Query2Report

Table of Contents

Overview

How many times have we run SQL queries on RDBMS using rich SQL editors to extract data, copy it to excel sheet to plot simple trend charts? Why do we not use any of the existing BI or reporting tools?

The overhead of using heavy weight BI tools is little too much for a simple report/charting requirement. Firstly, you need to have a licensed version, trial versions are good but they don't offer continuity. Secondly, there is a learning curve associated in building reports, no one lets you transform SQL queries to chart directly and you need to know the product in building even a simple report. Query2Report addresses both these limitations.

Introduction

Query2Report is a simple, light weight web based reporting solution that lets you map SQL queries to beautiful reports using google charts.

It is simple because there is no learning curve required to build reports. Any person with descent SQL knowledge should be able to build report quickly.

It is light weight because it doesn't require users to install any thick client, your web browser acts like a thin client to build and view reports and also to administrator the product.

Prerequisite

The application is a web application hence Tomcat Application Server v8.0 or above should be installed on the server where you will be hosting Query2Report web application.

The application uses JDBC to connect to the database hence licensed version of JDBC Driver(s) to connect to database(s) are required.

Installation

Download the q2r.war file from https://sourceforge.net/projects/query2report/ and place it under CATALINA_HOME/webapps/ directory and restart the application server.

Access the application using below URL,

http://<hostname>:<port>/q2r

Where, port -> port on which tomcat is listening

The default username/password to connect to the database is admin/admin. It's highly recommended to change the default password for the admin user once logged in for the first time.

Users and Roles

The application supports 3 roles which are

1. Administrator
2. Viewer
3. Guest

The "Administrator" role entitles user to perform all admin tasks which involves

- Creating users
- Registering JDBC drivers
- Creating Data Source
- Creating, Editing and Viewing Reports in "Public Reports" and "Personal Reports" folders

The "Viewer" role entitles users to

- View reports in "Public Reports" folder
- Creating, Editing and Viewing Reports in "Personal Reports" folder

The "Guest" role entitles users to

- Viewing reports in "Public Reports" folder

The application has a default “Administrator” user called “admin” with default password of “admin”. It’s highly recommended to change the default password for the admin user once logged in for the first time.

The application entitles users with “Administrator” role to add new users. New user can be added by clicking on **“Users” -> “Add User”**, this will open up add user dialog where you need to specify

- Display name
- Username
- Password
- Role

Users can access their profile and update when required by logging into q2r web application and clicking on user icon on the right of the top menu bar and clicking on **“Update Profile”**

As part of the profile, user can also specify the session timeout in seconds. The default value is 10 minutes after which the user is logged out. This is an important property that admins must consider if they are building auto refresh report, you don’t want user session timeout to happen while having auto refresh reporting being open. If you have reports/dashboard with auto refresh consider having 1 day as session timeout.

Registering JDBC Driver

The application uses JDBC to connect to database and run queries. Administrators should register required JDBC driver with Q2R if the enterprise is a heterogeneous database environment.

The drivers can be registered by clicking on **Drivers** -> **Add Driver** on the Q2R web console, this opens up a registration dialog where you need to specify

1. Driver Alias
2. Full class path for the JDBC driver
3. Path to jar file on local system

On saving the driver, the driver jar file will get uploaded from local system (Where the browser is launched) to the server hosting q2r web application. The jar file will be copied to CATALINA_HOME/webapps/q2r/WEB-INF/lib folder.

Once the save is successful, as directed by the onscreen message you need to

1. Verify if the jar file is copied to CATALINA_HOME/webapps/q2r/WEB-INF/lib on the server hosting q2r web application
2. Restart tomcat

Restarting tomcat can be done at the end once all the JDBC drivers are registered instead of doing one by one after every jar file upload.

Creating a Data Source

Creating a data source involves creating a JDBC connection to the database as pointed by the URL provided. The data sources can be created by clicking on **“Data Sources”** -> **“Create Data Source”,** this will open up create data source dialog where you need provide

1. Data source alias
2. Selecting appropriate JDBC Driver
3. Providing JDBC Connection URL
4. Database Username
5. Database Password

On successfully saving the data source perform a “Test Connection” to verify if the provided credentials are accurate.

Administrators can define multiple data sources each pointing different database instance or even to different database vendor in heterogeneous database vendor environment.

