# Test Aplicado

@Test

**public** **void** **testCrearUsuarioAcreditado**() {

UsuarioAcreditado usuario = biblioteca.crearUsuarioAcreditado("05021768483", "Juan Pérez");

Assert.*assertNotNull*(usuario);

Assert.*assertEquals*("05021768483", usuario.getId());

Assert.*assertEquals*("Juan Pérez", usuario.getNombreCompleto());

Assert.*assertTrue*(biblioteca.getUsuarios().contains(usuario));

}

## Fallo del código

El método crearUsuarioAcreditado retorna un objeto UsuarioAcreditado, pero la lista biblioteca.getUsuarios() **no contiene ese mismo objeto** (según el método equals() de la clase UsuarioAcreditado).

**¿Por qué ocurre esto?**

En tu clase Biblioteca, el método es:

Java

public UsuarioAcreditado crearUsuarioAcreditado(String id, String nombre) {

UsuarioAcreditado u = new UsuarioAcreditado(id, nombre);

usuarios.add(new UsuarioAcreditado(id, nombre));

return u;

}

**Aquí el problema:**

* Creas un objeto u.
* Luego agregas a la lista un **nuevo** objeto UsuarioAcreditado(id, nombre) (que no es u, sino otro objeto con los mismos datos).
* Retornas u.

Por lo tanto, getUsuarios().contains(usuario) será false a menos que la implementación de equals() compare solo por id y nombre, y no por referencia.

* Si tu clase UsuarioAcreditado **no** ha sobrescrito correctamente equals() y hashCode(), la comparación por defecto es por referencia, y por eso falla.

## Soluciones

**1. Corregir el método en Biblioteca**

Haz que agregues a la lista el mismo objeto que retornas:

Java

public UsuarioAcreditado crearUsuarioAcreditado(String id, String nombre) {

UsuarioAcreditado u = new UsuarioAcreditado(id, nombre);

usuarios.add(u);

return u;

}