

# Как создать свой набор шаблонов

В данной статье мы на примере разберём, как с нуля создать шаблон отчёта с помощью JasperReports Library, создать набор шаблонов и установить его в Платформу НЕЙРОСС.

Перед чтением данной статьи рекомендуем ознакомиться с [руководством разработчика](#).

## Оглавление

- [Общие сведения](#)
- [Создание шаблона в Jaspersoft Studio](#)
  - [Создание проекта](#)
  - [Создание шаблона](#)
  - [Выбор секций](#)
  - [Параметры](#)
  - [Источник данных](#)
  - [SQL-запрос](#)
  - [Поля](#)
  - [Внешний вид](#)
  - [Предпросмотр](#)
  - [Привязка полей ввода](#)
  - [Компиляция шаблона](#)
- [Создание набора шаблонов отчётов](#)
- [Установка набора шаблонов](#)
- [Материалы](#)

## Общие сведения

Отчёт будем строить по реляционной PostgreSQL базе данных. Для примера возьмём отчёт, в котором выведем список таблиц выбранной базы данных и размер, занимаемый этими таблицами на диске.

За основу отчёта возьмём запрос из [документации на PostgreSQL](#):

```

SELECT *
    , pg_size_pretty(total_bytes) AS total
    , pg_size_pretty(index_bytes) AS INDEX
    , pg_size_pretty(toast_bytes) AS toast
    , pg_size_pretty(table_bytes) AS TABLE
FROM (
    SELECT *, total_bytes-index_bytes-COALESCE(toast_bytes,0) AS table_bytes
FROM (
    SELECT c.oid,nspname AS table_schema, relname AS TABLE_NAME
        , c.reltuples AS row_estimate
        , pg_total_relation_size(c.oid) AS total_bytes
        , pg_indexes_size(c.oid) AS index_bytes
        , pg_total_relation_size(reltoastrelid) AS toast_bytes
    FROM pg_class c
    LEFT JOIN pg_namespace n ON n.oid = c.relnamespace
    WHERE relkind = 'r'
    ) a
) a;

```

Дополним SQL-запрос двумя фильтрами — по размеру таблицы и по имени PostgreSQL-схемы. Это позволит включить в отчёт только таблицы, занимающие на диске существенный объём, а также получить отчёт только по таблицам в выбранной схеме (чтобы, например, исключить системные таблицы в схемах `pg_catalog` и `information_schema`). Итоговый SQL-запрос выглядит следующим образом:

```

SELECT *
    , pg_size_pretty(total_bytes) AS total
    , pg_size_pretty(index_bytes) AS INDEX
    , pg_size_pretty(toast_bytes) AS toast
    , pg_size_pretty(table_bytes) AS TABLE
FROM (
    SELECT *, total_bytes-index_bytes-COALESCE(toast_bytes,0) AS table_bytes
FROM (
    SELECT c.oid,nspname AS table_schema, relname AS TABLE_NAME
        , c.reltuples AS row_estimate
        , pg_total_relation_size(c.oid) AS total_bytes
        , pg_indexes_size(c.oid) AS index_bytes
        , pg_total_relation_size(reltoastrelid) AS toast_bytes
    FROM pg_class c
    LEFT JOIN pg_namespace n ON n.oid = c.relnamespace
    WHERE relkind = 'r'
    ) a
) a
) a WHERE table_schema = 'public' AND a.total_bytes >= 10000 ORDER BY
table_name;

```

Поехали!

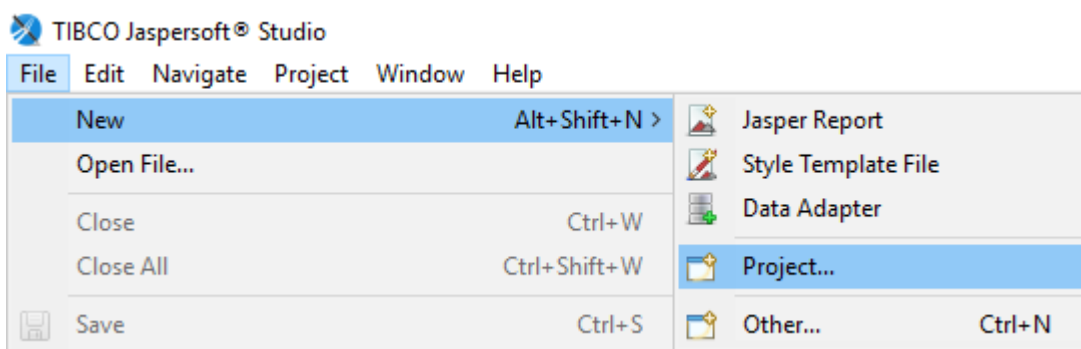
## Создание шаблона в Jaspersoft Studio

Запускаем Jaspersoft Studio (в примере используется версия 6.3.0):