Pool of JDBC connection is created for each data source created. The connections in the pool are not created aggressively instead connection are created only when all the existing connection are already being used and the limit to maximum number of connection to the data source is not reached. The default maximum number of connection limit is set to 5.

It is always recommended to use a read only user while creating the data source and the queries written while creating a report can be subjected to “SQL Injection” attacks.

Building a Report

Each report has one or more elements arranged in rows and columns. Each element has title, SQL Query, database connection pointing to database from where data is retrieved, type of chart renderer and refresh interval.

The application uses google charts. Google chart usage require you to be connected to the internet while rendering the report. The chart element supported by Q2R are

1. Bar Chart
2. Bar Stack Chart
3. Column Chart
4. Column Stack Chart
5. Line Chart
6. Annotated Line Chart
7. Pie Chart
8. Tabular Chart

In a report, some elements can be configured to fetch data from one database instance while the other elements can be configured to fetch data from different database instance. It’s possible to create a report that can pull data from multiple databases.

The element also has an option to auto refresh the data. This is turned off by default and the value -1 specifies no refresh. The report authors have an option to specify in seconds after which the element should get refreshed automatically. Different elements in the report have their own refresh rate. Element showing minutely data can be refreshed minutely while other elements in the same report showing hourly data can be configured to refresh hourly. The idea behind providing the refresh at element level is to prevent resource over utilization. User might want to see only one component's/element's data in real time and in such cases refresh entire report can be an over kill.

Authors can build reports and share them with others in the organization by saving them in "Public Reports" folder. Users can build reports and save them in "Personal Reports" folder for exclusive consumption.

Reports in “Public Reports” have view only access to all users whereas edit access to all Administrator users. Reports in “Personal Reports” folders are exclusively available only to the report authors.

Other Salient Features

- The application is built using HTML5 and bootstrap CSS for enhanced user experience.
- The application is based on AjgularJS framework making it easy to extend and faster performance.
- The application is based on responsive web design which makes the user interface compatitable with any device like laptops, tablets and mobile phones.
- Report building is web based, hence doesn't require any rich java client to be installed on the laptop/desktop.
- Report elements are loaded in parallel using AJAX and hence user need not wait for entire report to be rendered before see data/charts for simpler and smaller reporting elements.
- The application backend logic is implemented using RESTful web service which can be hosted on different server independently.

Appendix 1 – Sample Reports

Let's have following reporting use cases on the employees sample database to be addressed by Query2Report

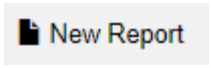
The demo is based on employees sample database available for MySQL database. More information about the content refer to https://dev.mysql.com/doc/employee/en/

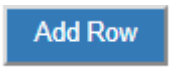
- Distribution of employees by departments, by designations and further distribution based on gender
- Employee hiring in the organization over the years

Distribution of employees by departments, by designations and gender

Build a report with two sections and four elements. The first section shows Pie chart of distribution of employees by department and a bar chart showing distribution by department by gender. The second section shows pie chart of employees by designation and a bar chart showing distribution by role by gender.

To build such a report in Query2Report we must create report template consisting of two sections and each section having two elements.

Click on 

and then click on 

button to create two rows in the report template then click 

on each row to create two columns per row

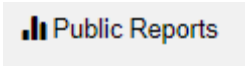
Click on edit image available in the top section to provide report title and report description.

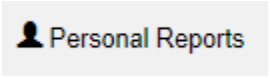
Click on edit image on the each of the elements, there will be a total of 4 elements created.

For each element, provide required title, SQL Query to fetch data, Chart type as per the requirement and refresh interval inputs. Refresh interval is default to -1 which means no automatic refresh, however you can click on the refresh icon to refresh each cell manually. But if you want auto refresh of any or all of the cells provide the refresh interval in seconds in this input.

