

# Single Responsibility

Code should only have a single reason to change.

This applies to:

- Classes
- Methods
- Data objects

Code that has multiple responsibilities should be split into multiple parts.

*Caveat: Still must avoid duplication, parallel hierarchies*

Smells:

- Large Class
- Feature Envy
- Primitive Obsession

```
// The "And" in the name is
// a dead giveaway:
// This method does two things!
void findItemAndInsertIntoDataBase(
    DBObject object,
    List<DBObject> incoming_items,
    backendDb server)
{
    int search_index =
        incoming_items.lastIndexOf(object);
    if(search_index <= incoming_items.size()) {
        DBObject item = incoming_items.get(search_index);
        server.insertRecord(item);
    }
}
```