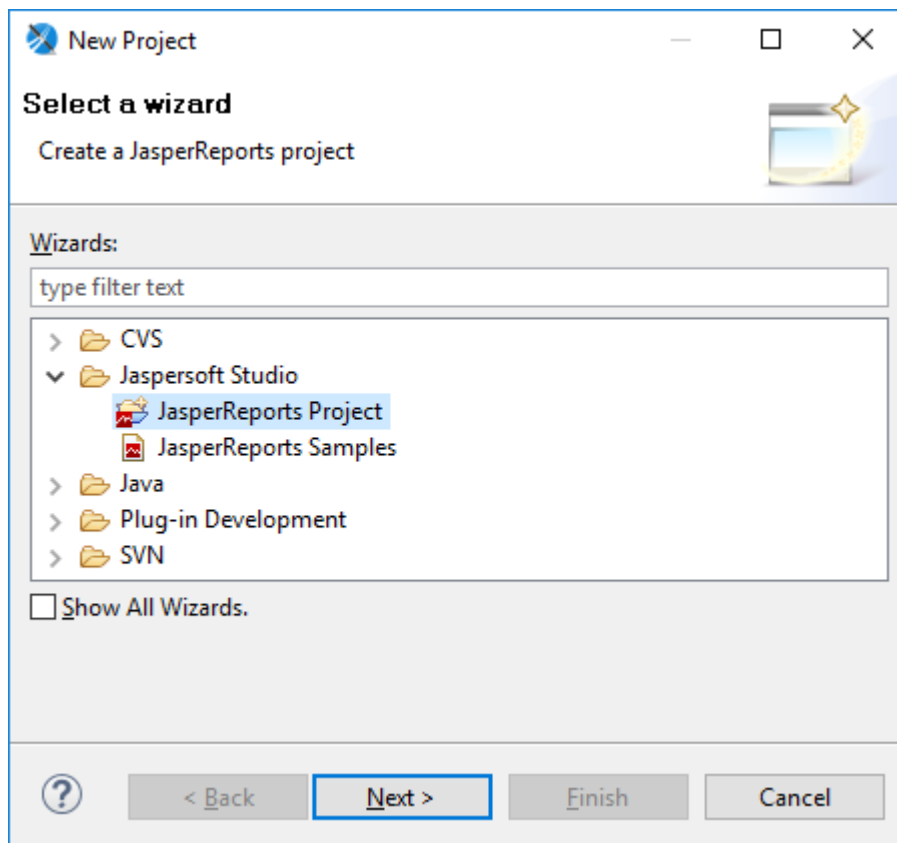


## Создание проекта

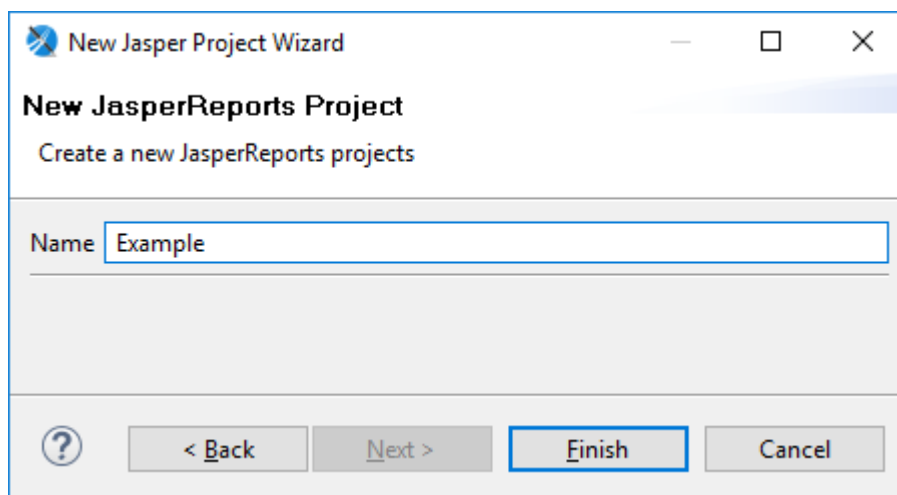
Создадим новый JasperReports проект:



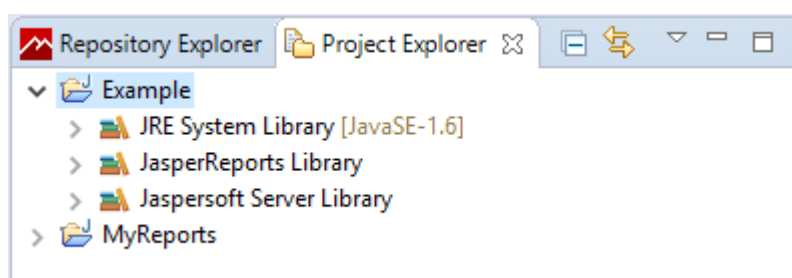
Выбираем тип проекта «JasperReports Project»:



Вводим название проекта и завершаем работу мастера:

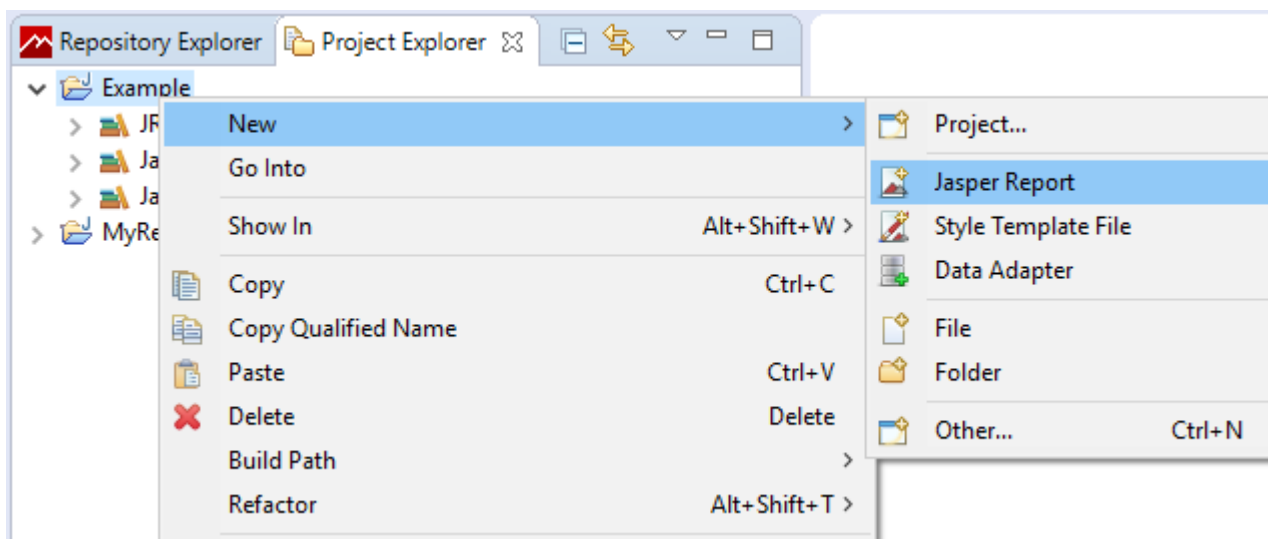


Проект создан. Его содержимое можно просматривать на вкладке Project Explorer:

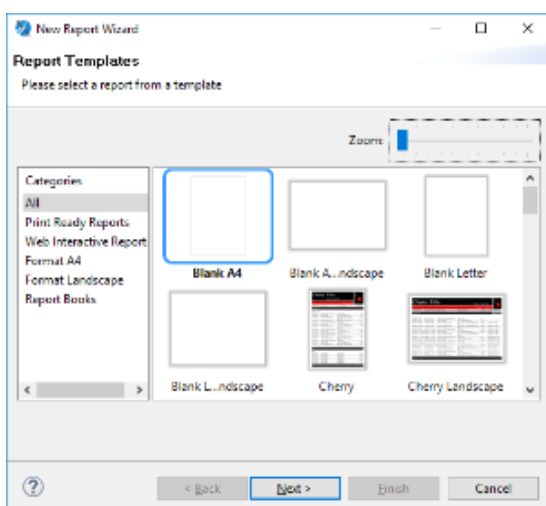


## Создание шаблона

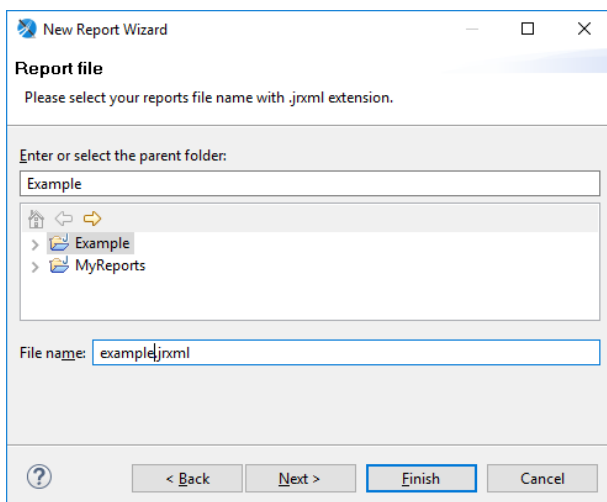
Добавим в проект шаблон отчёта. Для этого нажмите правой кнопкой мыши на элементе проекта в дереве Project Explorer. В открывшемся контекстном меню выберите New Jasper Report:



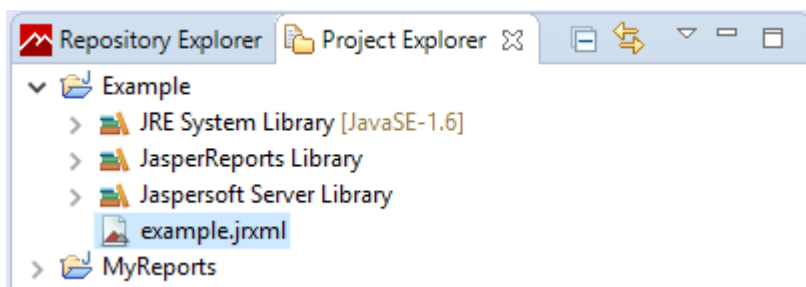
В открывшемся диалоговом окне выберите шаблон Blank A4:



