Create an OLIB report

Learn how to create an OLIB report.

**OLIB Reports Exchange – creating a new report by retrieving it from the OLIB Reports Repository**

**Get the URL**

If you have requested a specific report, you will be advised of the URL which you can then use to retrieve the definition from the OLIB Reports Repository. If you have identified the report yourself from the repository, you can copy the URL directly from there.

For example, to retrieve the **CA013c – sortable Accessions List (for exporting to Excel)** from the repository, copy the link from the Right-click and copy link location to get this report hyperlink:

Then paste it into the Definition URL field in your new report, as described in **Creating the report** below. Alternatively, the URL may be advised directly by your regional support desk in response to a support call.

**Create the report**

Once you have copied the report’s URL to your clipboard, you can use it to download the report itself into your OLIB system as follows:

In OLIB Web, on the Reports domain hitlist (not a report record display), click **New Record** to get:
Paste the report’s URL into the **Definition URL** field and click **Save and Continue**.

OLIB will then retrieve the report from the repository and it will be ready to parameterise, schedule, preview, run, etc.

Note that, for subsequent reference, the Definition URL will be saved along with the report definition.

Note also that OLIB systems where the OLIB server cannot connect to the OLIB Reports Repository will be able to “download” new reports using the **Raw Definition** field. Instructions for this will be provided when it is required to use this facility.

If the retrieved report has a combination of parameter types that is not compatible with any existing report in your system, then a new layout will be automatically generated for this report type when the report is downloaded from the repository. This layout is not immediately available, so the new report will continue to be displayed using the New Report layout, and the following message will be displayed:

```
The report has been added to your system. Please log out and log back in again to use it.
```

If this happens, simply log out and log back in again, then re-display the new report, and it will be displayed using the new layout.

**Update a report**

It is possible that the requirements for a report change over time, or a bug is found in the report. In such cases, the report will be modified or corrected in the OLIB Reports Repository, and you will be notified that a revision has been made to the report. When you receive notification of updates to existing reports, you can update your copy of the report by re-downloading the report’s definition using the **Refresh Definitions** action on the OLIB Reports hitlist:
This assumes that the Definition URL field on the Report Definition tab contains the correct URL for that report (https://reports.olib.oclc.org/reposi...&param1=nnnn, where nnnn is the report's number). The standard reports are delivered with the correct URL in this field. The contents of the Definition URL field should only be modified if you want to download a report's definition from a different OLIB system.

The Refresh Definitions action can be used on multiple reports at a time.

**Write a report from scratch**

**Get started**

When creating a new report that is not available to retrieve from another system, click New Record from the Reports domain hitlist, and then select the Parameter Type. This will cause the layout to change to the appropriate layout for the parameter type you selected.

If a suitable parameter type is not available or is not known at this point, then select the generic Run Report option. This type of report has the maximum number of optional free text parameters and could be used to run the report if required.

Once the new report screen is re-displayed, enter the name of the report in the Report Name field on the Run Report tab, and then move to the Report Definition tab to begin configuring the new report.

Tip: if you select a parameter type that includes one or more mandatory runtime parameter fields, you will not be able to move to the Report Definition tab until you have entered something in these fields. At this stage, it doesn't matter what you enter in these mandatory fields, so just enter a short random string so that the fields contain something so that you can switch to the Report Definition tab.

**Report Definition fields**

There are three groups of report definition fields:

- Report description fields
- Report parameters
- Report SQL

Each group is described in more detail in the following sections.
Report description fields

Enter a description for the report, plus the domain and (optionally) a location (if it is a location-specific report), a created date and a last updated date. Finally, if you want to allow the report to be run as an anonymous GETStream report, set Allow Anonymous GET to Yes.

