

Create Report

by [OSOE Project](#).

▼ Details

SQL & Python Report Wizard

- Invoke SQL Shell
- Create a SQL Report
- Make Dynamic SQL Methods
- Invoke the Report
- Invoke Python Shell
- Create a Python Report

▼ Details

The SQL Shell

The SQL Shell

▼ Details

The first part of this tutorial is about creating a report from the SQL shell. The idea of the SQL shell is to let you type any SQL query you wish to, then to create report from it if the result fits you.

First you must go to the SQL shell thanks to the '**SQL shell**' menu in **My Favourites**'.

The first SQL query you might want to try is '**SHOW TABLES**'

It will display all MySQL tables used within the ERP5 instance.

Once you wrote the SQL query correctly, you must click on Update to see the result. If it gives an error, you must go back, to correct your query. If you query is correct, you will see the result behind the shell.

ERP5 “catalog” Table

▼ Details

The '**catalog**' table is one of the table on which you will work a lot for report since it indexes every documents in your ERP5 instance.

To get familiar with this table, you can type in '**EXPLAIN catalog**' then click on '**Update**'.

You can then call this function on each table, so you can get an overview of all data available in the tables.

ERP5 Table Structure

▼ Details

Here is an example of the structure of ERP5 table. ERP5 doesn't have so many tables, thanks to its unified business model. But anyway, a good knowledge of the table structure is required to be able to create complicated reports.

ERP5 Background Indexing

▼ Details

'SHOW FULL PROCESSLIST' shows you which threads are running. You will see the process related to the running ERP5 instance.

Design Report SQL Code

▼ Details

Once you are familiar with the table structure and content, you can create you first SQL report. In our example we want to

create a report which shows us how many documents we have in the database:

```
SELECT portal_type, count(uid) as portal_count
```

```
FROM catalog
```

```
GROUP BY portal_type
```

```
ORDER BY portal_count DESC
```

We can see that on a fresh ERP5 instance on TioLive, the type of documents we most have is the 'Category' one. Categories are automatically defined during the configuration. We will improve our report so we only see 'Final User' documents. We will not consider the objects which are not interesting:

```
SELECT portal_type, count(uid) as portal_count
```

```
FROM catalog
```

```
WHERE portal_type NOT IN ("Category", "Email Thread", "Base Type", "Base Category", "Business Template", "Role information", "Action information", "Active Process", "Trash Bin")
```

```
AND portal_type NOT LIKE "Cache%" AND portal_type NOT LIKE "%Builder" AND portal_type NOT LIKE "%Group" AND portal_type NOT LIKE "%Tester"
```

```
GROUP BY portal_type ORDER BY portal_count DESC
```

Invoke the Report Wizard

▼ Details

Once the SQL query gives us a good report, we will use the 'Create SQL Report'.

We will have to choose few options:

Report Title: which is the human readable title of the report

Report ID: which is the unique ID for the report. It is used to name forms created automatically by the wizard. It must respect naming convention.

Report Columns: tells which data will be put in which column.

Selected Skins ID: tells where will be created the forms for the report within the portal_skins folder. In development time you must select custom.

Portal Type: tells from where will the report be available Portal type are all available type of documents in ERP5. In our case we will chose '**Business Template**' so our report will be available in the business template module. We could have chosen any other module.

Once everything is well selected, you can click on '**Create ERP5 Report**'

Three documents will be created in the custom folder:

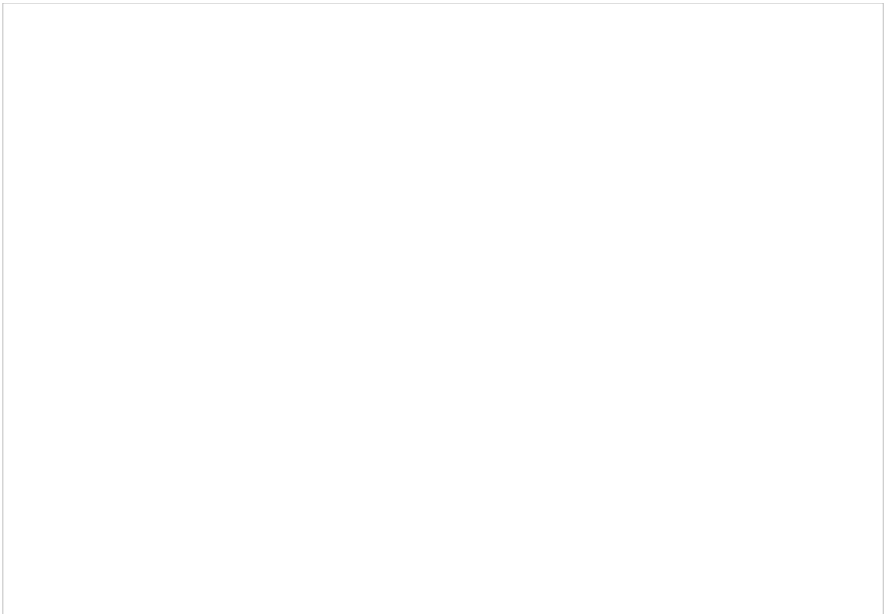
- BusinessTemplate_view<REPORTID>Report: which is an ERP5 Form contains the listbox associated with the report.
- BusinessTemplate_view<REPORTID>ReportDialog: which is an ERP5 Form containing the dialog option for the report
- BusinessTemplate_zGet<REPORTID>List: which is the SQL method

Define Input Dialog

▼ Details

On BusinessTemplate_view<REPORTID>Report, you can change many options of your report.

Make SQL Method Dynamic



▼ Details

BusinessTemplate_zGet<REPORTID>List you can change the SQL query.

Invoke Report

▼ Details

In order to display the report, you must first go to the module to which you linked the report, in our case the business template module.

Then you click on the report icon. On the dialog box you can select the report we just created, then click on 'print' report.

Finally the report is displayed by ERP5.

The Python Shell

▼ Details

We can do the same than we just did but with python scripts. It will allow you to create even more complex reports very easily thanks to python. You might first want to take a look at the python introduction.

As we did with SQL query, you can write your python script, then click on update to see the result of it:

```
kw={ 'portal_type': 'Person', 'owner': '%', }  
return context.portal_catalog(**kw)
```

Result: each person whos owner is everyone (*)

```
kw={ 'portal_type': 'Person', 'owner': '%', 'src__': 1, }  
return context.portal_catalog(**kw)
```

Result: gives the SQL method associated with the python script.

```
kw={ 'portal_type': 'Person', 'role_title': 'Internal', 'src__': 1, }  
return context.portal_catalog(**kw)
```

Result: every person which role is 'Internal'

Design Python Reports

▼ Details

So once we created a good python request, we can create the report as we did for SQL query.

Invoke the Report Wizard

▼ Details

We will again have to choose few options:

Report Title: which is the human readable title of the report

Report ID: which is the unique ID for the report. It is used to name forms created automatically by the wizard. It must respect naming convention.

Report Columns: tells which data will be put in which column.

Selected Skins ID: tells where will be created the forms for the report within the portal_skins folder. In development time you must select custom.

Portal Type: tells from where will the report be available Portal type are all available type of documents in ERP5. In our case we will chose '**Business Template**' so our report will be available in the business template module. We could have chosen any other module.

Once everything is well selected, you can click on '**Create ERP5 Report**'

Three documents will be created in the custom folder:

- BusinessTemplate_view<REPORTID>Report: which is an ERP5 Form contains the listbox associated with the report.
- BusinessTemplate_view<REPORTID>ReportDialog: which is an ERP5 Form containing the dialog option for the report
- BusinessTemplate_zGet<REPORTID>List: which is the SQL method

Define Input Dialog

▼ Details

The dialog form let you create options for your reports (ex: you want to display all support request assigned to one person)

Make Python Script Dynamic

▼ Details

Even after invoking the report wizard, you can change your python script.

Invoke Report

▼ Details

In order to display the report, you must first go to the module to which you linked the report, in our case the business template module.

Then you click on the report icon. On the dialog box you can select the report we just created, then click on 'print' report.

Finally the report is displayed by ERP5.