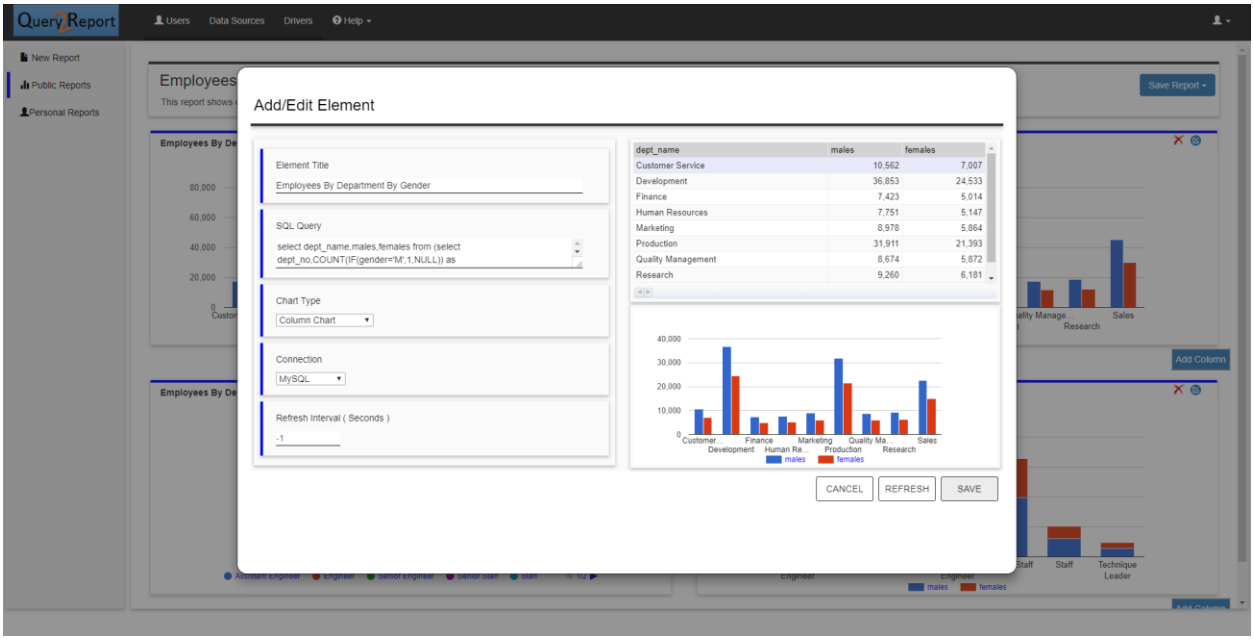
Once the element is defined, you can test the SQL Query using the 

button and then 

button to see the corresponding chart getting rendered with real data. Once all the element definition is finalized, you can choose to save reports in either of 

which means the report is available for all users or 

which means the report is available only to that user.