Define the new report's parameters

<table>
<thead>
<tr>
<th>Report Parameters</th>
<th>Parameter Field Label</th>
<th>Parameter Type</th>
<th>Helper</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameter 1</td>
<td>Accession Date from</td>
<td>Date (Mandatory)</td>
<td></td>
</tr>
<tr>
<td>SQL / Date Format</td>
<td>DD-Mon-YYYY</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Parameter 2</td>
<td>Accession Date to</td>
<td>Date (Mandatory)</td>
<td></td>
</tr>
<tr>
<td>SQL / Date Format</td>
<td>DD-Mon-YYYY</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Parameter 3</td>
<td>Locations</td>
<td>Multipick (Mandatory)</td>
<td></td>
</tr>
<tr>
<td>SQL / Date Format</td>
<td>SELECT KeyValuePair(loc_id, locid) FROM locs WHERE loc_id = '0000' ORDER BY UPPER(loc_id)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Each parameter has 3 fields for defining the label, the parameter type and a helper (a drop-down list) to facilitate the formatting of the parameter value. There is also a fourth field to aid the completion of the SQL for populating the drop-down lists, or, if the parameter is a date, to define the date format.

1. **Field label**

The first field for each parameter is the field label. This text will appear on the Run Report sheet against the respective parameter selection field so that the operator knows what values to enter. This may be, for example, User Location, Copy Location or Transaction Location.

The second field is a drop-down to specify the parameter type and whether or not the parameter is mandatory.

2. **Parameter type**

- **Combo**

This is a drop-down parameter that will produce a single value in this position in the parameter list. This is also the method for delivering a *Rolling Date* value.

There are *combo* options for selecting the date at the start or end of a calendar (holiday) from the *User Management Reference Data / Calendar* domain – *Holiday Start Dates* and *Holiday End Dates*. These enable you to readily include term start and end dates and/or academic years for reporting purposes.

Caution is required, however, to ensure that the format of the date returned by such SQL is compatible with the date format recognised by the SQL in the report. For further information on this, see [OLIB report sections](https://help.oclc.org/Library_Management/OLIB/Reports/Use_OLIB_reports/030Create_an_OLIB_report) (Adding Parameters).

- **Date**

This will present the parameter selection field as a Date-Picker, and place the value selected into the parameter value using the format given in the SQL / Date Format field for this parameter.
In the SQL, this can be converted back to a date using, for example:

```sql
AND trandate BETWEEN to_date(#3#, 'DD/MM/YYYY')
    AND to_date(#4#, 'DD/MM/YYYY')
```

When referenced in the SQL in this fashion, ensure that the date format for the parameter matches the format used in the SQL, example in red above, so that the value is recognised correctly.

For rolling date parameters and holiday start/end date parameters, the parameter should be a combo type, not a date.

- **Multipick**

This is a drop-down parameter that will produce a comma separated list of values in this position in the parameter list. Such parameters can be handling using the `csv()` function described below.

- **Text**

This is a free-text parameter. It is also used for values such as barcodes and title numbers. Note, however, that it is possible to present a list of titles for the user to select the title number by using a Combo or Multipick parameter and writing an SQL SELECT statement to select and sort the relevant titles, users, etc.

3. **SQL / Date Format helper**

The third field for each parameter is a drop-down list to assist with the population of the fourth and final field. For Combo and Multipick parameters, this provides a set of SQL SELECT statements suitable for choosing commonly used values for a report. For Date parameters, this enables you to specify the date format. The SQL can be modified manually after the SQL / Date Format field is populated. For rolling dates, the date format here must match the date format required by the query.

There are several such pieces of SQL that exclude internal or no longer used values to avoid confusion for the operator running the report.

4. **SQL / Date Format**

This field is used to provide one of several features:

1. The SQL to execute to populate the list of values in the drop-down list for this parameter. Note that this must return a specific `TYPE` as given by the predefined SQL. For example:

   ```sql
   SELECT `KeyValuePair`(copycat , copycatld ) FROM copycats
   ORDER BY UPPER(copycatld )
   ```

2. The SQL to provide a rolling date value:

   ```sql
   SELECT `KeyValuePair`
   REPLACE(ref_desc, '{dateformat}', 'DD/MM/YY'), ref_key
   )
   ```
FROM wod_reference
WHERE ref_type = 4018 ORDER BY UPPER(ref_key)

When using this, the date format in red must be amended as necessary to match the format required by the query.