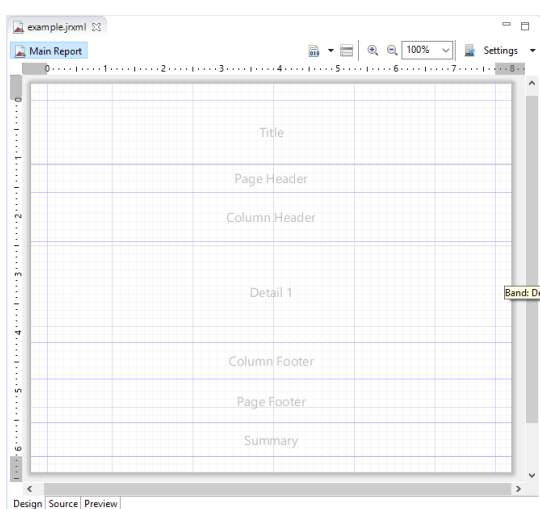
Нажимаем Next. На следующем шаге работы мастера выбираем наш проект (Example) и указываем имя создаваемого файла шаблона отчёта (example.jrxml):



На следующем шаге можно добавить источник данных (Data Source), но мы сделаем это позже. Поэтому нажимаем Finish, файл шаблона появляется среди файлов проекта.

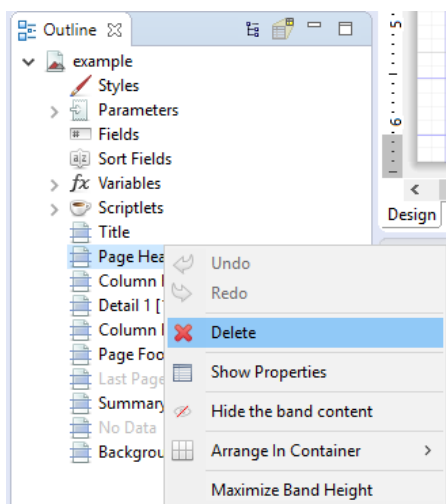


В центре окна представлен визуальный редактор нового шаблона:

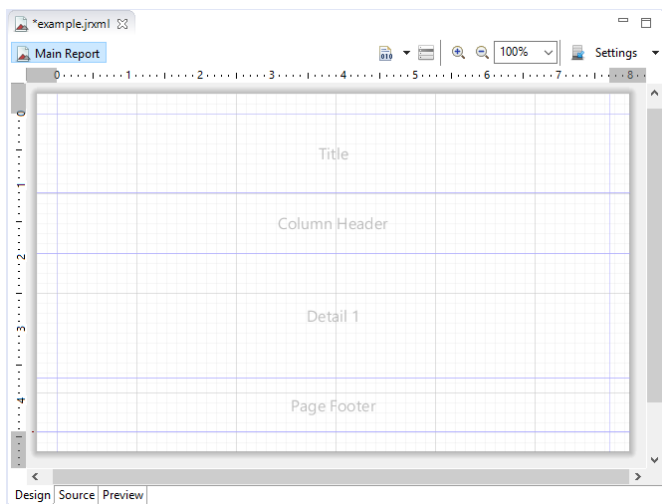


## Выбор секций

Нам потребуются не все секции, а только Title, Column Header, Detail 1, Page Footer и Background. Отключим / удалим ненужные секции на вкладке Outline. Для этого откройте контекстное меню на ненужной секции и выберите пункт Delete. И так для каждой ненужной секции:



В результате макет страницы будет выглядеть следующим образом:

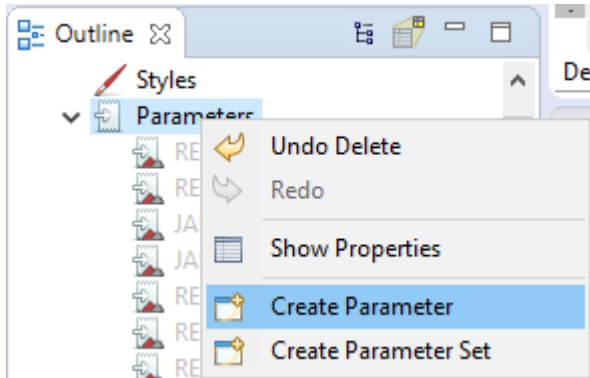


## Параметры

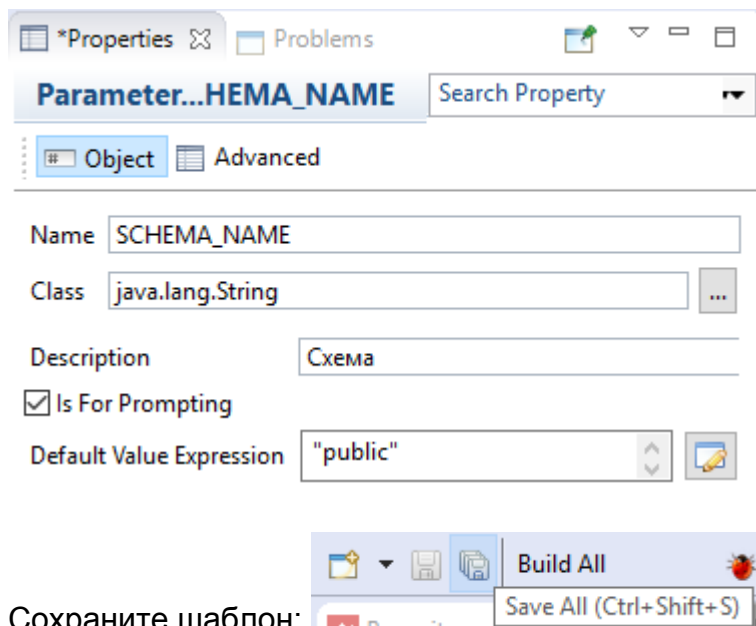
Входными параметрам для шаблона будут:

1. `SCHEMA_NAME` — строковой параметр для фильтрации по PostgreSQL-схеме.
2. `MIN_TABLE_SIZE` — числовой параметр для фильтрации таблиц в отчёте по минимальному размеру, занимаемому на диске.

Добавим соответствующие параметры. Для этого на вкладке Outline выберите в списке Parameters и создайте параметр через контекстное меню:

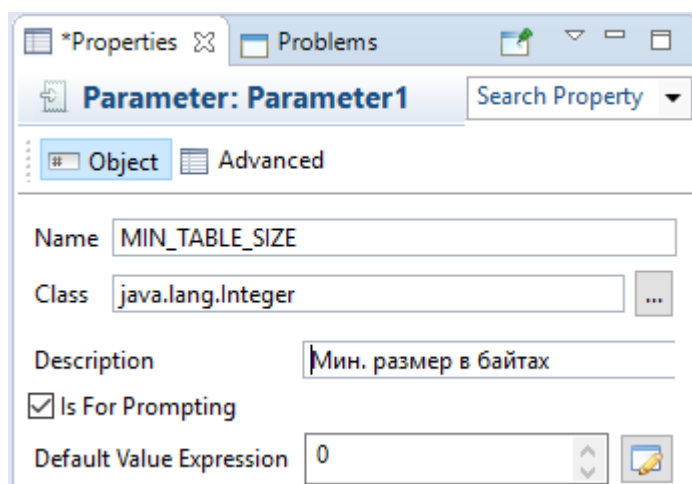


В списке появится новый параметр с именем `Parameter1`. На вкладке Properties для данного параметра укажите имя (`SCHEMA_NAME`), описание (Схема) и значение по умолчанию (`"public"`, в кавычках). Тип данных параметра (поле `Class`) оставьте без изменения — `java.lang.String`.

