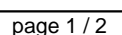


Download



anti-pattern? Is there a better design pattern for this situation? A: This design pattern is common. You are basically defining a subclass for each type of list. The common attributes, type, and list length are all shared by all the subclasses. Then for each subclass, you have one or more specialized implementations of the list. For example: class FileList { protected IList fileNames = new ArrayList(); public void AddFile(string fileName) { fileNames.Add(fileName); } public IList ListFiles() { return fileNames; } } class LocalFileList : FileList { public LocalFileList(IList 2d92ce491b