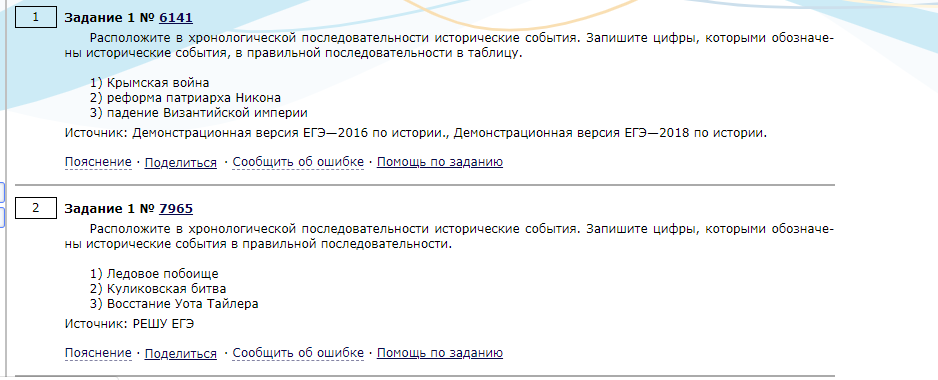
# Постановка задачи

Реализовать алгоритм генерирующий задание по истории следующего формата:

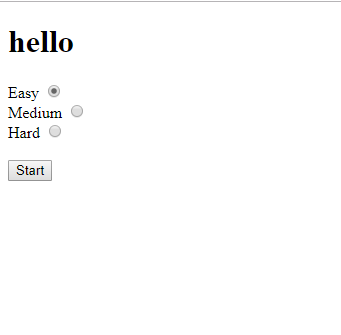


# Ход работы

Был выбран язык программирования Java для написания логики взаимодействия с пользователем. Как хранилище данных была использована СУБД Postgresql. Общий принцип работы следующий: из хранилища данных достаются случайные записи. Из них создается задание для пользователя. Пользователь решает задание, записывает ответ. Ответ проверяется и пользователю сообщается результат.

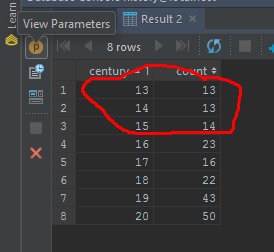
# Описание работы алгоритма

В программе было предложено 3 подхода к формированию задания для пользователя. Первому подходу соответствует название easy, второму medium и третьему hard (не смотря на названия, нельзя сказать однозначно, что подход 1 генерирует задачки проще чем подход 2 и 3. Другими словами, все задания зависят от «рандомайзеров» языка Java и однозначно судить о сложности подходов нельзя). Стоит заметить что на хранилище данных накладываются строгие ограничения на данные. Не больше одного события на год, чтобы избежать коллизии в заданиях.

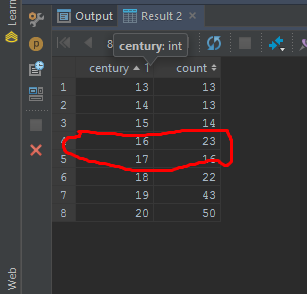


Суть первого подхода в следующем:

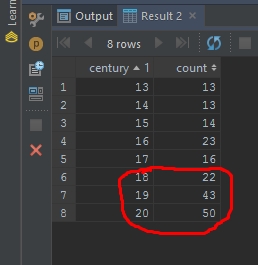
1. Из хранилища берутся 3 исторических события. Первое событие из нескольких первых доступных веков



Второе событие из середины доступных веков

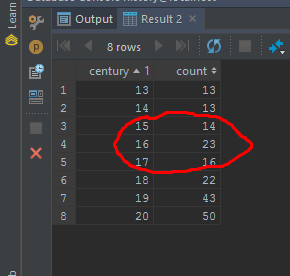


3 событие из последних оставшихся веков

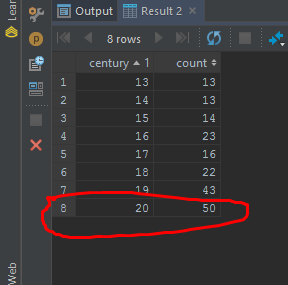


События перемешиваются и в случайном порядке отдаются пользователю

1. Берутся наугад 3 стоящих рядом столетия и из каждого из них берется по одному событию. Далее события перемешиваются для получения случайного порядка и отдаются пользователю



1. Берутся 3 события но уже из одного столетия перемешиваются и отдаются пользователю.



# Выводы

Представленные подходы хорошо подходят для генерации заданий, которые содержат конечное множество исходных данных. А качество и объем генерированных заданий целиком зависят от объема данных в хранилище.

Так же можно усложнить задание, если добавить к записям в хранилище более точную дату. И в условиях выборки опираться на месяц и день события.

Программный код: https://github.com/RomanHi/TaskGenerator