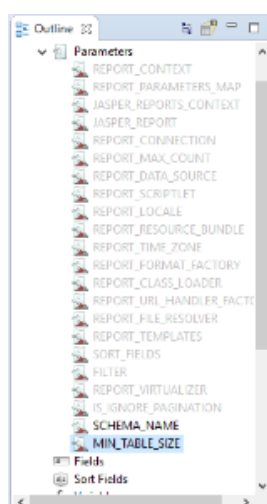


Сохраните шаблон:

Повторите процедуру для второго параметра. Только в данном случае установите тип данных параметра (Class) в `java.lang.Integer` (числовой параметр):



Сохраните шаблон. В списке параметров помимо системных теперь присутствуют два созданных нами параметра:

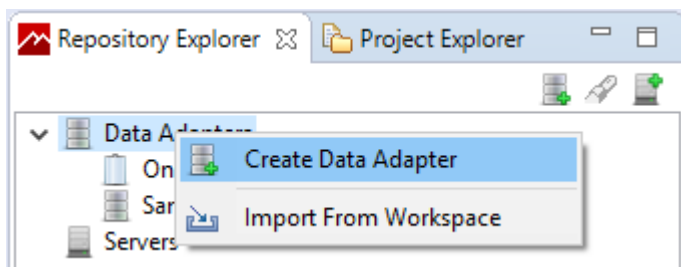




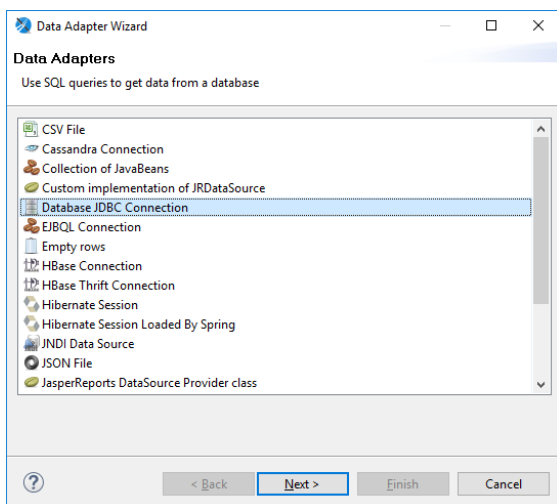
## Источник данных

Теперь добавим источник данных — мы будем работать с реляционной PostgreSQL базой данных. Это может быть любая существующая база данных. В статье используется база данных с именем ultima.

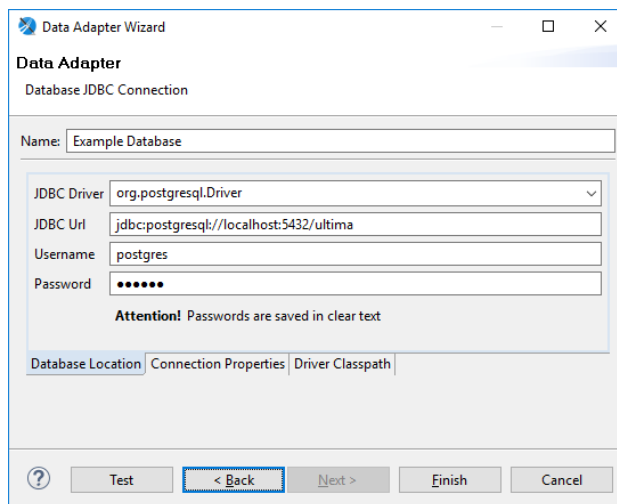
На вкладке Repository Explorer выберите Data Adapters и в контекстном меню выберите Create Data Adapter.



В открывшемся диалоговом окне выберите Database JDBC Connection.



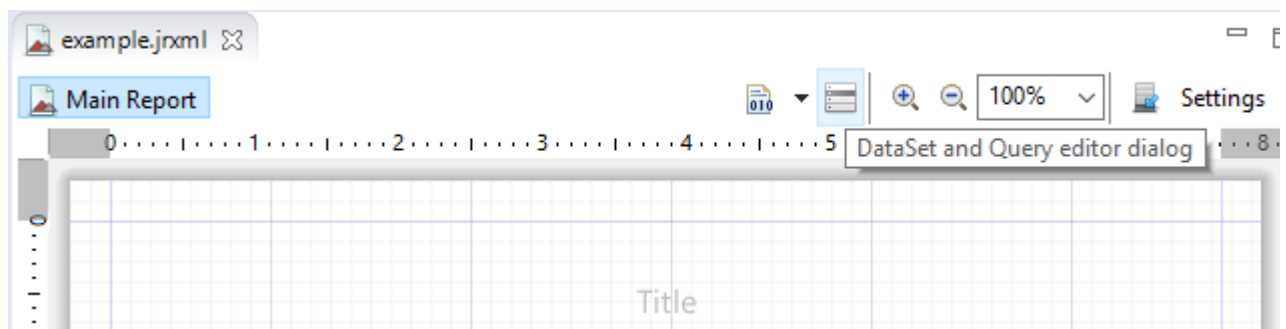
Нажимаем Next. На следующем шаге укажите название источника данных (Example Database), выберите драйвер подключения к PostgreSQL (org.postgresql.Driver), укажите корректную строку подключения к базе данных (например, jdbc:postgresql://localhost:5432/ultima для подключения к базе данных ultima на той же машине, где используется Jaspersoft Studio), укажите имя пользователя и пароль подключения к базе данных. Проверьте корректность введенных параметров — нажмите Test для проверки подключения к базе данных. После чего нажмите Finish.



Новый источник данных появился в списке на вкладке Repository Explorer.

## SQL-запрос

Теперь зададим для шаблона целевой SQL-запрос. Для этого в редакторе нажмите на соответствующую кнопку:

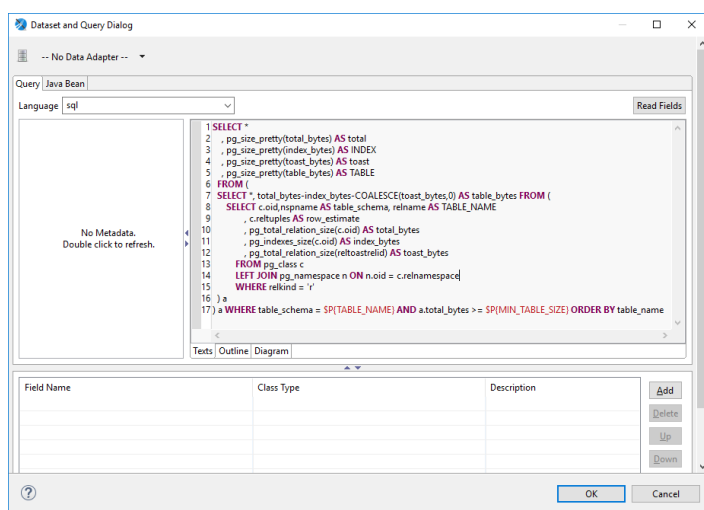


В открывшемся диалоговом окне вставьте целевой SQL-запрос:

```

SELECT *
    , pg_size_pretty(total_bytes) AS total
    , pg_size_pretty(index_bytes) AS INDEX
    , pg_size_pretty(toast_bytes) AS toast
    , pg_size_pretty(table_bytes) AS TABLE
FROM (
    SELECT *, total_bytes-index_bytes-COALESCE(toast_bytes,0) AS table_bytes
FROM (
    SELECT c.oid,nspname AS table_schema, relname AS TABLE_NAME
        , c.reltuples AS row_estimate
        , pg_total_relation_size(c.oid) AS total_bytes
        , pg_indexes_size(c.oid) AS index_bytes
        , pg_total_relation_size(reltoastrelid) AS toast_bytes
    FROM pg_class c
    LEFT JOIN pg_namespace n ON n.oid = c.relnamespace
    WHERE relkind = 'r'
) a
) a WHERE table_schema = ${SCHEMA_NAME} AND a.total_bytes >= ${P
{MIN_TABLE_SIZE} ORDER BY table_name

