Hashability

- In Python, set keys need to be "hashable"
- Generally, must be immutable:
    - Lists are mutable (can be modified), so cannot be keys in a set or dictionary
    - Strings, integers, floats, bools are immutable, so can be used
- If something is mutable, then when passed into a hash function it may have a different result. Keys cannot change their values - violates expectation of hash table and may not be able to find keys again later
- Example: if A = [1, 2] were allowed to be a key, A.append(3) could make it impossible to look up the elements assigned to A using A's hash, because when hashed, [1, 2, 3] could point to a different bucket from [1, 2]

5. Sorting