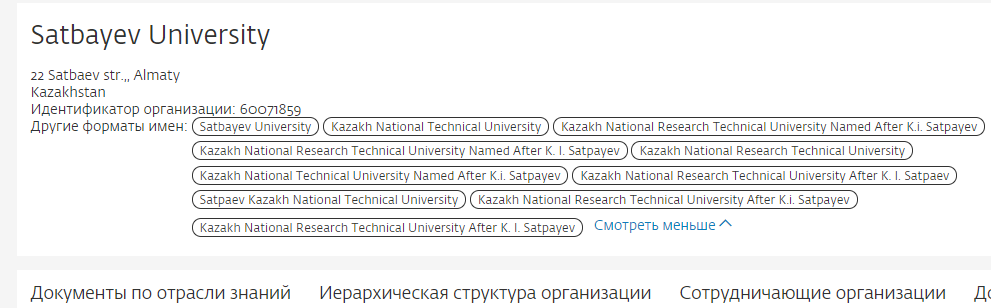
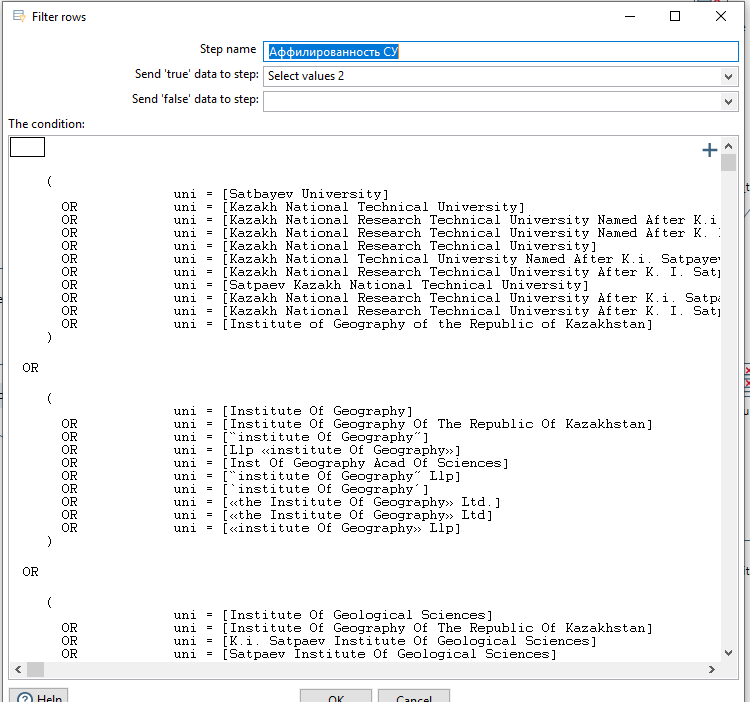
**Загрузка НТ в базу**

1. [Microsoft Excel input] получаем список Scopus ID с файла Scopus ID(с official+Данияр).xls;
2. В [Microsoft Excel input 2] получаем скрауленные данные Леонидом со Scopus API\_and\_SU\_docs\_cit\_2015\_2021\_20210614.xlsx;
3. В узле [Аффилированность] фильтруем публикаций, чьи аффилированности совпадают с вариантами названия Satbayev University:

<https://www.scopus.com/affil/profile.uri?afid=60071859>



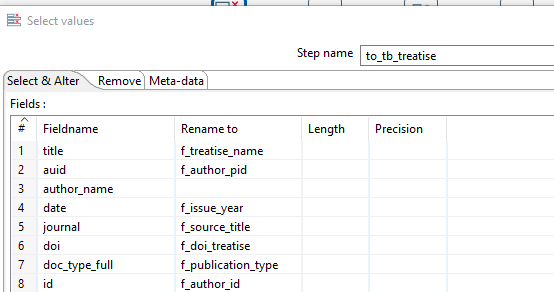
И в сотрудничающих организациях:

