Objects in Memory

- Everything you work with in Python is an object, including numbers, strings, lists, functions, and classes.
- When you create an object, Python automatically allocates space in memory to store the object and its associated data.
- Variables are references to objects in memory. They don't store objects directly.
- Python has automatic garbage collection. This means that is reclaims memory automatically when an object is no longer references in the program.
- Each object has a unique number called "id." This number represents the unique memory address where the object is currently stored. It's unique for each object during its lifetime.
- An object's lifetime is determined by the number of references to it in the program. If the number of references reaches 0, the object is garbage collected.
- To get this unique id, you can call the **id()** function.
- Objects have an identity, type, and value.
- The 'is' operator returns True when the two operands point to the same object in memory. Else, it returns False.