1. Правильный нейминг переменных (они должны отражать суть и намерения программиста)

Имя переменной, функции или класса должно отвечать на все главные вопросы. Оно должно сообщить, почему эта переменная (и т. д.) существует, что она делает и как используется. Если имя требует дополнительных комментариев, значит, оно не передает намерений программиста.

Не использовать O и L как названия переменных  
Не использовать List и array или другие названия контейнеров в коде  
accountList == bankaccount

«Числовые ряды» вида (a1, a2, .. aN) являются противоположностью сознательного присваивания имен. Такие имена не дезинформируют — они просто не несут информации и не дают представления о намерениях автора.

Неинформативные слова также применяются для создания бессодержательных различий. Допустим, у вас имеется класс Product. Создав другой класс с именем ProductInfo или ProductData, вы создаете разные имена, которые по сути обозначают одно и то же. Info и Data не несут полезной информации, как и артикли a, an и the. Следует учесть, что использование префиксов a и the вовсе не является ошибкой, но только при условии, что они создают осмысленные различия.

При отсутствии жестких именных схем имя moneyAmount не отличается от money, customerInfo не отличается от customer, accountData не отличается от account, а theMessage — от message. Записывайте различающиеся имена так, чтобы читатель кода понимал, какой смысл заложен в этих различиях

Использовать удобнопроизносимые названия (без шифров и абривиатур)

Выбирать имена удобные для поиска

Лично я считаю, что однобуквенные имена могут использоваться ТОЛЬКО для локальных переменных в коротких методах. Длина имени должна соответствовать размеру его области видимости [N5]. Если переменная или константа может встречаться или использоваться в нескольких местах кодового блока, очень важно присвоить ей имя, удобное для поиска

Не использовать префиксы в названиях перменных

m\_dsc = description

Такая проблема часто возникает при использовании однобуквенных имен переменных. Конечно, счетчик цикла можно назвать i, j или k (но только не l!), если его область видимости очень мала, и он не конфликтует с другими именами. Это связано с тем, что однобуквенные имена счетчиков циклов традиционны. Однако в большинстве других контекстов однобуквенные имена нежелательны; в сущности, вы создаете временный заменитель, который должен быть мысленно преобразован пользователем в реальную концепцию. Нет худшей причины для выбора имени c, чем та, что имена a и b уже заняты

Старайтесь не использовать в именах классов такие слова, как Manager, Processor, Data или Info. Имя класса не должно быть глаголом.