Slide 1

Slide 2

Большая часть программных систем необоснованно перегружена практически ненужными функциями, что ухудшает удобство их использование конечными пользователями, а также усложняет их поддержку и развитие разработчиками. Следование принципу «KISS» позволяет разрабатывать решения, которые просты в использовании и в сопровождении.

Slide 3

Что такое принцип KISS?

KISS — это принцип проектирования, принятый в военно-морских силах США в 1960. Принцип KISS утверждает, что большинство систем работают лучше всего, если они остаются простыми, а не усложняются. Поэтому в области проектирования простота должна быть одной из ключевых целей и следует избегать ненужной сложности.

Slide 4

Есть много вариантов расшифровки аббревиатуры: «keep it simple, stupid». Это ‘Keep it Simple, Silly’, ‘keep it short and simple’, ‘keep it simple and straightforward’ и ‘keep it small and simple’.

Slide 5

По имеющимся сообщениям, термин был придуман Кларенсом Джонсоном, ведущим военным инженером Lockheed Skunk Works (создатели Lockheed U-2, SR-71 Blackbird и многих других самолётов).

Slide 6

В то время как уже несколько десятилетий популярно использование расшифровки «Keep it simple, stupid. Этот принцип лучше всего иллюстрируется историей, когда Джонсон вручил команде инженеров-авиаконструкторов набор инструментов, поставив им условие: механик среднего уровня должен суметь отремонтировать реактивный самолёт, который они проектировали, в полевых условиях только с этими инструментами. Таким образом, «stupid» относится к отношению между тем, что всё ломается, и сложностью необходимого для этого ремонта.

Slide 7

В программировании следование принципу KISS можно описать так:

не стоит использовать заумные и неочевидные решения, нужно писать код максимально просто

не стоит подключать огромную библиотеку, если вам от неё нужна лишь пара функций;

не имеет смысла беспредельно увеличивать уровень абстракции, надо уметь вовремя остановиться;

абсолютная математическая точность или предельная детализация нужны не всегда — большинство систем создаются не для запуска космических шаттлов, данные можно и нужно обрабатывать с той точностью, которая достаточна для качественного решения задачи, а детализацию выдавать в нужном пользователю объёме, а не в максимально возможном объёме.

Часто наиболее правильное решение – это наиболее простая реализация задачи, в которой нет ничего лишнего.

Чем проще код, тем легче в нём разобраться, как вам, так и другим людям, занимающимся его поддержкой. Под простотой подразумевается отказ от использования хитроумных приемов и ненужного усложнения.

Slide 8

Примеры:

В качестве примера нарушения этого принципа можно назвать написание отдельной функции только лишь для осуществления операции сложения или использование побитового оператора (right shift >> 1) для деления целых чисел на 2.

(4 >> 1) === (4 / 2)

Последнее может оказаться более эффективным для некоторых программных компиляторов, чем обычное деление / 2, но при этом очень сильно снижается понятность кода.

Важное примечание: Для JS движков эффективность операции сдвига нивелируется дорогостоящим приведением к целочисленному.

Применяя такой подход, вы осуществляете clever coding and over-optimization. И то, и другое в долгосрочной перспективе будет делать ваш код всё менее и менее понятным как другим разработчикам, так и вам самим, ведь, возможно, вам придётся разбираться с этим кодом снова через месяц, два, год.

Slide 9

 KISS – это очень общий и абстрактный принцип проектирования, который содержит в себе практически все остальные принципы проектирования. Принципы проектирования описывают как писать «хороший» код. Однако что значит хороший код? Некоторые считают, что это код, который выполняется максимально быстро, некоторые – что это код, в котором задействовано как можно больше паттернов проектирования… Но верный ответ лежит на поверхности. Код – это информация в чистом виде.

Slide 10

А основные критерии ценности информации – это 1)достоверность 2)доступность 3)понятность. То, почему важны достоверностью и доступность – очевидно. От кода нет проку, если он работает с ошибками или если сервер с приложением «лежит». Почему же важна понятность кода? В понятном коде проще искать ошибки, проще его изменять, дорабатывать и сопровождать. Итак, понятность – основная ценность, к которой должен стремиться программист. Однако тут есть одна неувязочка. Дело в том, что понятность – вещь сугубо субъективная. Нужен некий более объективный критерий понятности. И этот критерий – простота. Действительно, простое приложение более понятное, нежели сложное. Однако простоты достичь сложно.