Report Queries

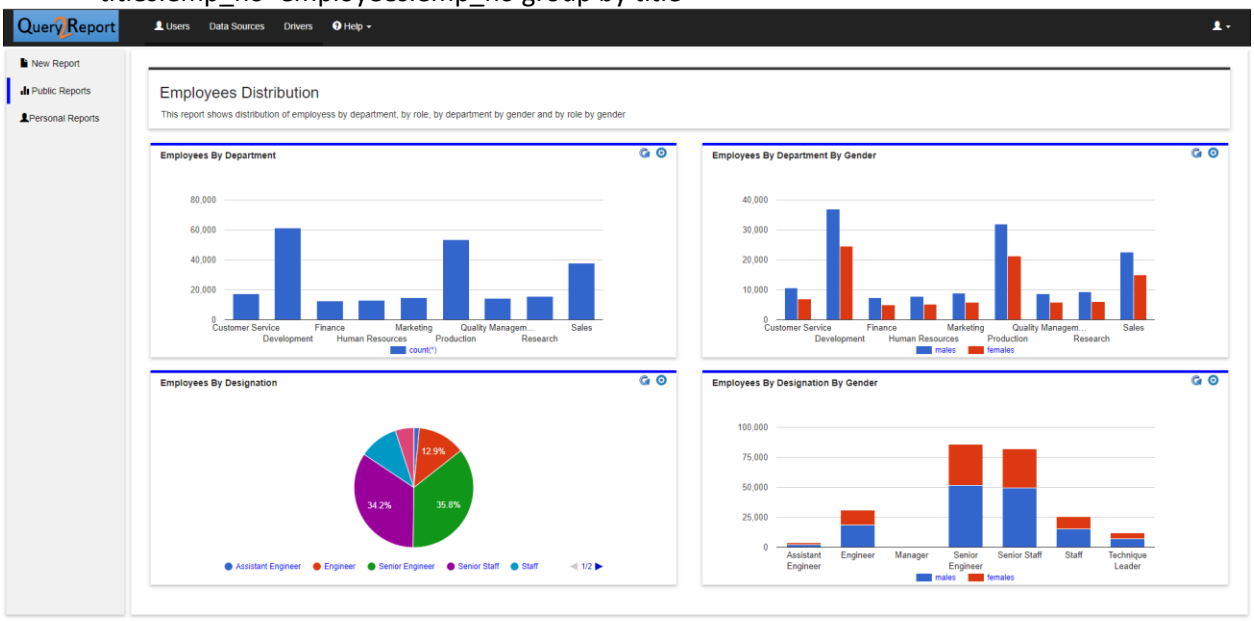
- **Employee By Department**

select dept_name,count(*) from employees.dept_emp,employees.departments where to_date='9999-01-01' and dept_emp.dept_no=departments.dept_no group by dept_name

- **Employees By Department By Gender**
  select dept_name,males,females from (select dept_no,COUNT(IF(gender='M',1,NULL)) as males,COUNT(IF(gender='F',1,NULL)) as females from employees.dept_emp,employees.employees where to_date='9999-01-01' and dept_emp.emp_no=employees.emp_no group by dept_no) t,employees.departments d where t.dept_no=d.dept_no

- **Employees By Designation**
  select title,count(*) from employees.titles,employees.employees where to_date='9999-01-01' and titles.emp_no=employees.emp_no group by title

- **Employees By Designation By Gender**
  select title,COUNT(IF(gender='M',1,NULL)) as males,COUNT(IF(gender='F',1,NULL)) as females from employees.titles,employees.employees where to_date='9999-01-01' and titles.emp_no=employees.emp_no group by title



Employee intake over the years

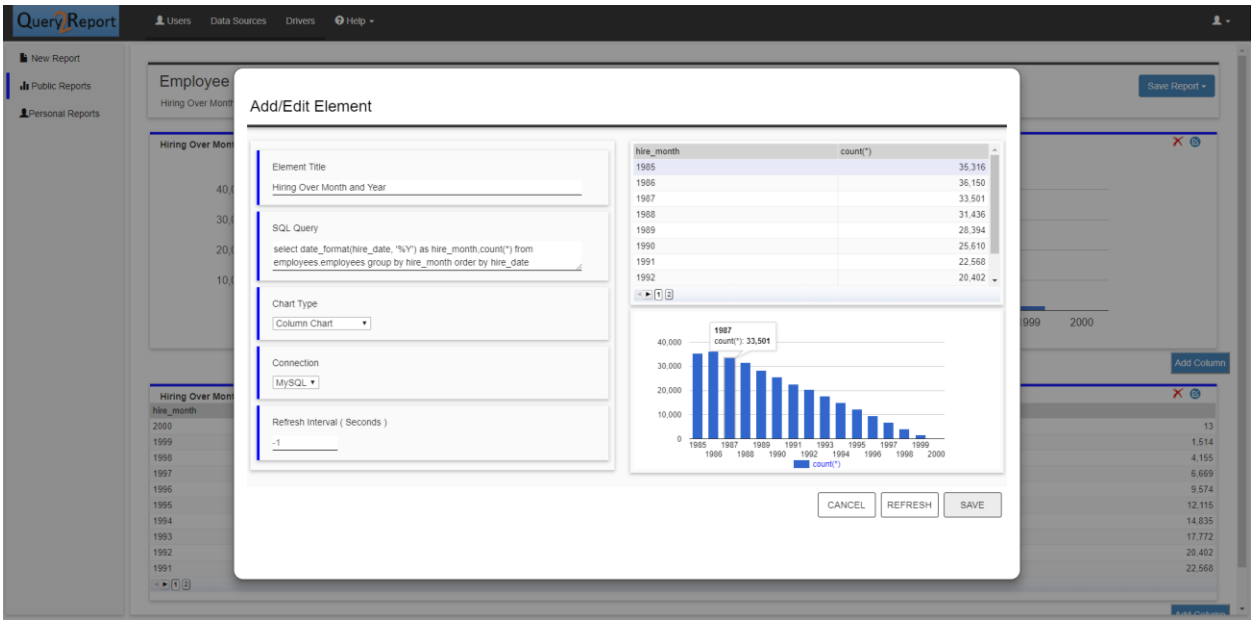
Build a report that shows trending of employee intake in the organization over the years. The report has two section, first section shows a bar chart of number of employees hired by year and second section shows a table displaying the same information

