Week zero

Help

Comments — CodeSkulptor

- Non-computational parts of the program that textually describe the behavior of the program.
- Comments begin with `#`, everything to right of the hash is ignored by Python.
- Comments should be frequent so you and others can understand the code.
- Lecture examples - CodeSkulptor
- More examples - Comments, Strings, and Print

Strings — CodeSkulptor

- Sequence of characters enclosed by a pair of single or double quotes
- Examples are `"cats hate dogs"` and `'Strings are fun!'`.
- Strings are one kind of data in Python. Their data type is denoted `str`.
- Lecture examples - Hello World
- More examples - Comments, Strings, and Print

Numbers — Arithmetic Expressions

- There are two kinds of numerical data in Python: integers and decimal numbers.
- Integers correspond to the data type `int`. Decimal numbers are represented by floating-point numbers corresponding to the data type `float`.
- Floating-point numbers have around 15 decimal digits of accuracy.
- In CodeSkulptor, all numbers (even integers) are represented internally as floating-point numbers.
- Lecture examples - Arithmetic Expressions
- More examples - Floats and Ints

Arithmetic Operators — Arithmetic Expressions

- Five basic arithmetic operators; addition (`+`), subtraction (`-`), multiplication (`*`), division (`/`) and exponentiation (`**`)
- CodeSkulptor implements a subset of Python 2. In Python 2, the division operator (`/`) returns a float approximation to the exact answer if either of the operands is a float. If both operands are integers, division returns the exact answer round down to the nearest integer.
- The integer division operator `//` returns the quotient of two numbers..
- Lecture examples - Arithmetic Expressions
- More examples - Arithmetic Operations, Division

Arithmetic Expressions — Arithmetic Expressions

- An arithmetic expression is either a number or an operator applied to two arithmetic expressions.
- Arithmetic expressions are entered as a sequence of numbers and arithmetic operators.
- Expressions are formed by grouping operators and operands via precedence: "Please excuse my dear Aunt Sallie"; parentheses, exponentiation, multiplication, division, addition, subtraction.
- Lecture examples - Arithmetic Expressions
- More examples - Order of Operations for Arithmetic Expressions, Possible Errors for Arithmetic Expressions

Variables — Variables

- Variable names consist of a sequence of letters, number and underscores (`_`).
- Variable names start with a letter or underscore and are case sensitive.

- Single equals (`=`) is used for assignment to variables. Double equals (`==`) is used for testing equality.
- Lecture examples - Variables
- More examples - Variable Naming, Vabiable Assignment, Variable Operations, Formulas

---

Created Wed 20 Mar 2013 11:20 AM PDT

Last Modified Tue 2 Jul 2013 1:14 PM PDT