3. The date format to use for passing the entered value to the SQL. Some commonly used date formats are selectable as examples:

4. A list of hardcoded values in “key=value” style, separated by semicolons. For example:

L=Location; U=User; D=Department

With this, the user will be presented with a drop-down list containing:

Location
User
Department

When selected, the parameter value to be passed to the SQL will be L, U or D respectively.

After defining (or amending) the parameters

Once the parameter types have been defined, the appropriate report type will need to be selected in the Report Type field immediately below the Description field further up the screen. If an appropriate report type does not yet exist, it and its corresponding layout will need to be generated. This can be done from the Reports hitlist by selecting the report and executing the Generate Layout action.

This will advise if any new layouts have been created. If new layouts have been created, then you will need to log out and back in again to use them.

If no new layouts have been generated, a simple refresh may be needed to ensure that the correct layout is used.

Define the report’s SQL

The SQL for the report is entered in one of three ways, depending on the type of output you require:

• A flat list, e.g. to list a selection of titles, copies or users, or to provide raw circulation transaction data for input into
Excel – see **List/Excel output**

- A multi-level report for simple output to HTML – see **Report Lines for HTML Output**
- A multi-level report for styled output to HTML, Word or PDF – see **Report Sections for HTML, PDF or Word output**

In all cases, parameter values are referenced in the SQL by specifying the parameter number with a “#” either side. For example, “#3#” for the value from parameter 3.

For a parameter that is a list of values, these will be separated by commas. A function has been incorporated into OLIB Reports to simplify the processing of such parameters.

For example, a mandatory list of copy categories in parameter 2 can be processed with:

```
AND copycat IN csv(#2#)
```

If the parameter is optional, this must be replaced with:

```
AND (#2# IS NULL OR copycat IN csv(#2#))
```

A date parameter value is passed in as plain text using the configured format for that parameter. These can be processed by converting the value to a date, for example:

```
AND trandate < TO_DATE(#3#, 'DD-MM-YYY')
```

**List/Excel output**

To produce a simple, comma-separated file from an SQL statement, the SQL statement can be written in the **SQL To Run** field. This will automate much of the processing required to present the results in a format that Microsoft Excel will interpret readily.

If used as a GETStream report, it will generate a tab separated or HTML output. If emailed, it will be a comma-separated or HTML file attached to the email or an HTML formatted message depending on the Output Type selected. Excel will handle the results in the same way by presenting the data in the intended columns.

If HTML output is chosen, OLIB will automatically place each record into its own HTML table row and each result column into an HTML table cell. Numeric columns will be right justified. If no header is specified, then a header will be generated from the column names generated from the SQL To Run statement. The generated header will start with “<table>”. Similarly, a “</table>” will be added at the end.

If control over the HTML produced is required, then the SQL should contain:

```
/* html */
```

This will inform the report generation code that there is no need for any automated HTML table generation. The code will expect the HTML to be manually coded in the SQL itself.

The following SQL statements will generate the same HTML, if HTML style output is chosen:

---

https://help.oclc.org/Library_Management/OLIB/Reports/Use_OLIB_reports/030Create_an_OLIB_report
Printed: Thu, 24 Mar 2022 12:47:14 GMT
### Automated HTML

```sql
select titleno, title
from titles
where titleno in csv#{1#}
```

### Manual HTML

```sql
/* html */
select '<tr><td align="right">' || titleno || '
</td><td>' || nvl(htmlescape(title), ' &nbsp;') || '</td></tr>'
from titles
where titleno in csv#{1#}
```

The former may also generate comma separated output if chosen.

An Excel template can be attached to such reports. When a template is present, the URL for retrieving the data will be updated by the GETStream URL button.

**Report Lines for HTML output**

This functionality remains available for backward compatibility and can be used to produce reports that generate HTML, but not PDF or Word. This is documented elsewhere (OLIBReportsReportLines.docx).

**Report Sections for HTML, PDF or Word output**

For more complicated reports, and if output to PDF or Word is required, the Report Sections feature is recommended. This allows for the greatest flexibility of output options. It is described in detail in the following section.

(Notahe that the Microsoft Word output must be opened by the Word module of Microsoft™ Office 2007 or later. Tests have shown that these outputs currently cannot be viewed by Open Office, Google Docs or LibreOffice.)