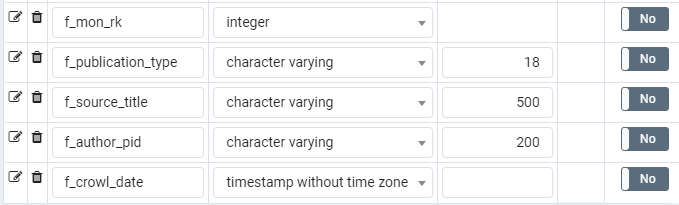
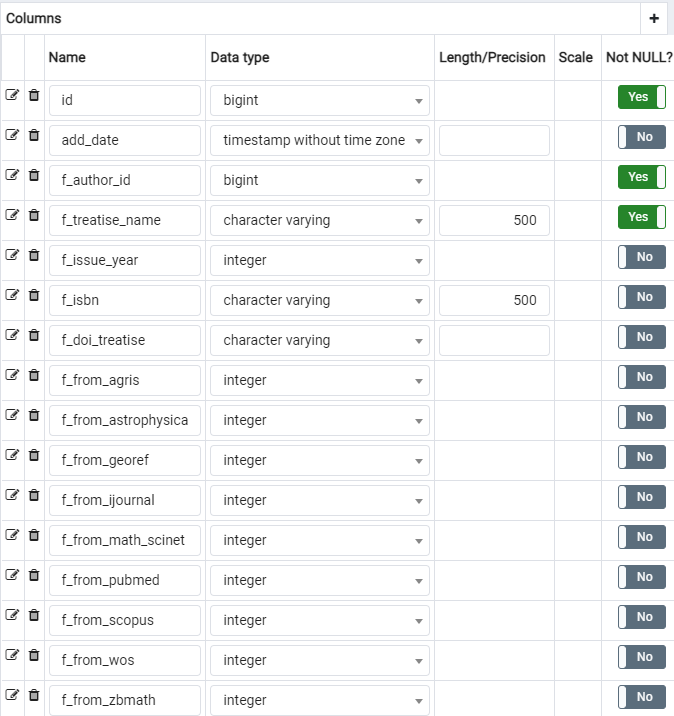


1. В [Merge join] связываем скопус айди автора с id с [Microsoft Excel input];
2. Далее, для получения идентификаторов этих людей связываем с таблицей tb\_indiv\_subject с pgadmin через их ИИН;
3. После разделяем данные на 5 частей, для пяти таблиц;

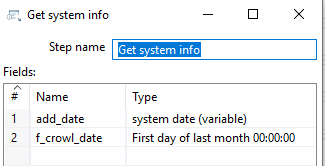
***TB\_TREATISE - хранит общие сведения о публикации***

1. Узел [to\_tb\_treatise] выбирает нужные поля для загрузки их в tb\_treatise, переименовываем поля как в tb\_treatise;

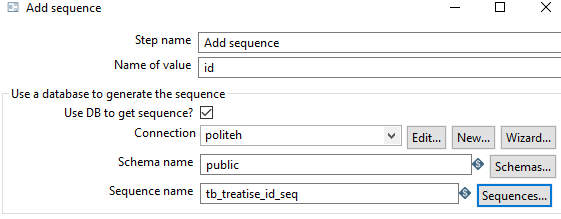




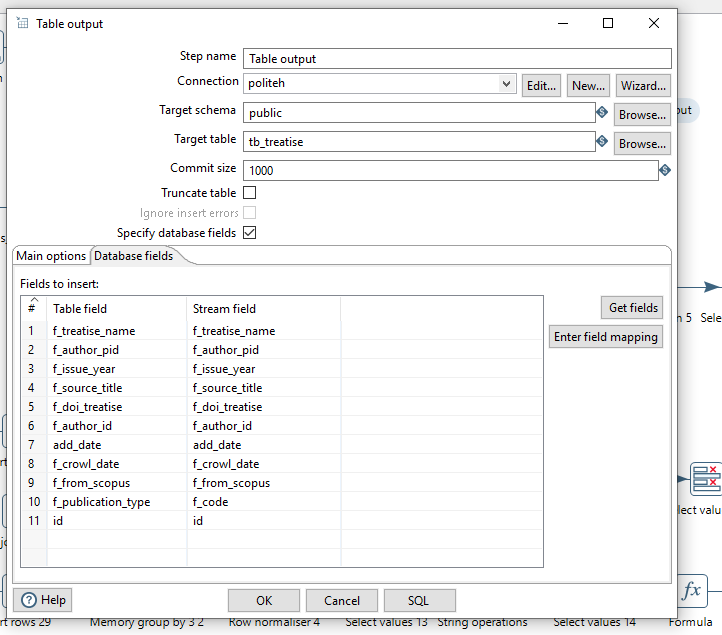
1. В [Get system info] получаем дату добавления данных и дату краулинга публикаций:



