**Python Data Types**

After learning about variable initialization and assignment, you should be aware that data types are serious business. They can determine the success or failure of your project. Therefore, you should know them extremely well. This document should serve as a quick reference guide for the data types we will be using most often in this class. Research each of the terms below and write their definitions in the boxes below

|  |
| --- |
| **str :**   * **used for text** * **normal (interpreter stores text as one byte/character; good for ASCII text)**  1. **surround string by double (“”) or single (‘’) quotes** 2. **\ is the escape character** 3. **string with line breaks 🡪 use triple quotes** 4. **to use special characters in string without having to escape them** 🡪 **can use a raw string** 🡪 **precede string with r**   **(BUT CAN’T END RAW STRING WITH BACKSLASH for some reason)**   * **and Unicode(computer stores text as text; precede string with u)**  1. **good for non-ASCII characters** |
| **integer:**   * **whole, non-decimal number** * **two types**🡪**integer and long integer (for integers larger than a certain set amount)** * **max range limited by computer** * **can be positive or negative** |
| **float:**   * **decimal numbers (i.e. 7.1)** * **whole numbers with decimal (i.e. 7.0)** * **slower to be processed by computer compared to integer** * **can use scientific notation (i.e. 1.86e47)** |
| **list:**   * **stores group of items (strings, other lists, numbers, etc.)** * **written with square brackets with commas between items (i.e. [1, 2, “first”, [5, 6, 7], “seventh] )** * **access items in list through indexing 🡪integer rep. position of item (first item at position 0) (i.e. x = [1, 2, 3, 4, 5] // print x[1] 🡪 2)** * **access items starting with end through negative indexes🡪 position -1 = last member // position -2 = second to last** * **used with homogenous sequences (in theory)** |
| **tuple:**   * **like lists 🡪 store members (sequences of info)** * **can’t be changed after creation, but can be indexed and sliced (parts of worked on)** * **interpreter knows length and types of items in tuple** * **created using ( ) (i.e. x = (1, 2, 3, 4) )** * **index with square brackets 🡪 x[0] 🡪 1** * **tuple with 1 item 🡪 (1, )** * **empty tuple 🡪 ()** * **tuples used with heterogeneous items (in theory)** |

<http://www.voidspace.org.uk/python/articles/python_datatypes.shtml>