## Hardcode

В четвертом ДЗ по структурным потернам захардкодил название файла, а также текст, который в него пишется:  


**Чтобы устранить:** можно вынести формирование урла и названия файла в отдельный метод или класс работы с файлом. Туда-же передать параметром текст, который нужно в него записать.

## Copy–Paste

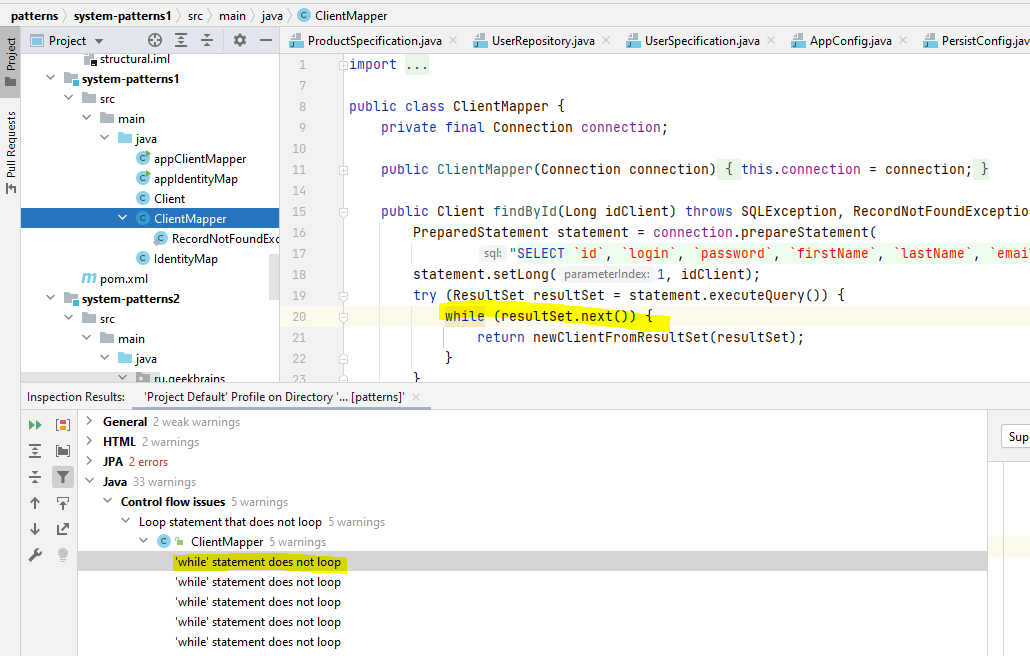
Этот паттерн я у себя не нашел, т.к. борюсь с ним уже давно. Как только вижу, что я у себя копирую что-то сразу выношу это в метод или класс. Возможно, это является «золотым молотком» или «преждевременной оптимизацией», и порой я просто придумываю себе работу, ради одного единственного копипэйста.

## God object

100% писал код с этим антипатерном, но не нашел его в проектах. Видимо во время рефакторинга разделил на несколько классов.

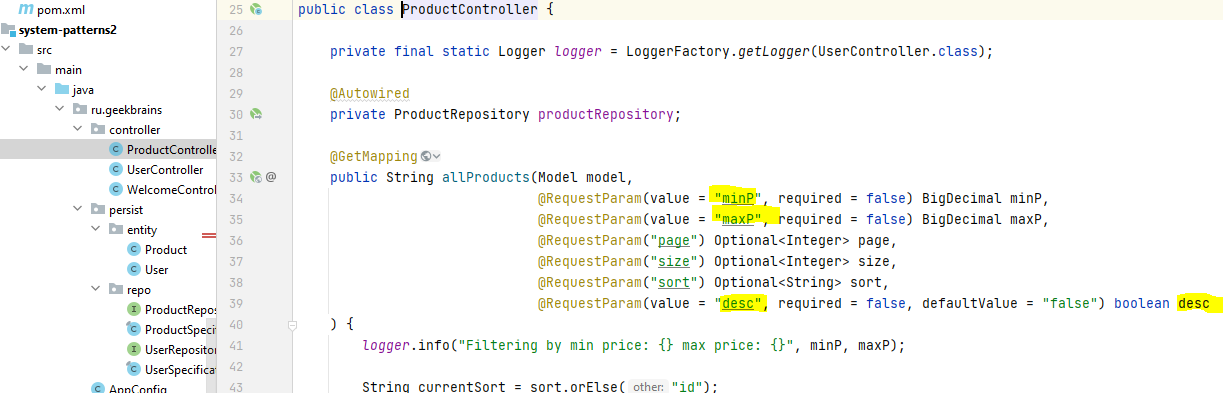
## Blind faith

С помощью code analyzer-а нашел еще один антипатерн, слепая вера, не убедился, что цикл может уйти в бесконечность:



Этот антипатерн в данном случае не получится устранить, у ResultSet невозможно получить все элементы без бесконечного цикла, тут надо просто поверить ☺

## Cryptic Code

Для второй части системных паттернов, взял одну из предыдущих работ по спрингу и немного подправил. Сейчас заметил, что там немного пере шифровал, без детального рассмотрения кода сходу не понятно что делают переменные и что означают значения:  


**Чтобы устранить:** переименовать переменные: minP, maxP и desc в minPrice, maxPrice и description, значения передаваемые в параметре также переименовать, и не забыть, это поправить в шаблоне.

! PS: После лекций по антипатернам пришел на работу, открыл код автомэйшена и понял, это сплошной антипатерн!!! Как теперь с этим жить… Весь день провел над рефакторингом своей зоны ответственности ☺ К сожалению примеры не могу привести, т.к. дома нет доступа к рабочему гиту.