1. В [Add constants] добавляем поле f\_from\_scopus=1 для всех публикаций;
2. Для получения id типа публикации со справочника в [Merge join 3] получаем id для каждого типа публикации;
3. В [add sequence] создаем последовательность ID, он получает sequence таблицы tb\_treatise\_id\_seq:

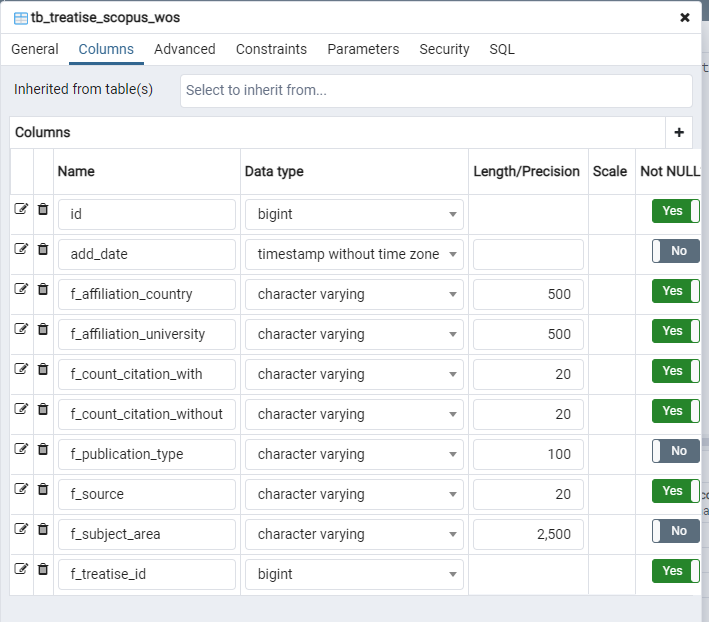


1. [Table output] загружает данные в tb\_treatise, неупомянутые поля будут как пустые:

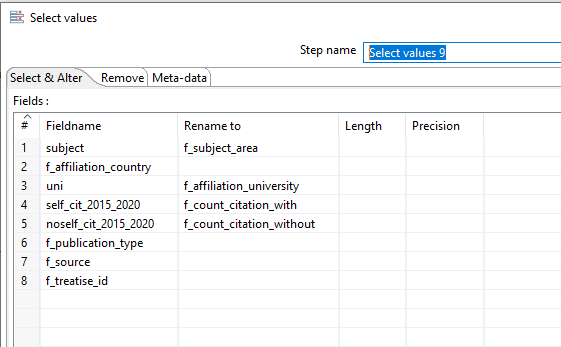


***TB\_TREATISE\_SCOPUS\_WOS – хранит характеристики публикации со Scopus/WOS***

1. [tb\_treatise\_scopus\_wos] из данных выбираем поля подходящие для таблицы tb\_treatise\_scopus\_wos;