```

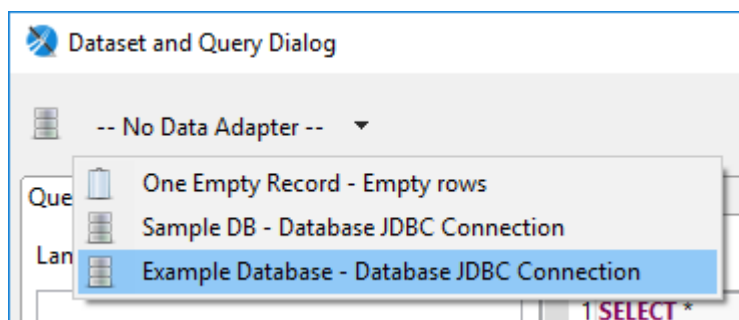


Обратите внимание, что в данном случае мы заменили константы имени схемы и минимального размера на выражения `${P{SCHEMA_NAME}}` и `${P{MIN_TABLE_SIZE}}` — таким образом в SQL-запрос будут подставлены значения созданных нами ранее параметров.

Не закрывайте диалоговое окно. Теперь мы добавим поля (fields), которые хотим вывести в отчёте. JasperSoft Studio позволяет построить список полей автоматически, выполним запрос к источнику данных.

## Поля

Выбираем в списке вверху (где написано -- No Data Adapter --) созданный нами ранее источник данных:



После чего в правом верхнем углу нажимаем кнопку Read Fields. Если источник данных указан правильно, к нему есть доступ, и SQL-запрос составлен верно, то в списке внизу окна автоматически будут созданы поля:

Field Name	Class Type	Description
oid	java.lang.Long	
table_schema	java.lang.String	
table_name	java.lang.String	
row_estimate	java.lang.Float	
total_bytes	java.lang.Long	
index_bytes	java.lang.Long	
toast_bytes	java.lang.Long	

Fields Parameters Sorting Filter Expression Data preview

В отчёт мы включим не все поля, а только table\_name (название таблицы), row\_estimate (количество строк) и total (общий размер таблицы на диске). Остальные поля можно удалить, выбрав их в списке и нажав кнопку Delete.

Также измените тип (Class Type) поля row\_estimate с `java.lang.Float` на `java.lang.Long`.

Field Name	Class Type	Description
table_name	java.lang.String	
row_estimate	java.lang.Long	
total	java.lang.String	

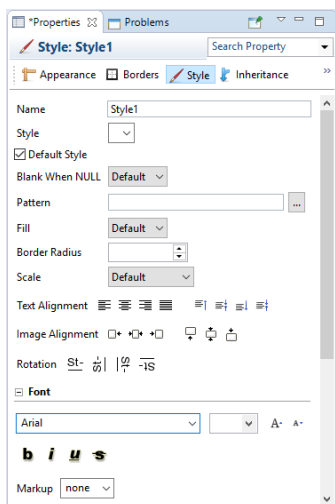
Fields Parameters Sorting Filter Expression Data preview

Нажмите ОК, чтобы применить изменения и закрыть диалоговое окно.

## Внешний вид

Для последующего использования шаблона в программе НЕЙРОСС Отчёты необходимо определить стиль по умолчанию с поддерживаемым шрифтом. На момент написания данной статьи НЕЙРОСС Отчёты поддерживается шрифт Arial.

Для добавления стиля на вкладке Outline выберите Style и в контекстном меню выберите Create Style. На вкладке Properties для созданного стиля откройте секцию настроек Style, отметьте флаг Default Style и укажите имя шрифта Arial:

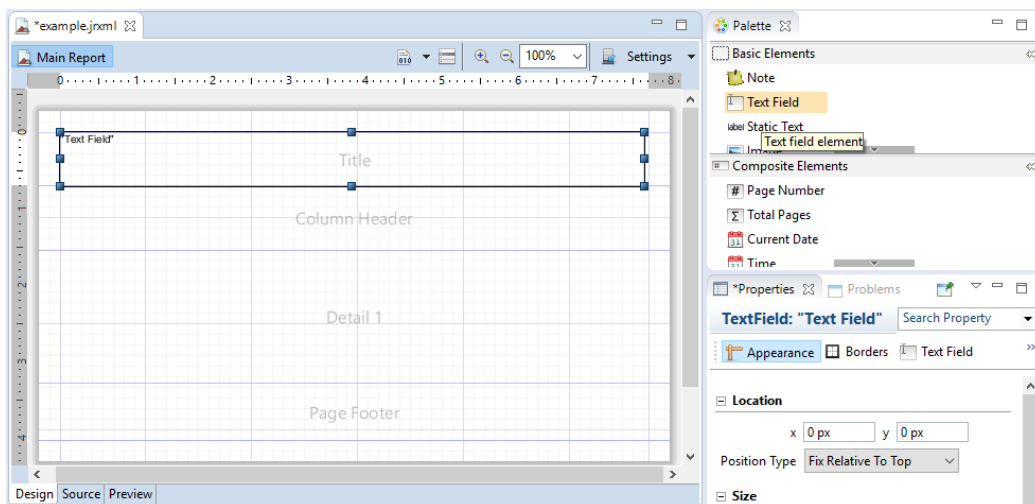


Сохраните шаблон.

⚠ Если использовать неподдерживаемый шрифт, то кириллический текст в отчёте, сгенерированном в программе НЕЙРОСС Отчёты по данному шаблону, не будет отображаться.

Теперь наполним отчёт содержимым. В визуальном редакторе разместим на макете надпись с названием отчёта, названия колонок, поля (созданные на предыдущем шаге). Дополним нижний колонтитул номером страницы.

