|  |  |
| --- | --- |
|  | Chapter 4 Practice Questions - Lists |
| **Q1** | **What is []** |
| A | Indicates the beginning and ending of a list; an empty list. A list value that contains no items |
| **Q2** | **How would you assign the value ‘hello’ as the third value in a list stored in a variable named spam? (Assume spam contains [2,4,6,8,10])**  **For the following three questions, let’s say spam contains the list [‘a’,’b’,’c’,’d’]** |
| A | Spam[2] = ‘hello’ |
| **Q3** | **What does spam[int(int(‘3’\*2)/11)] evaluate to?** |
| A | ‘d’  Because ‘3’\*2 = 33/11 = 3 = index of spam list |
| **Q4** | **What does spam[-1] evaluate to?** |
| A | ‘d’  Because the integer -1 refers to the last index in a list |
| **Q5** | **What does spam[:2] evaluate to?**  ***For the following three questions, let’s say bacon contains the list [3.14, ‘cat’, 11, ‘cat’, True]*** |
| A | ‘a’,’b’  Because the integer 2 is where the slice ends i.e. at ‘c’ |
| **Q6** | **What does bacon.index(‘cat’) evaluate to?** |
| A | 1 i.e. gives the index number or position of the item in the list |
| **Q7** | **What does bacon.append(99) make the list value in bacon look like?** |
| A | [3.14, ‘cat’, 11, ‘cat’, True, 99] The append method adds the new item to the end of the list |
| **Q8** | **What does bacon.remove(‘cat’) make the list value in bacon look like?** |
| A | [3.14, 11, ‘cat’, True] The remove method removes the first instance of a duplicate item |
| **Q9** | **What are the operators for the list concatenation and list replication?** |
| A | For list concatenation the operator is:  += e.g. names += ‘Steve’  For list replication the operator is:  \*= e.g. names \*= 2  += Addition Assignment; Adds the value and the variable and assigns the new result to the variable |
| **Q10** | **What is the difference between the append() and insert() list methods?** |
| A | Both add new values to a list but the difference is that append() adds the value to the end of the list while the insert() method can insert the value at any point in the list |
| **Q11** | **What are two ways to remove values from a list?** |
| A | Use the remove() method to remove a value passed to it as an argument. Use the del statement when you know the index of the value to be removed e.g. del spam[1] removes the value at index 1 |
| **Q12** | **Name a few ways that list values are similar to string values** |
| A | 1. They share some methods such as len() which do not change their values 2. They both can be copied |
| **Q13** | **What is the difference between lists and tuples?** |
| A | Tuples are typed with parentheses () instead of square brackets and more importantly tuples are immutable i.e. cannot have their values modified, appended or removed |
| **Q14** | **How do you type the tuple value that has just the integer value 42 in it?** |
| A | Add a trailing comma after the value inside the parentheses otherwise Python will think it’s just a value inside a parentheses e.g. (42,) |
| **Q15** | **How can you get the tuple form of a list value? How can you get the list form of a tuple value?** |
| A | Use the function tuple([]) to get the tuple form of a list value and the function list(()) to get the list form of a tuple |
| **Q16** | **Variables that “contain” list values don’t actually contain lists directly. What do they contain instead?** |
| A | They contain references to the list values so that for example if you have two variables (different names) reference to the same list and alter the list referring to the first variable; the second variable list is also altered. |
| **Q17** | **What is the difference between copy.copy() and copy.deepcopy()?** |
| A | They are methods from the copy module:   1. copy.copy(): can be used to copy a mutable list or dictionary so that a change to the copy does not change the original due to the referencing nature of lists and dictionaries 2. deepcopy(): copies inner list of a list |