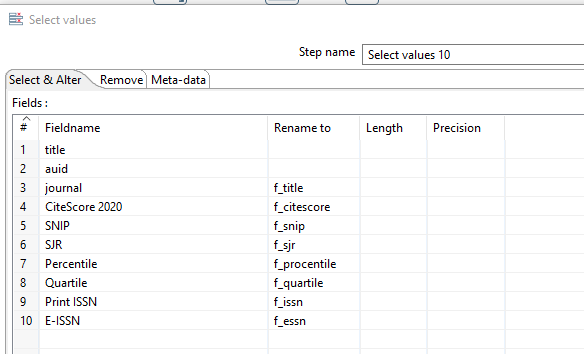
1. В таблице tb\_treatise\_scopus\_wos поле f\_affiliation\_country должен содержать город и страну автора, поэтому в узле [Concat fields] объединяем поля city, country через запятую;
2. Далее делаем группировку для каждой публикации нужно отрасль знаний объединить с помощью запятой, для других характеристик берем последние значения.
3. Теперь привязываем к публикациям их тип публикации, собственно id типа публикации в [Merge join 4];
4. Убираем дубли с помощью [Memory group by 4];
5. Для получения treatise\_id связываем с tb\_treatise [Merge join 5];
6. Избавляемся от дублей;
7. Добавляем с помощью [Add constants 2] f\_source=S1, что означает публикация со скопуса;
8. В [Select values 9] переименовываем поля под названия таблицы;



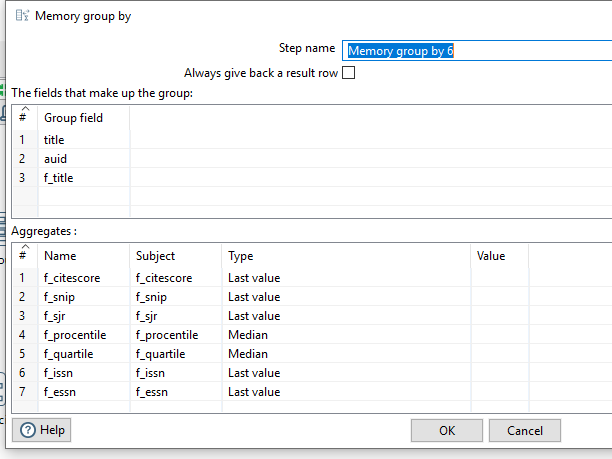
1. Добавляем add\_date, id и данные загружаются с помощью [Table output 2];

***TB\_JOURNAL – хранит характеристики журналов каждой публикации***

1. Данные журналов «journ-scops.xlsx» загружаем в узле [journal];
2. В [Merge join 6] связываем по названию журнала публикации;
3. [Select values 10] выбираем поля и переименовываем под названия полей в tb\_journal;



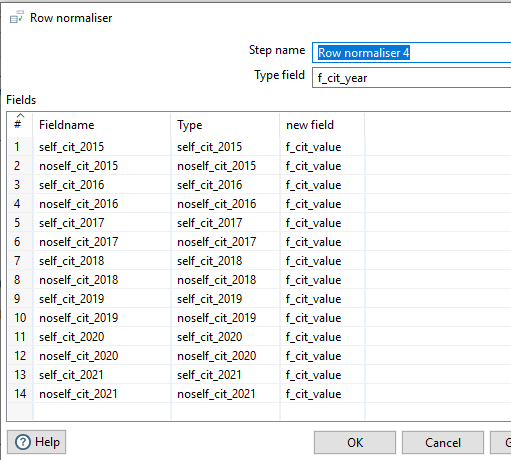
1. В [Memory group by 6] группируем и получаем последние значения для некоторых полей, и для процентиля и квартиля получаем медианные значения, т.к. характеристики журнала повторяются в зависимости от отраслей знаний:



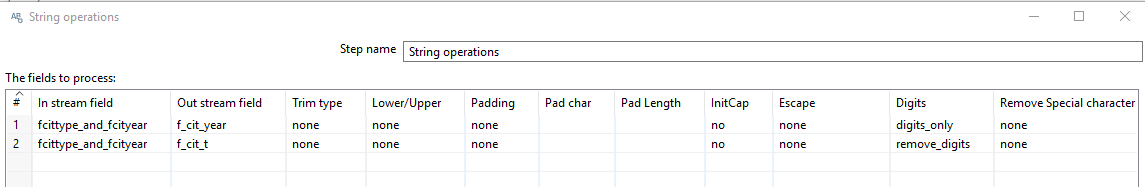
1. Получаем с помощью [Merge join 7] получаем treatise\_id с tb\_treatise;
2. Т.к. в данных журнала квартили только числа, их нужно подогнать под тип Q1, поэтому сначала с помощью Add constants добавляем букву Q и через узел [Concat fields 2] объединяем букву Q с f\_quartile;
3. Добавляем add\_date, id;