Для названия переместите (drag & drop) с панели Palette элемент Text Field в секцию Title и измените его размер до границ полей секции по ширине и высоте:

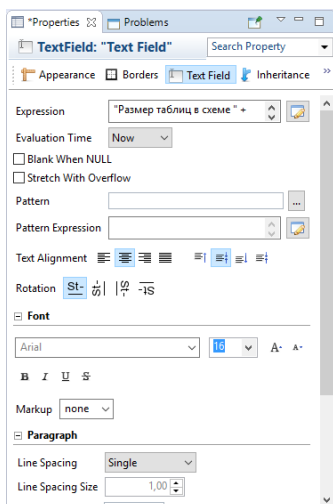


Для выбранного текстового поля на вкладке Properties в секции Text Field укажите для Expression выражение

"Размер таблиц в схеме " +  $\$P\{SCHEMA\_NAME\}$

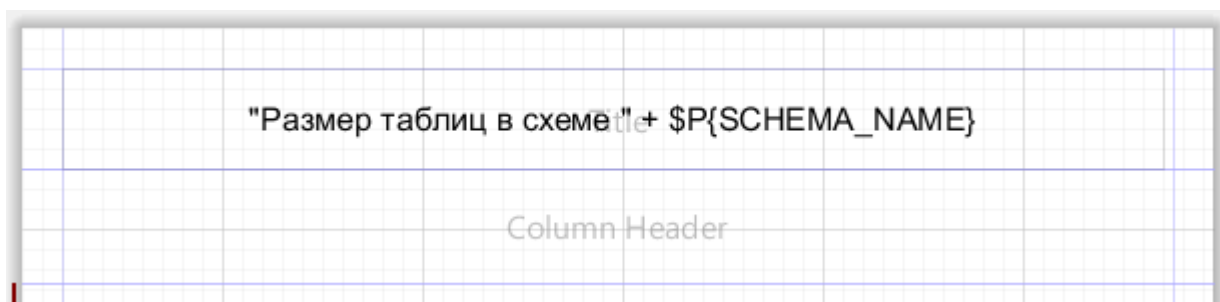
Таким образом в заголовке будет выведен текст вида «Размер таблиц в схеме public» (то есть вместо  $\$P\{SCHEMA\_NAME\}$  при генерации отчёта будет подставлено значение параметра).

Также задайте выравнивание текста по центру (по середине ширины и высоты) и установите размер шрифта в заголовке на 16:



Уменьшите высоту секции Title: выберите на вкладке Outline секцию Title, во вкладке Properties, в секции Appearance установите высоту (поле Height) в 50px.

Сохраните документ. В результате макет должен выглядеть так:



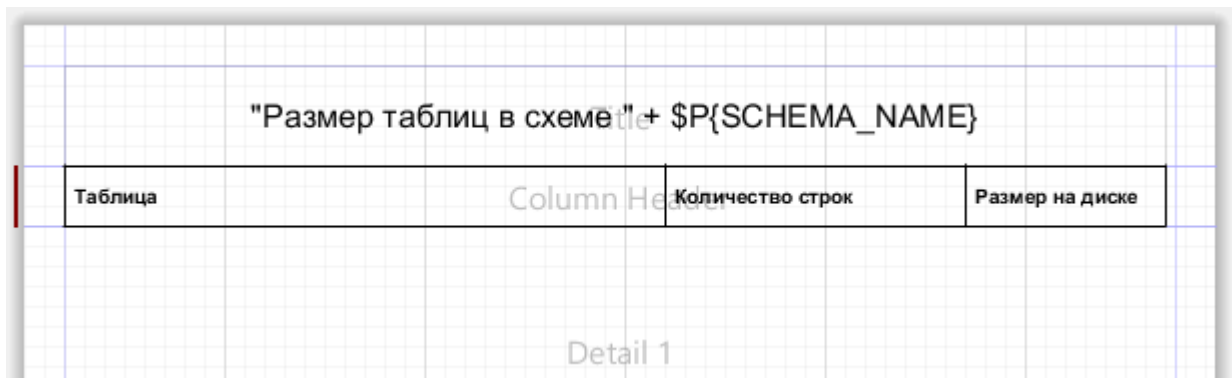
Добавим заголовок таблицы в отчёте. Перетащите из вкладки Palette три элемента Static Text в секцию Column Header:

1. Выберите первый элемент и на вкладке Properties:
  - a. в секции Appearance установите для него Location  $x=0$  px,  $y=0$  px, Size  $w=300$ px,  $h=30$ px;
  - b. в секции Borders выберите отображение всех внешних границ, Padding left=5;
  - c. в секции Static Text выровняйте текст по центру по высоте, установите жирное начертание (кнопка B), в поле Text впишите *Таблица*.
2. Выберите второй элемент и на вкладке Properties:
  - a. в секции Appearance установите для него Location  $x=300$  px,  $y=0$  px, Size  $w=150$ px,  $h=30$ px;
  - b. в секции Borders выберите отображение всех внешних границ, Padding left=5;
  - c. в секции Static Text выровняйте текст по центру по высоте, установите жирное начертание (кнопка B), в поле Text впишите *Количество строк*.
3. Выберите третий элемент и на вкладке Properties:
  - a. в секции Appearance установите для него Location  $x=450$  px,  $y=0$  px, Size  $w=100$ px,  $h=30$ px;

- b. в секции Borders выберите отображение всех внешних границ, Padding left=5;
- c. в секции Static Text выровняйте текст по центру по высоте, установите жирное начертание (кнопка B), в поле Text впишите *Размер на диске*.

Уменьшите высоту секции Column Header: выберите на вкладке Outline секцию Column Header, во вкладке Properties, в секции Appearance установите высоту (поле Height) в 30px.

Сохраните документ. В результате макет должен выглядеть так:

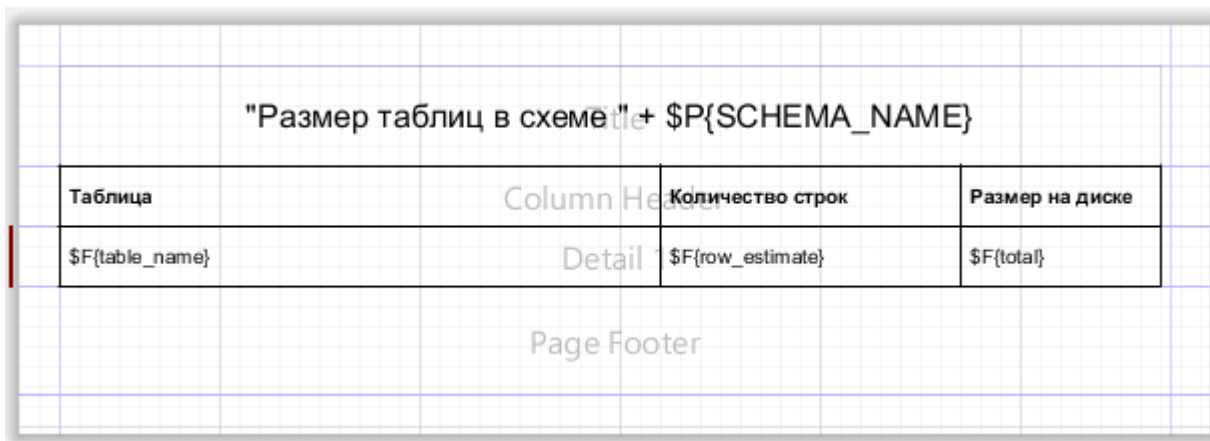


Добавим отображение созданных ранее полей в секцию Detail 1. Перетащите из вкладки Palette три элемента Text Field в секцию Detail 1:

1. Выберите первый элемент и на вкладке Properties:
  - a. в секции Appearance установите для него Location x=0 px, y=0 px, Size w=300px, h=30px;
  - b. в секции Borders выберите отображение всех внешних границ, Padding left=5;
  - c. в секции Text Field выровняйте текст по центру по высоте, в поле Expression впишите выражение `$F{table_name}`
2. Выберите второй элемент и на вкладке Properties:
  - a. в секции Appearance установите для него Location x=300 px, y=0 px, Size w=150px, h=30px;
  - b. в секции Borders выберите отображение всех внешних границ, Padding left=5;
  - c. в секции Text Field выровняйте текст по центру по высоте, в поле Expression впишите выражение `$F{row_estimate}`
3. Выберите третий элемент и на вкладке Properties:
  - a. в секции Appearance установите для него Location x=450 px, y=0 px, Size w=100px, h=30px;
  - b. в секции Borders выберите отображение всех внешних границ, Padding left=5;
  - c. в секции Text Field выровняйте текст по центру по высоте, в поле Expression впишите выражение `$F{total}`

Уменьшите высоту секции Detail 1: выберите на вкладке Outline секцию Detail 1, во вкладке Properties, в секции Appearance установите высоту (поле Height) в 30px.

