

Code hygiene: на примере R

Георгий Мороз

Международная лаборатория языковой конвергенции (НИУ ВШЭ)

18 июня 2021, ReproducibiliTea (Moscow)

To comment or not to comment?

Pets or livestock?

DRY or WET?

To comment or not to comment?

- через полгода вы все забудите (поэтому кроме комментариев, стоит еще следовать принципу простоты, **KISS, Keep it simple, stupid**)
- легче ориентироваться в коде и его исправлять, в случае возникновения ошибки
- не стыдно отдавать и показывать код (рано или поздно придется) другим (даже не знающим языка программирования)
- код может стать обучающим материалом

To comment or not to comment?

- через полгода вы все забудите (поэтому кроме комментариев, стоит еще следовать принципу простоты, **KISS, Keep it simple, stupid**)
- легче ориентироваться в коде и его исправлять, в случае возникновения ошибки
- не стыдно отдавать и показывать код (рано или поздно придется) другим (даже не знающим языка программирования)
- код может стать обучающим материалом
- а еще в комментарии имеет смысл записать версии R и всех пакетов...

Как комментировать?

- каждый комментарий должен отображать, что вы хотите сделать этим кусочком кода
 - а не, например, просто повторять то, что очевидно из названий функции
- каждый комментарий нужно обновлять, если ваши цели поменялись
- подробность и количество комментариев зависит от ваших целей и сил, это шкала:
 - код без комментариев
 - ...
 - код который можно переписать на другом языке программирования на основе комментариев

Как комментировать в R?

- Скрипт

- # ваш комментарий

- Ctrl+R -----

- RMarkdown

To comment or not to comment?

Pets or livestock?

DRY or WET?

Pets or livestock? (см. пост Maëlle Salmon, 2017/08/08)

- люди видят в учебниках названия переменных x , y , `fit1`, `fit2`
- имеет смысл заботиться о ваших переменных, файлах, папках, путях, кусочках кода в Rmakrdown

Pets or livestock? (см. пост Maëlle Salmon, 2017/08/08)

- люди видят в учебниках названия переменных x , y , `fit1`, `fit2`
- имеет смысл заботиться о ваших переменных, файлах, папках, путях, кусочках кода в Rmakrdown
 - через полгода вы все забудите
 - легче ориентироваться в коде и его исправлять, в случае возникновения ошибки
 - не стыдно отдавать и показывать код (рано или поздно придется) другим (даже не знающим языка программирования)
 - код может стать обучающим материалом

Pets or livestock? (см. пост Maëlle Salmon, 2017/08/08)

- люди видят в учебниках названия переменных `x`, `y`, `fit1`, `fit2`
- имеет смысл заботиться о ваших переменных, файлах, папках, путях, кусочках кода в Rmakrdown
 - через полгода вы все забудите
 - легче ориентироваться в коде и его исправлять, в случае возникновения ошибки
 - не стыдно отдавать и показывать код (рано или поздно придется) другим (даже не знающим языка программирования)
 - код может стать обучающим материалом
- имеет смысл придерживаться единого стиля в назывании
 - `extract_coefficients`
 - `extract.coefficients`
 - `ExtractCoefficients`

To comment or not to comment?

Pets or livestock?

DRY or WET?

DRY or WET?

- DRY — Don't repeat yourself
- WET — Write Everything Twice / We enjoy typing
- Напишите функцию
- Напишите пакет
- отдайте пакет в [rOpenSci](#)
 - через полгода вы ничего не забудите — вас замучают issues доброжелателей и ревьюеров
 - легче ориентироваться в коде и его исправлять, в случае возникновения ошибки
 - воспользоваться вашим кодом можно, просто установив пакет и запустив функции
 - документация пакетов в R устроена значительно лучше, чем отдельные комментарии

Внимабо за спасиние!