CSE 115
Introduction to Computer Science I
Road map

▶ Review ◀

miscellaneous items

relational expressions

Boolean expressions
Defining and calling functions
Defining functions

Here's an example of a function definition:

```python
def averageOfThree(x, y, z):
    average = (x + y + z) / 3
    return average
```
Calling functions

Another example: the value of

averageOfThree( 5, 2, 8 )
Another example: the value of

\[
\text{averageOfThree}(5, 2, 8)
\]

def averageOfThree( x, y, z ):
    average = (x + y + z) / 3
    return average

<table>
<thead>
<tr>
<th>name</th>
<th>value</th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td>5</td>
</tr>
<tr>
<td>y</td>
<td>2</td>
</tr>
<tr>
<td>z</td>
<td>8</td>
</tr>
</tbody>
</table>
Calling functions

Another example: the value of

\[
\text{averageOfThree}(5, 2, 8)
\]

def averageOfThree(x, y, z):
    average = (x + y + z) / 3
    return average

\[
(x + y + z) / 3 = (5 + 2 + 8) / 3 = 15 / 3 = 5.0
\]
Calling functions

Another example: the value of

averageOfThree( 5, 2, 8 )

def averageOfThree( x, y, z ):
    average = (x + y + z) / 3
    return average

<table>
<thead>
<tr>
<th>name</th>
<th>value</th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td>5</td>
</tr>
<tr>
<td>y</td>
<td>2</td>
</tr>
<tr>
<td>z</td>
<td>8</td>
</tr>
<tr>
<td>average</td>
<td>5.0</td>
</tr>
</tbody>
</table>

The value returned is 5.0
Another example: the value of

\texttt{averageOfThree( 5, 2, 8 )}

is 5.0
Road map

Review

▶ miscellaneous items ◀

relational expressions

Boolean expressions
Imports

Some functions are built in and directly accessible.

More are available in other modules.

To gain access, import the module.
Example

```python
import math

print( math.sin( math.pi/2 ) )
```
Road map

Review

miscellaneous items

▶ relational expressions ▶

Boolean expressions
## Relational operators

[https://docs.python.org/3.7/library/stdtypes.html#comparisons](https://docs.python.org/3.7/library/stdtypes.html#comparisons)

<table>
<thead>
<tr>
<th>Operation</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;</code></td>
<td>strictly less than</td>
</tr>
<tr>
<td><code>&lt;=</code></td>
<td>less than or equal</td>
</tr>
<tr>
<td><code>&gt;</code></td>
<td>strictly greater than</td>
</tr>
<tr>
<td><code>&gt;=</code></td>
<td>greater than or equal</td>
</tr>
<tr>
<td><code>==</code></td>
<td>equal</td>
</tr>
<tr>
<td><code>!=</code></td>
<td>not equal</td>
</tr>
</tbody>
</table>
Relational expressions

general form

expression  relational-operator  expression
Relational expressions

examples

3 < 5
Relational expressions

examples

3 < 5

x >= y
Relational expressions

examples

\[ 3 < 5 \]

\[ x \geq y \]

\[ 2*(x-y) \neq y/(3*z) \]
Relational expressions

Boolean value

The value of a relational expression is always a Boolean value: either True or False.
Road map

Review

relational expressions

▶ Boolean expressions ▼
### Boolean operators

[https://docs.python.org/3.7/library/stdtypes.html#boolean-operations-and-or-not](https://docs.python.org/3.7/library/stdtypes.html#boolean-operations-and-or-not)

<table>
<thead>
<tr>
<th>Operation</th>
<th>Result</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>x or y</code></td>
<td>if <code>x</code> is false, then <code>y</code>, else <code>x</code></td>
<td>(1)</td>
</tr>
<tr>
<td><code>x and y</code></td>
<td>if <code>x</code> is false, then <code>x</code>, else <code>y</code></td>
<td>(2)</td>
</tr>
<tr>
<td><code>not x</code></td>
<td>if <code>x</code> is false, then <code>True</code>, else <code>False</code></td>
<td>(3)</td>
</tr>
</tbody>
</table>

**Notes:**

1. This is a short-circuit operator, so it only evaluates the second argument if the first one is false.
2. This is a short-circuit operator, so it only evaluates the second argument if the first one is true.
3. `not` has a lower priority than non-Boolean operators, so `not a == b` is interpreted as `not (a == b)`, and `a == not b` is a syntax error.
Boolean expressions

examples

True  or  False

a  and  b

x < y  and  y <= z
Boolean expressions
examples

True or False

a and b

x < y and y <= z

x < y <= z

Convenient, but unusual across languages.