Сохраните документ. В результате макет должен выглядеть так:



Добавим вывод номера страницы в секцию Page Footer (нижний колонтитул). Перетащите из вкладки Palette один элемент Text Field в секцию Page Footer.

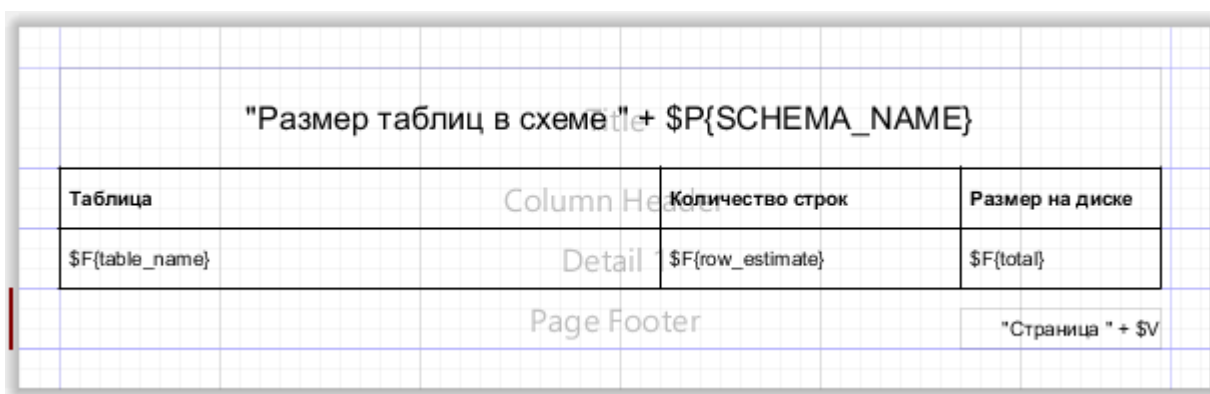
Выберите элемент и на вкладке Properties:

1. в секции Appearance установите для него Location x=450 px, y=10 px, Size w=100px, h=20px;
2. в секции Text Field выровняйте текст по центру по высоте, выровняйте по правому краю, в поле Expression впишите выражение: " " + \$V { PAGE\_NUMBER }

В данном случае для вывода номера страницы мы используем переменную **PAGE\_NUMBER**, которую автоматически предоставляет JasperReports Library.

Уменьшите высоту секции Page Footer: выберите на вкладке Outline секцию Page Footer, во вкладке Properties, в секции Appearance установите высоту (поле Height) в 30px.

Сохраните документ. В результате макет должен выглядеть так:

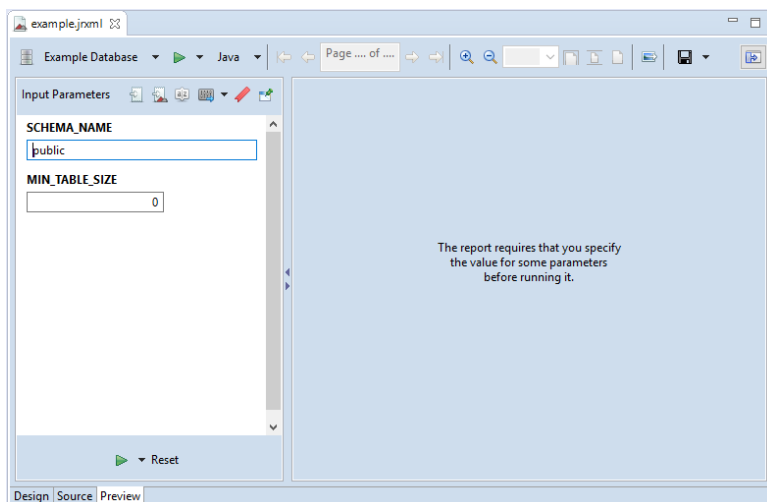


## Предпросмотр

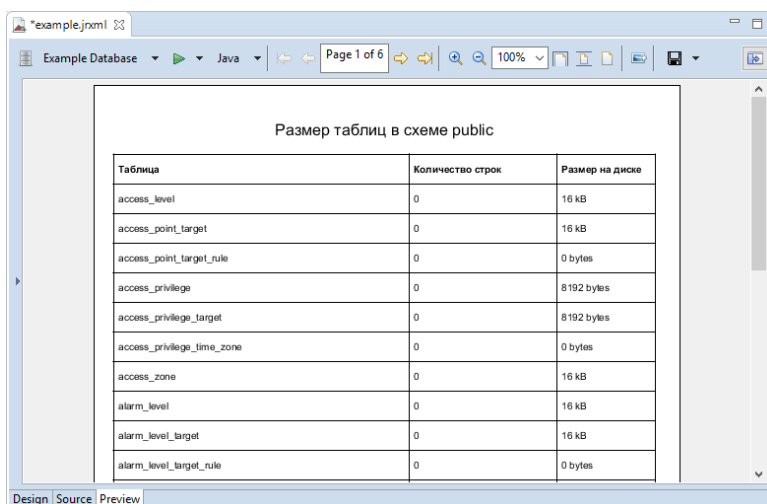
Ну что же, макет отчёта закончен. Можно посмотреть, как он будет выглядеть.

Для этого перейдите на вкладку Preview:





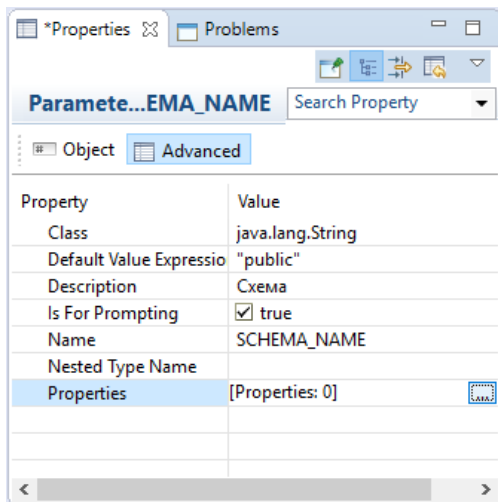
Убедитесь, что вверху в списке выбран источник данных Example Database, после чего нажмите на зелёную кнопку предпросмотра. Спустя пару секунд окне вы увидите результат:



