**Summary of Lecture 10: Collection Part III (Nested Collections, Nested Loops)**

1. **Nested Data Types**:
   * Definition: Collections (lists, dictionaries) containing other collections as elements.
   * Examples: Lists of lists, dictionaries with lists as values.
   * Memory Model: Visualized with Python-specific and non-Python-specific diagrams.
2. **Accessing Elements**:
   * Use multiple indices or keys for nested structures.
   * Example: For a = [[[0, 12], [3]], [4, 5, 6]], access 12 via a[0][0][1].
3. **Updating Elements**:
   * Mutable nested elements (e.g., lists within lists) can be modified directly.
   * Example: b = a[0]; b[1] = 100 updates a if a contains mutable elements.
4. **Copying Nested Data**:
   * Shallow Copy (copy()): Copies outer structure but references inner mutable objects. Changes affect the original.
   * Deep Copy (deepcopy()): Creates independent copies of all nested objects.
5. **Looping Through Nested Structures**:
   * Use nested loops to iterate over elements (e.g., for i in a: for j in i: print(j)).
   * Index-based iteration: for i in range(len(a)): for j in range(len(a[i])): print(a[i][j]).