Chapter 2 Notes

Basic Python Definitions:

A Python program (or Python **script**) is a sequence of definitions and commands. These definitions are evaluated, and the commands are executed by the Python interpreter in something called a **shell**. A **command**, often called a **statement**, instructs the interpreter to do something

Objects

**Objects** are the core things that Python programs manipulate. Every object has a **type** that defines the kinds of things that programs can do with that object.

* Types are either scalar or non-scalar.
* **Scalar** **objects** are indivisible.
* **Non**-**scalar** **objects**, however, have internal structure.

Python has four types of scalar objects:

* **Int** is used to represent integers.
* **Float** is used to represent real numbers.
* **Bool** is used to represent the Boolean values True and False
* **None** is a type with a single value.

Objects and operators can be combined to form expressions, each of which evaluates to an object of some type. This is the value of that expression.

**The == operator** is used to test whether two expressions evaluate to the same value, and the != operator is used to test whether two expressions evaluate to different values.

**The symbol >>>** is a shell prompt indicating that the interpreter is expecting the user to type some Python code into the shell.

The build-in function **type** can be used to find out the type of an object. **type(<object>)**

Basic Operators

There are six basics operators on objects of type int and float: addition (**+**), subtraction (-), multiplication (\*), division (/), integer division (//)(which returns the quotient and ignores the remainder), remainder (%), power (\*\*). Arithmetic operators have the usual precedence.

There are three primitive operators on type bool: **and**, **or** and **not**

Variables provide a way to associate names with objects. The variable names will bind with different objects with a different type. In Python, a variable is just a name, and nothing more.

**An assignment statement (with =)** associates the name to the left of the = symbol with the object denoted by the expression to the right of the =. An object can have one, more than one, or no name associated with it. Python allows multiple assignment.

Variable Names

In Python, variable names can contain upper case and lower-case letters, digits (but they cannot start with a digit), and the special character \_. Python variable names are case sensitive. There are keywords that cannot be used as name, listed as follows:

* and, assert, break, class, continue, def, del, elif, else, except, False, finally, for, from, global, if, import, in, is, lambda, nonlocal, None, not, or, pass, raise, return, True, try, while, with, and yield.

Comments

A good way to enhance the readability of code is to add comments, which are text following the symbol # is not interpret by Python.