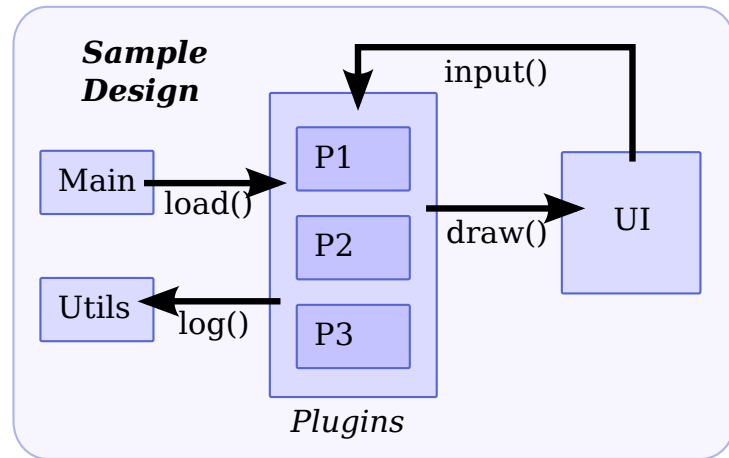


Templates

Software design groups related code, such as plugins or widgets, so that large system can be reasoned about.



Interfaces and **classes** only capture part of this software design. They tell us that Plugins have load and input methods and UI has a draw method.

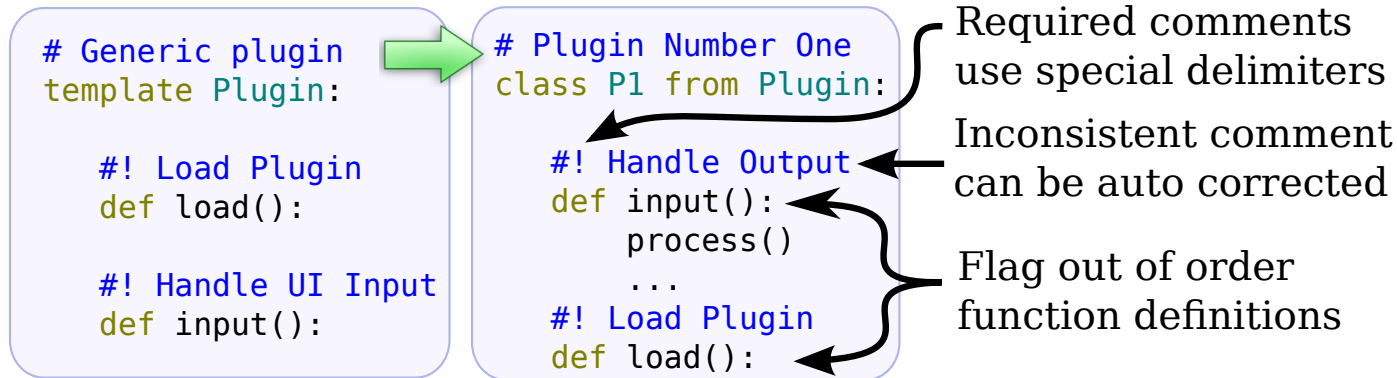
The fact that plugins should call the draw() method or that UI calls the plugin's input() is lost during coding.

We would also like all plugins to be implemented consistently.

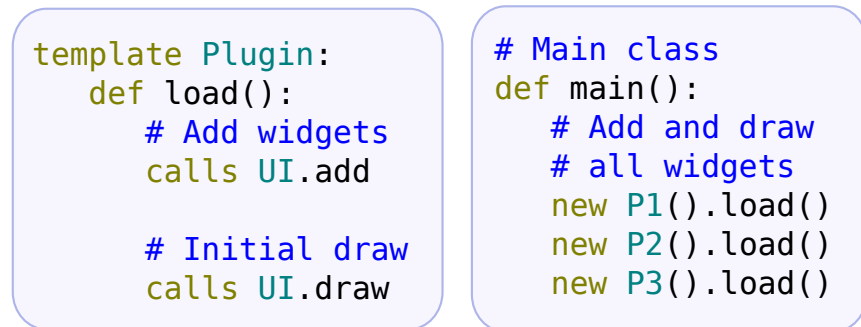
Like interfaces, **templates** list the methods that an object provide, but they also help with syntactic formatting and provide information about the behavior of the object, such as expected function calls.

Consistent Formatting

Provide easy synchronization of code
Automatic stubbing of new classes



Behavioral information



Static checks for method existence and also method behavior.

Plugin.load may call UI.add directly or using a helper method.

Behavioral keywords:

calls func - required call to func
maycall func - optional call to func