

I chose to implement this method by generating a list of tuples of all of the free spaces in the map, shuffling those free spaces to provide randomness, and then placing objects on the first `num_objects` worth of spaces. I chose this approach as it was one of the methods described in the instructions and it seemed like the clearest approach to me. The use of the `random.shuffle` method was very beneficial in adding randomness and I quickly found it searching online for a shuffling algorithm.

While implementing the method all went surprisingly smoothly. Following the instructions, the solution is essentially given to you, you just have to implement it. If you look at my work trail I had some issues with PEP 8, but that was just me being dumb.

As required by the assignment my big-O complexity is $O(L \cdot S)$ as the first part of my algorithm runs through every coordinate of the map. The only other data changes is the shuffle function which according to the web runs in $O(1)$ so there is no impact to the overall complexity.