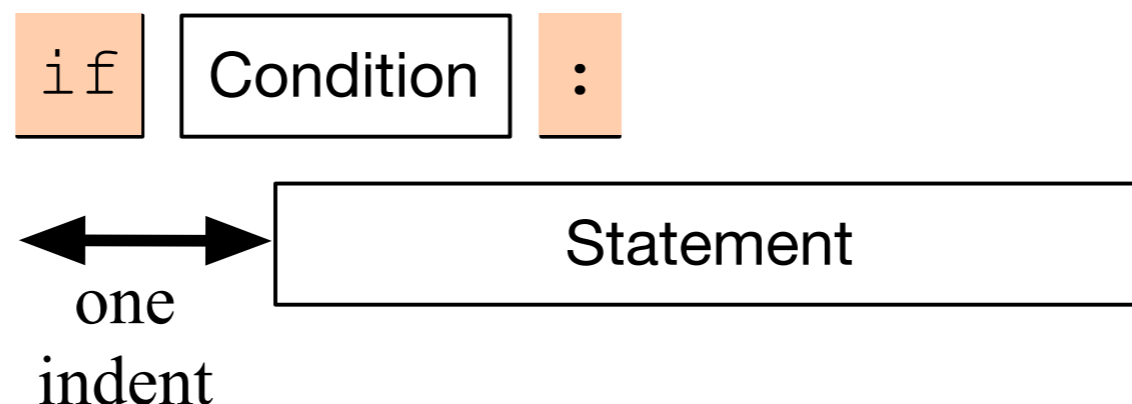


Conditional Statements

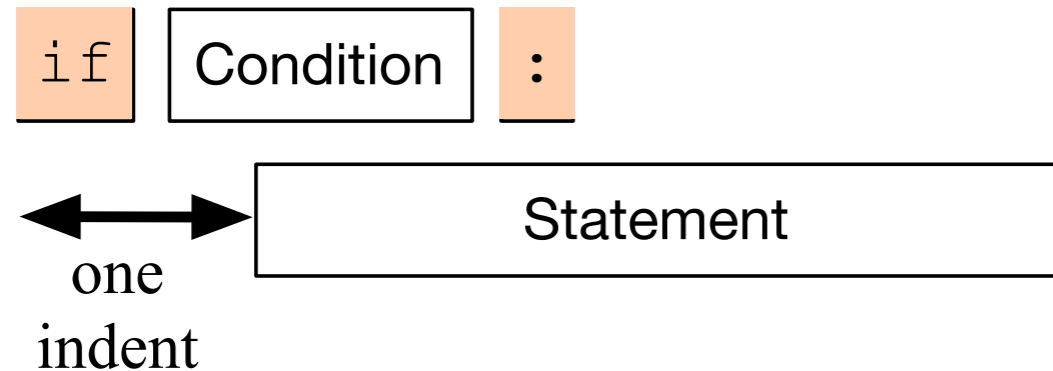
Python

Conditional Statements

- Sometimes a statement (or a block of statements) should only be executed if a condition is true.
- Conditional execution is implemented with the if-statement
- Form of the if-statement:



Conditional Statements



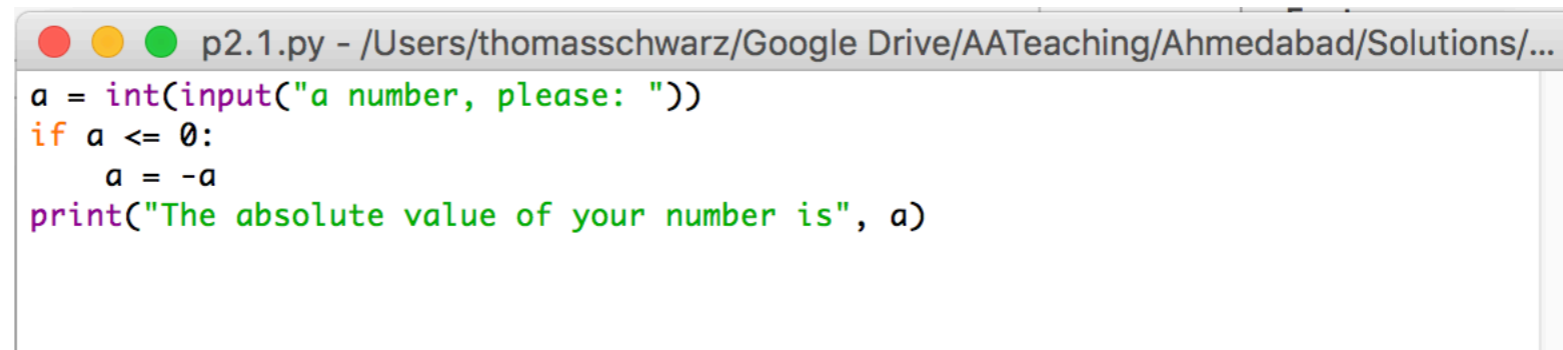
- `if` — is a keyword
- Condition: a Boolean, something that is either True or False
- Statement: a single or block of statements, all indented
 - Indents are tricky, you can use white spaces or tabs, but not both. Many editors convert tabs to white spaces
 - The number of positions for the indent is between 3 and 8, depending on the style that you are using. Most important, keep it consistent.

Example

```
● ● ● p2.1.py - /Users/thomasschwarz/Googl  
a = int(input("a number, please: "))  
if a < 5:  
    print("that is a small number.")
```

- First line asks user for integer input.
- Second line checks whether user input is smaller than 5.
- In this case only, the program comments on the number.

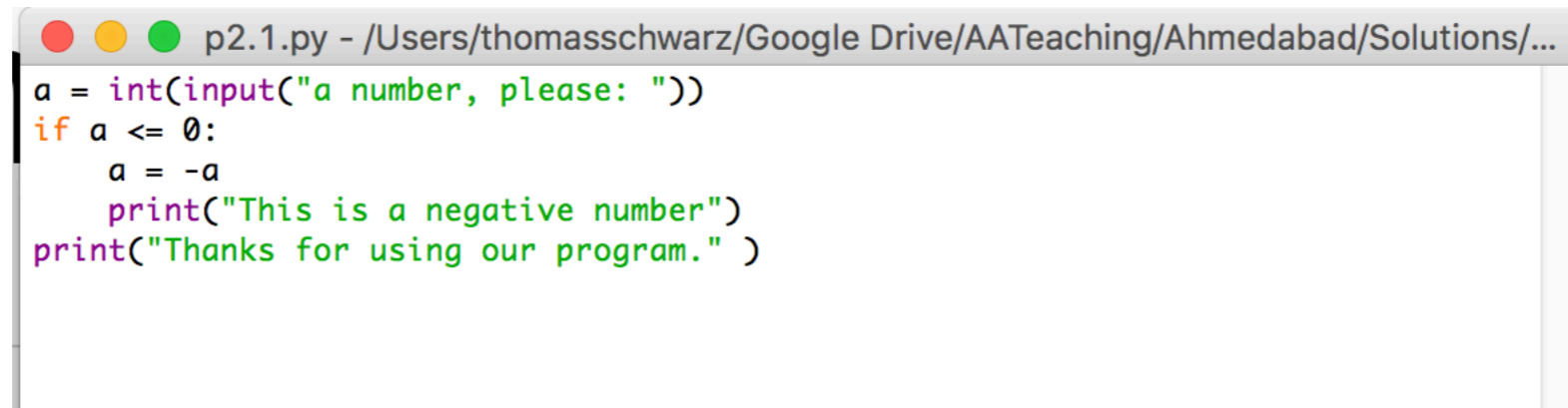
Example

A screenshot of a text editor window showing a Python script. The window title is "p2.1.py - /Users/thomasschwarz/Google Drive/AATeaching/Ahmedabad/Solutions/...". The code is as follows:

```
a = int(input("a number, please: "))
if a <= 0:
    a = -a
print("The absolute value of your number is", a)
```

- Here we calculate the absolute value of the input.
- The third line is indented.
- The fourth line is not, it is always executed.

