Which of the following are true statements about putting `final` on `n`?

- [ ] final can't be used on n because n is a parameter of a method
- [ ] final can't be used on n because its type is int and int is already immutable
- [x] final can't be used on n because n is reassigned to other integer values in the body of the method
- [ ] final can be used on n, and it prevents n from being reassigned

✔

Which of the following are true statements about putting `final` on `list`?

- [ ] final can't be used on list because list is a local variable
- [ ] final can't be used on list because its type is List and List is a mutable type
- [ ] final can't be used on list because list.add() is used to change the list in the body of the method
- [x] final can be used on list, and it prevents the list variable from being reassigned

✔

Explanation

`final` can't be used on `n` because `n` needs to be reassigned in the body of the method. But `final` can indeed be used on `list`.

`final` can be used on both parameters and local variables. When used on a parameter, `final` means that the parameter is assigned when the method is called, and then can't be reassigned during the body of the method. When used on a local variable, `final` means that the variable can't be reassigned after its first assignment, until the variable's scope ends.

`final` can be used on variables of any type -- not just immutable types like int, but also mutable types like `List`. If a `final` variable points to a mutable object, then the variable cannot be reassigned, but the object it points to can still be mutated, say by calling `add()` on a `List`.

Submit