Slide 11

Вот что пишет Питер Гудвин в книге «Ремесло программиста»:

Если проект прост, его легко понять… Разработать простой проект не так легко. Для этого нужно время. Для всякой сколько-нибудь сложной программы окончательное решение получается в результате анализа огромного объема информации. Когда код хорошо спроектирован, кажется, что он и не мог быть иным, однако возможно, что его простота достигнута в результате напряженного умственного труда (и большого объема рефакторинга). Сделать простую вещь сложно. Если структура кода кажется очевидной, не надо думать, что это далось без труда.

Slide 12

**Итак, принцип проектирования KISS провозглашает, что простота кода – превыше всего, потому что простой код – наиболее понятный.**  
Практически все принципы проектирования направлены на достижение понятности кода. Нарушая какой-либо принцип проектирования, вы уменьшаете понятность кода.

Slide 13

Непонятный код автоматически вызывает у человека ощущение того, что код сложный, так как его сложно понимать и модифицировать. При нарушении любого из этих принципов также нарушается и принцип KISS. Поэтому можно говорить, что KISS включает почти все остальные принципы проектирования.

Slide 14  
Патерны проектирования описывают наиболее удачные, простые и понятные решения некоторых проблем. Если вы используете паттерн проектирования там, где нет проблемы, которую решает данный паттерн – то вы нарушаете KISS, внося ненужные усложнения в код. Если вы НЕ используете паттерн проектирования там, где есть проблема, соответствующая паттерну – то вы опять-таки нарушаете KISS, делая код сложнее, чем он мог бы быть.

Slide 15

Всвязи с тем, что представления разных людей о таком понятии как «простота» могут различаться, приобрели широкое распространение следующая **заблуждения относительно KISS-a**:  
*Заблуждение 1.* Если считать, что простой код – это такой код, который проще всего написать, то можно истолковать, что принцип KISS призывает писать первое что взбредёт в голову, вообще не задумываясь о проектировании.  
*Заблуждение 2.* Если считать, что простой код – это такой код, для написания которого требуется как можно меньше знаний, то можно истолковать, что принцип KISS призывает не использовать паттерны проектирования.

Slide 16

Под простотой же в данном случае следует понимать [не сложный, лишенный искусственности, самый естественный, не трудный, легко доступный для понимания](http://www.classes.ru/all-russian/russian-dictionary-Ushakov-term-59601.htm).

Slide 17

Еще раз вернёмся к истории, мы уже говорили, что принцип KISS зародился в авиастроении и там известны случаи, когда слишком усердные рабочие прибивали на самолёт лишние пластины брони, чтобы сделать самолёт более живучим в бою, в результате чего масса самолёта становилась больше расчётной и самолёт попросту не мог взлететь. Кроме того, квалификация многих рабочих была низкой. В таких условиях конструкции самолётов, которые пьяный неквалифицированный рабочий не смог бы собрать неправильно, даже если бы захотел, обладали особенной ценностью. Один из отголосков конструкторских решений того времени — невозможность перепутать и воткнуть неверный штекер в гнездо внутри компьютера. Однако, если результатом труда авиа-инженера является чертёж, по которому будет создан продукт, то в случае с программистом продуктом является сам чертёж (образно выражаясь). В случае программиста он должен написать код так, чтобы пьяный неквалифицированный программист смог внести в него изменения в соответствии с изменившимися бизнес-требованиями (то есть изменить чертёж, а не собрать самолёт). В силу различий в специфике авиастроения и программирования, расшифровка «Keep it simple stupid», подходящая в авиастроении, уже не так хорошо отражает суть принципа для программиста. Многие ленивые программисты расшифровывают «сделайте это до идиотизма простым» как «не утруждайте себя проектированием».

Slide 18/1

К счастью, у KISS есть ещё и [некоторые другие расшифровки](https://en.wikipedia.org/wiki/KISS_principle), о которых я говорил выше, одна из которых, на мой взгляд, лучше всего отражает суть KISS в программировании — «keep it simple and straightforward». Straightforward значит простой, честный, прямолинейный, откровенный. «Keep it simple and straightforward», таким образом, можно вольно перевести как «Сделайте это простым и декларативным», а для достижения декларативности требуется проектирование.

Slide 18/2

Используйте KISS в своей работе и всем нам станет лучше!