Example



```
p2.1.py - /Users/thomasschwarz/Google Drive/AATeaching/Ahmedabad/Solutions/...  
a = int(input("a number, please: "))  
if a <= 0:  
    a = -a  
    print("This is a negative number")  
print("Thanks for using our program." )
```

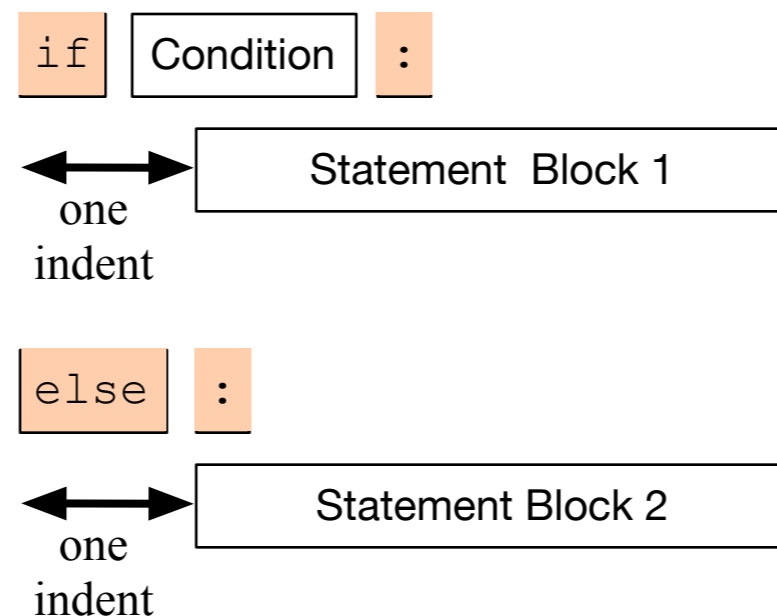
- Here, lines 3 and 4 are indented and are executed if the input is a negative integer.
- The last line, line 5, is always executed since it is not part of the if-statement

Alternative statements

- Very often, we use a condition to decide which one of several branches of execution to pursue.
- The else-statement after the indented block of an if-statement creates an alternative route through the program.

Alternative Statements

- The if-else statement has the following form:



- We add the keyword `else`, followed by a colon
- Then add a second set of statements, indented once
- If the condition is true, then Block 1 is executed, otherwise, Block 2.

Examples

- I can test equality by using the double = sign.
- To check whether a number n is even, I take the remainder modulo 2 and then compare with 0.

```
p2.2.py - /Users/thomasschwarz/Google Drive/AATeaching/Ahmedabad/Solutions/...
number = int(input("Enter a number: "))
if number%2 ==0:
    print("The number is even.")
    print("Its square is", number**2)
else:
    print("The number is odd.")
    print("Its square-root is", number**0.5)
|
```

Alternative Statements

- Often, we have more than two alternative streams of execution.
- Instead of nesting if expressions, we can just use the keyword “elif”, a contraction of else if.

