

Variables and Expressions



CHAPTER 2

Statements and Expressions



- A **statement** is like a sentence in python.
 - Generally, one line of code
 - Made up of **expressions**.
- Expression are made up of 1 or more “building blocks”
 - These building blocks are themselves expressions.

```
print("The answer is: " + str(round(x * 5, 1)) )
```

The code line is enclosed in a blue dashed box. Within it, the string "The answer is: " is enclosed in a yellow dashed box. The str() function call is enclosed in a green dashed box. The round() function call is enclosed in a red dashed box. The expression x * 5 is enclosed in a purple dashed box.

“Building Blocks” of expressions



- I. Constants
- II. Operators
- III. Variables.
- IV. Function definitions and calls.
- V. Classes and Object definitions and calls.
- VI. Sequences

I.A: Numbers constants



- **Integer constants:**
 - A “hard-coded” number without a decimal point.
 - Internal representation
- **Floating-point constants (float's)**
 - A number with a decimal point.
 - Approximations to some numbers
 - ✦ Handling this



I.B: String constants

- glyphs **delimited** by quotes
- 3 Types
- Escape Sequences



II: Operators



- **binary operators and operands.**

- The first batch:

+

-

*

/

//

%

**

- **Precedence** and parentheses
- **Type** of result

III: Variables



- A **variable** is a name for a (group of) memory location(s).
 - A "storage box" containing a value
- Python uses **implicit declarations**.
- Used in place of a constant
- = operator and augmented assignment operators

Variables as "storage"



- Think of a variable as holding a value.
 - A variable is just a reference to one section of memory
 - Those memory contents are set to (the number) we indicate in our assignment.
- That value stays fixed until we change it.
- [Example]: hello _____

IV: Functions



- A collection of (python) statements that accomplish some task.
 - E.g. print: displays text on the screen.
- Two parts:
 - definition (Ch. 6)
 - call

```
print("The answer is:", x * 5, sep="\t")
```

*parentheses,
(needed even if there
are no arguments)*

0 or more arguments, separated by commas

IV: Functions to remember



- print
- input
- round
- min and max
- *conversion functions*
 - *int*
 - *str*
 - *float*

Example



- Ask the user for their name and age.
- Print out their name and age in seconds (with explanatory text)

V. Objects



- We'll look at this in much more detail later.
- But an object is basically
 - a container for other variables
 - a set of associated functions that only work with this container.
- [A few methods of built-in types (strings and int's)]

Another Day...



- **Sequences (Ch. 4 & 5)**
 - Collections of related data (accessed by index)
- **Functions (Ch. 6)**
 - writing / using our own functions
 - using built-in "libraries" (random, math, etc.)
 - using / creating add-on "libraries" (e.g. pygame)
- **Classes and Objects (Ch. 8 & 9)**
 - Ch. 8 & 9
 - Creating your own classes is a great way to approach problems