PYTHON

SECTION THREE

Introduction to Python| Numerical Data | Pragmatic Agility

LITERAL INTEGERS

- Numbers typed directly into code
- Can not be prefixed with a 0
- Can not use commas
- Supports standard integer operations. e.g. +, -, *, etc.

BASES

- Binary
- Octal
- Base 10
- Hex

Python supports multiple bases. You may never have a reason to use anything other then base 10.

FLOATS & BOOL

- Decimal numbers
    - 2.25
- Booleans
    - True/False
    - 1/0
    - exist/not exist

KEY CONCEPTS

The underscore '_'

The underscore can be used to make numbers more readable. For example, 2_000.00

Booleans

Booleans are represented by the word True or False. However, some implementations will use a 0 for false and any number 1 or greater as true.

Math Functions

Functions are blocks of code hidden behind a name or label. For example, we can import the math library and call math.factorial against a number to get, well, the factorial of a number

Immutable

Numerical data is immutable. Changes to data cause new data objects to be made

Equality

a = x + 5 is not a test for equality as in Algebra. In this example a is having its value set to 5 + the value of variable x. Keep in mind this causes 'a' to generate a new data object in the background.

Type Conversions

Use the int() function to convert a data object to a integer. It has to be a value that can be interpreted as an integer. '2' vs two