

Introduction to Python Programming

2



Objectives

In this chapter, you'll:

- Continue using IPython interactive mode to enter code snippets and see their results immediately.
- Write simple Python statements and scripts.
- Create variables to store data for later use.
- Become familiar with built-in data types.
- Use arithmetic operators and comparison operators, and understand their precedence.
- Use single-, double- and triple-quoted strings.
- Use built-in function `print` to display text.
- Use built-in function `input` to prompt the user to enter data at the keyboard and get that data for use in the program.
- Convert text to integer values with built-in function `int`.
- Use comparison operators and the `if` statement to decide whether to execute a statement or group of statements.
- Learn about objects and Python's dynamic typing.
- Use built-in function `type` to get an object's type.

2 Introduction to Python Programming

Note: Throughout the Instructor Solutions Manual, solutions are *not provided* for project, research and challenge exercises—many of which are substantial and appropriate for term projects, directed-study projects, capstone-course projects and thesis topics. **Before assigning a particular exercise for homework, instructors should check the IRC to be sure the solution is available.** These Instructor Solutions Manual PDFs contain **only answers to short-answer exercises and any discussion questions asked in other exercises.** Code corresponding to programming exercises can be found in the **solutions** folder's chapter-specific subfolder—e.g., **ch01** for Chapter 1, **ch02** for Chapter 2, etc. Code generally is provided both in **Python source-code files (.py)** and **Jupyter Notebooks (.ipynb)**.

Exercises

Unless specified otherwise, use IPython sessions for each exercise.

2.1 (*What does this code do?*) Create the variables $x = 2$ and $y = 3$, then determine what each of the following statements displays:

a) `print('x =', x)`

Answer: `x = 2`

b) `print('Value of', x, '+', x, 'is', (x + x))`

Answer: `Value of 2 + 2 is 4`

c) `print('x =')`

Answer: `x =`

d) `print((x + y), '=', (y + x))`

Answer: `5 = 5`

2.2 (*What's wrong with this code?*) The following code should read an integer into the variable `rating`:

```
rating = input('Enter an integer rating between 1 and 10')
```

Answer: Function `input` returns a string, so to get an integer, you'd have to use:

```
rating = int(input('Enter an integer rating between 1 and 10'))
```

2.3 (*Fill in the missing code*) Replace `***` in the following code with a statement that will print a message like 'Congratulations! Your grade of 91 earns you an A in this course'. Your statement should print the value stored in the variable `grade`:

```
if grade >= 90:
    ***
```

Answer:

```
if grade >= 90:
    print('Congratulations! Your grade of', grade,
          'earns you an A in this course')
```