Click on "New Report" and build the report template consisting of two sections and each section having single elements. Click on "Add Row" button to create two rows of the report

Click on edit image available in the top section for report title and report description

Click on edit image on the each of the elements and provide required title, SQL Query to fetch data, Chart type as per the requirement and refresh interval inputs

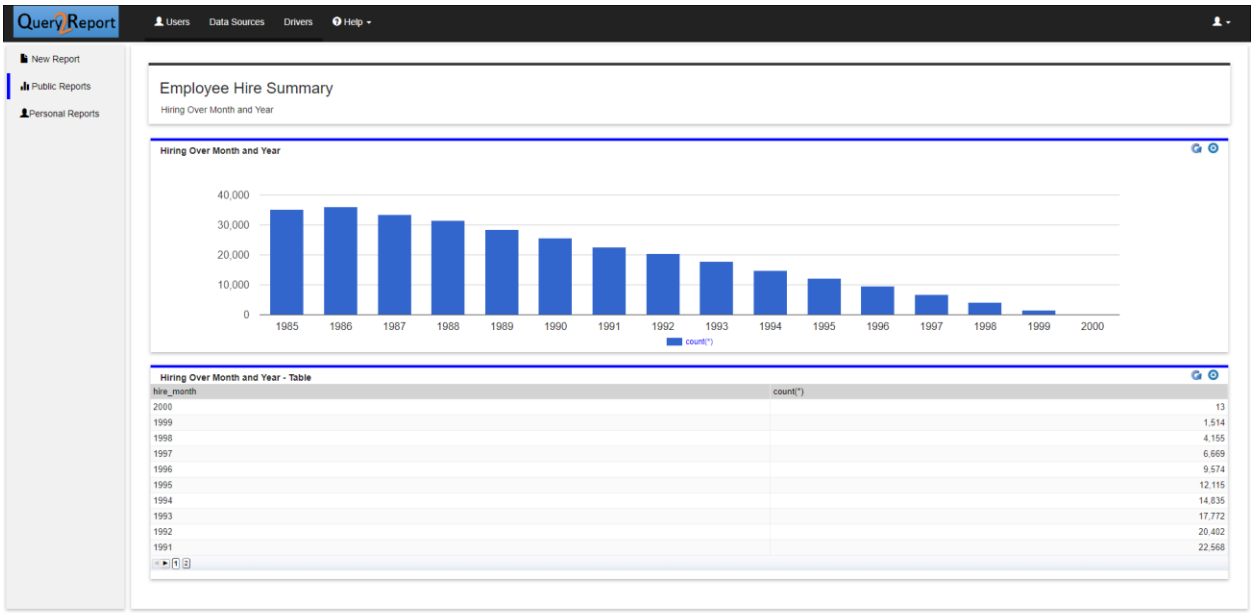
Once the element is defined, you can test the SQL Query using the "Test Element" button and then "Close Edit" button to see the corresponding chart getting rendered with real data. Once all the element definition is finalized, you can choose to save reports in either of "Personal Folder" which means the report is available for all users or "Private Folder" which means the report is available only to that user.



Report Queries

- **Hiring Over Month and Year**
  select date_format(hire_date, '%Y') as hire_month,count(*) from employees.employees group by hire_month order by hire_date

- **Hiring Over Month and Year - Table**
  select date_format(hire_date, '%Y') as hire_month,count(*) from employees.employees group by hire_month order by hire_date desc


QueryReport
Users
Data Sources
Drivers
Help
New Report
Public Reports
Personal Reports
Employee Hire Summary
Hiring Over Month and Year
Hiring Over Month and Year
40,000
30,000
20,000
10,000
0
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
count(*)
Hiring Over Month and Year - Table
hire_month
count(*)
2000 13
1999 1,514
1998 4,155
1997 6,669
1996 9,574
1995 12,115
1994 14,835
1993 17,772
1992 20,402
1991 22,568