

Bloaters		
Long Method	Too many lines of code.	1, [Replace Temp with Query, 3, 4], [Replace Method with Method Object], [Decompose Conditional, 1]
Large Class	Class contains many fields/methods/lines of code.	2, Extract Subclass, Extract Interface, Duplicate Observed Data
Primitive Obsession	Use of constants for coding information. Use of primitives instead of small objects for simple tasks.	Replace Data Value with Object, [3 or 4], [Replace Type Code with Class, Subclasses or State], Replace Array with Object
Long Parameter List	More than three or four parameters for a method.	Replace Parameter with Method Call, 4, 3
Data Clumps	Different parts of the code contain identical groups of variables	2, 3, 4. 1. Extract Method 2. Extract Class 3. Introduce Parameter Object 4. Preserve Whole Object

Refactoring Benefits
- Improves code organization.
- Reduces code duplication.
- Simplifies support.
- Can improve organization of code.
- Easier maintenance.
- Code becomes more flexible
- Better Code Readability

Object-Orientation Abusers		
Alternative Classes with Different Interfaces	2 classes perform identical functions but have different method names.	Rename Methods, [Move method, Add Parameter, Parameterize Method], Extract Superclass
Refused Bequest	Subclass uses only some of the methods and property	Replace Inheritance with Delegation, Extract Superclass
Switch Statements	Complex switch operator	Replace Conditional with Polymorphism, Ex
Temporary Field	Get their values only under certain circumstances	Extract class, Replace Method with Method Object
Incorrect application of object-oriented programming principles.		

Change Preventers	
Divergent Change	Change many unrelated methods when you make changes to a class.
Parallel Inheritance Hierarchies	Whenever you create a subclass for a class, you find yourself needing to create a subclass for another class.
Shotgun Surgery	Making any modifications requires that you make many small changes to many different classes.
Need to change something in one place in your code, you have to make many changes in other places too.	

Dispensables	
Comments	A method is filled with explanatory comments.
Duplicate Code	Two code fragments look almost identical.
Data Class	Class that contains only fields and crude methods for accessing them. Don't contain any additional functionality and can't independently operate on the data that they own.
Dead Code	A variable, parameter, field, method or class is no longer used
Lazy Class	Class doesn't do enough to earn your attention,



Dispensables (cont)

Speculative Generality There's an unused class, method, field or parameter.

Something pointless and unneeded whose absence would make the code cleaner, more efficient and easier to understand.

Couplers

Feature Envy A method accesses the data of another object more than its own data. Move, Extract Method

Inappropriate Intimacy Uses the internal fields and methods of another class [Move Method and Field], [Extract Class and Hide Delegate], [Replace Delegation with Inheritance]

Incomplete Library Class [Introduce Foreign Method or Local Extension]

Message Chains a->b()->c()->d() Hide Delegate, [Extract Method and Move Method]

Middle Man Class only delegates work Remove Middle Man

Excessive coupling between classes



By **Heavenel**

cheatography.com/heavenel/

Published 1st November, 2019.

Last updated 1st November, 2019.

Page 2 of 2.

Sponsored by **ApolloPad.com**

Everyone has a novel in them. Finish Yours!

<https://apollopad.com>