Alternative Statements

`if` `Condition 1` `:`

↔ `Statement Block 1`
one indent

`elif` `Condition 2` `:`

↔ `Statement Block 2`
one indent

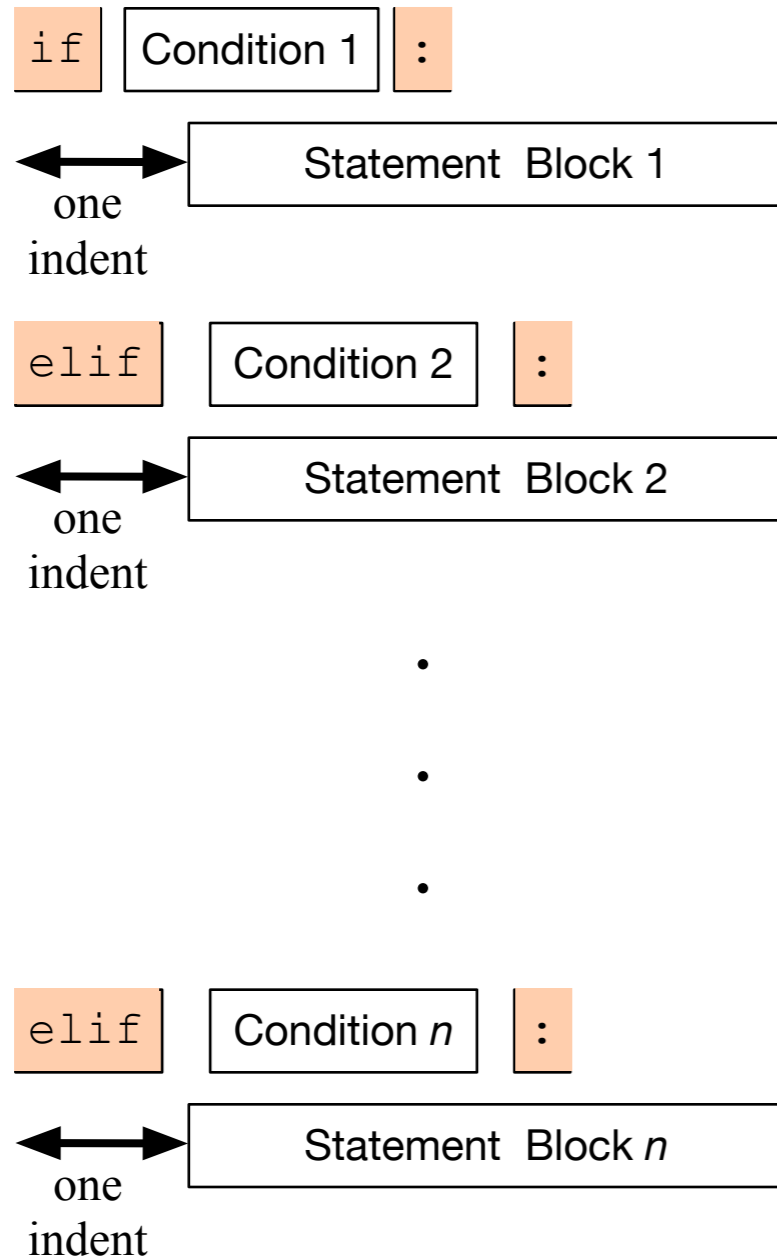
.
. .
.

`else` `:`

↔ `Statement Block n`
one indent

- One of the statement blocks is going to be executed
- The else block contains the default action, if none of the conditions are true

Alternative Statements



- Here, there is no else statement, so it is possible that none of the blocks is executed.

Examples

- Categorization of temperatures

```
if temperature < -25.0:
    feeling = "arctic"
elif temperature < -10.0:
    feeling = "Wisconsin in winter"
elif temperature < 0.0:
    feeling = "freezing"
elif temperature < 15.0:
    feeling = "cold"
elif temperature < 25.0:
    feeling = "comfortable"
elif temperature < 35.0:
    feeling = "hot"
elif temperature < 45.0:
    feeling = "Ahmedabad in the summer"
else:
    feeling = "hot as in hell"
```

Boolean Expressions

- Nested loops to implement decision tree:

```
if x<10:  
    if y<3:  
        if x<2:  
            result=0  
        else:  
            result=1  
    else:  
        result=0  
else:  
    if y<2:  
        result=1  
    else result=0
```