## Привязка полей ввода

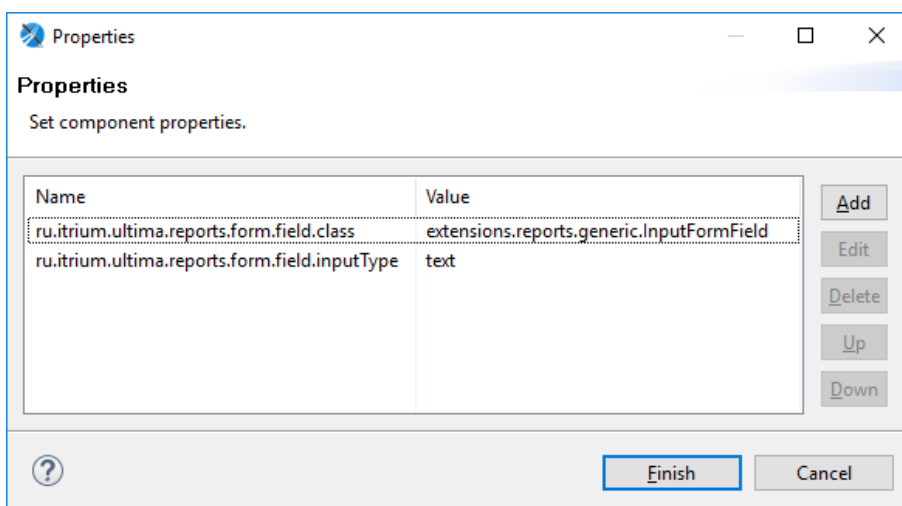
Для того, чтобы пользователь в программе НЕЙРОСС Отчёты мог задать входные параметры, необходимо к этим параметрам в Jasper-шаблоне привязать соответствующие поля ввода данных — текстовое для ввода имени схемы, числовое для ввода ограничения на размер таблицы на диске.

Выберите на вкладке Outline параметр SCHEMA\_NAME. На вкладке Properties в секции Advanced на строке Properties нажмите на кнопку «...».



В открывшемся диалоговом окне с помощью кнопки Add добавьте два свойства:

1. Property Name: `ru.itrium.ultima.reports.form.field.class` Value: `extensions.reports.generic.InputFormField`
2. Property Name: `ru.itrium.ultima.reports.form.field.inputType` Value: `text`



Нажмите Finish.

Повторите процедуру для параметра MIN\_TABLE\_SIZE. Установите для параметра следующие свойства:

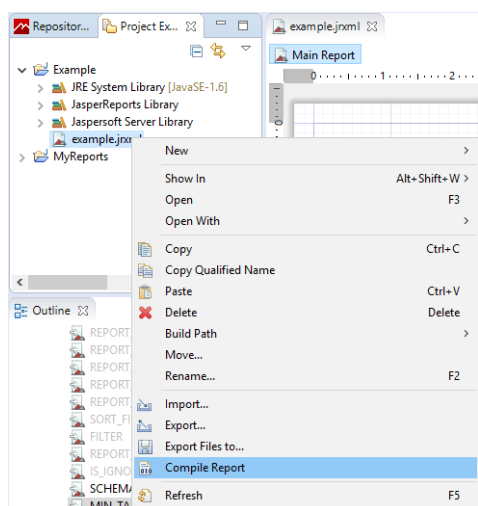
1. Property Name: `ru.itrium.ultima.reports.form.field.class` Value: `extensions.reports.generic.InputFormField`
2. Property Name: `ru.itrium.ultima.reports.form.field.inputType` Value: `number`

Сохраните шаблон.

### Компиляция шаблона

Итак, мы почти закончили редактирование шаблона в JasperSoft Studio. Осталось скомпилировать шаблон.

Для этого, выберите файл шаблона на вкладке Project Explorer и в контекстном меню выберите команду Compile:



В директории Jasper-проекта рядом с файлом example.jrxml появится файл example.jasper. Этот файл потребуется при создании набора шаблонов.

## Создание набора шаблонов отчётов

Для того, чтобы созданным шаблоном можно было пользоваться из [Платформы НЕЙРОСС](#), его необходимо упаковать в *набор шаблонов*. Набор шаблонов — это zip-архив, содержащий один или более шаблонов и соответствующие файлы-описания.

Создайте пустую директорию **exampleModule** на диске **C:\**. Это корневая директория будущего набора шаблонов.

В директории **C:\exampleModule** создайте вложенную директорию **definitions** (**C:\exampleModule\definitions**). В этой директории будут храниться шаблоны отчётов (в нашем случае — один шаблон).

В директории **C:\exampleModule\definitions** создайте вложенную директорию **exampleReport** (**C:\exampleModule\definitions\exampleReport**). Это корневая директория шаблона.

Скопируйте из Jaspersoft Studio с вкладки Project Explorer полученный ранее файл **report.jasper** (откройте на элементе контекстное меню и выберите команду Copy) в директорию **C:\exampleModule\definitions\exampleReport**.

Создайте в этой же директории файл **report.conf** со следующим содержимым:

```
definition.class=extensions.reports.GenericJasperReportDefinition
definition.htmlPaging=false
definition.supportedFormats=["pdf", "xls"]

definition.generic.key="org.example.module:exampleReport"
definition.generic.title="Размер таблиц на диске"
definition.generic.description=""
```

```
definition.jasper.design = {  
  main: "example.jasper",  
  html: "example.jasper"  
}
```

Создайте файл-определение набора шаблонов `deployment.conf` в директории `C:\exampleModule` следующего содержания:

```
deployment.key=org.example.module  
deployment.title="Новый набор шаблонов"  
deployment.version=0.1.0
```



### ВНИМАНИЕ

Текстовые файлы `report.conf` и `deployment.conf` должны быть созданы в кодировке UTF-8. В противном случае в интерфейсе Платформы НЕЙРОСС соответствующий текст может отображаться некорректно.

Перейдите в проводнике Windows в директорию `C:\exampleModule`, выберите файл `deployment.conf` и директорию `definitions` и в контекстном меню выберите пункт **Отправить Сжатая ZIP-папка**.

В директории `C:\exampleModule` появится новый zip-файл. Измените его имя на `exampleModule-0.1.0.zip`.

Это файл набора шаблонов отчётов. Его можно загрузить в Платформу НЕЙРОСС.

## Установка набора шаблонов

Осталось установить созданный набор шаблонов в Платформу НЕЙРОСС.

Для установки набора шаблонов откройте интерфейс Платформы, авторизуйтесь и перейдите в раздел «Отчёты Наборы шаблонов отчётов».

Выберите файл набора шаблонов и нажмите кнопку «Загрузить». В таблице установленных наборов шаблонов должна появиться новая запись с названием «Новый набор шаблонов» (это название набора шаблонов из файла `deployment.conf`):

## Источники данных

Загрузить модуль

Загрузите модуль с формами отчётов (zip-архив).

Файл:  Файл не выбран

Загрузить

## Модули отчётов

Название	Версия	Действия
Новый модуль отчётов	0.1.0	<a href="#">Удалить</a>   <a href="#">Выгрузить</a>
ITRIUM: учёт рабочего времени	0.1.0	<a href="#">Удалить</a>   <a href="#">Выгрузить</a>

Вернитесь в начало и добавьте подключение к PostgreSQL базе данных в разделе «Источники данных».

Перейдите в раздел «Отчёты».

Выберите в списке источник данных (подключение в PostgreSQL базе данных). В списке доступных шаблонов должна присутствовать опция «Размер таблиц на диске» (это название шаблона из файла [report.conf](#)). Выберите этот шаблон.

Программа отобразит в интерфейсе форму ввода параметров с полями для ввода имени схемы и ограничения по размеру таблицы.

Укажите требуемые параметры и сформируйте отчёт.

[illegible]

PDF-файл: [Размер таблиц на диске \(16-09-2018 15-58\).pdf](#)

## Материалы

Пример набора шаблонов, созданного в соответствии с приведённой инструкцией, доступен [по ссылке](#).

Архив также содержит исходный код Jasper-шаблона в директории `definitions\exampleReport` — это файл `example.jrxml`.