***TB\_TREATISE\_CITATION – хранит кол-во цитирований публикации***

1. В [tb\_treatise\_citation] выбраны поля нужные для таблицы tb\_treatise\_citation;
2. Избавляемся от дублей;
3. С помощью [Row normaliser 4] значения 14 полей превращаем в одно поле f\_cit\_value;



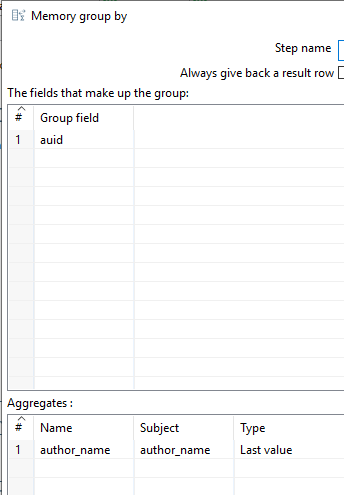
1. [Select values 13] выбирает нужные поля и f\_cit\_year переименовываем в fcittype\_and\_fcityear, т.к. он содержит в себе не только год цитирований, но и тип;
2. С помощью String operations разделяем тип и год цитирования друг от друга:



1. В [Formula] превращаем self\_cit в S(что означает цитирования с самоцитированием), а остальное в NS(noself- без самоцитирования);
2. Получаем treatise\_id, добавляем add\_date, id и загружаем в таблицу данные.

***TB\_TREATISE\_AUTHORS – хранит соавторов публикации***

1. В [Select values 16] выбираем авторов публикаций;
2. Получаем treatise\_id в [Merge join 10];
3. [Select values 17] получаем scopus id и ФИО автора;
4. У одного scopus id несколько ФИО, поэтому в [Memory group by 9] группируем по scopus id и выбираем из ФИО последнее значение;



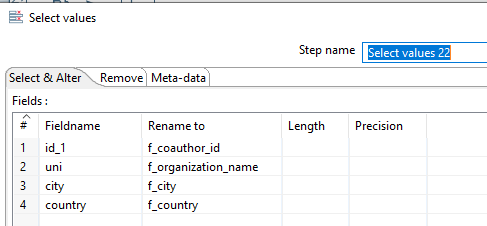
1. Далее добавляем add\_date, id;

***TB\_CO\_AUTHORS – таблица связывающая соавторов с их публикациями***

1. В [tb\_co\_author] выбираем нужные для таблицы поля;
2. Связываем публикаций с tb\_treatise\_authors в [Merge join 12] для получения id соавтора;
3. Связываем публикаций с tb\_treatise в [Merge join 13] для получения id публикаций;
4. Оставляем поля f\_subject\_id, f\_treatise\_id в [Select values 20];
5. Избавляемся от дублей;
6. Добавляем id, add\_date и загружаем в таблицу;

***TB\_INT\_COAUTH – таблица международных соавторов***

1. В [tb\_int\_coauth] выбираем поля подходящие для таблицы с их аффированностью;
2. В [данные с tb\_co\_author] выбираем поля с id соавтора;
3. Далее в [Merge join 14] будем связывать эти двнные;
4. Избавляемся от дублей;
5. В [Select values 22] переименовываем поля под названия полей таблицы;



1. В [Microsoft Excel input 3] загружаем Экзель Топ50.xlsx для заполнения поля f\_in\_top50;
2. В [Merge join 16] прикрепляем к публикациям f\_in\_top50 на основе стран и в узле [Formula 2] заполняем f\_in\_top50;
3. В [Filter rows 2] определяем заполненность полей f\_uni, f\_city, f\_country, если эти поля заполнены добавляем данные в f\_in\_top50;