

How to create your own RSS feed using MySQL and PHP

Introduction

[RSS](#) (Real Simple Syndication) feeds can be found everywhere on the internet. News and information are delivered using it and discussions forums, blogging sites enable usually different types of feeds for their users.

Whenever content defined by the RSS feed changes, feed offers to get an update. As an example, a blogging site could offer types "Full site feed", "Comments-only feed", "Label-specific site feed", "Individual post comment feed".

- Note about Atom format: In addition to the RSS, some sites offer Atom feeds too. There are some technical differences with the formats, but both are XML formats and in wide use. Earlier, Atom was designed to solve incompatibilities between prior RSS 2nd version, and is somewhat more complex. Both formats are IETF (Internet Engineering Task Force) specifications, RFC4287.
- Ovi has a new service called Ovi app wizard, using which anyone can create an RSS feed based app and publish it on Nokia Store:

http://oviappwizard.com/web_nokia/signin.jsp

Not only that the wizard will enable you to create an app basically of any valid RSS feed, it would now mean that you could create easily a RSS feed front client with app with it and get the information from your database distributed to a huge audience.

- Usage ideas: you could combine this with e.g. a WRT application that would read user input and data collected, send those to your web server and then push all or selected data out of it in form of RSS to feed the app you created with the Ovi app wizard for general public or your customers.

Prerequisites

- A web server; localhost (e.g. WAMP, LAMP) or remote hosting
- MySQL installed, service up and running on host
- PHP support enabled on host
- In this article the described MySQL operations are performed in WAMP environment.

You are free to use your favourite tools and environment. Also the SQL clauses are generic, not dependent on any specific environment, but in Wiki here are already a couple of examples related to [setting up WAMP and MySQL in How to install WAMP on Windows](#).

Implementation

For this article we will create two database tables with a couple of test records in MySQL. If you have not been working with MySQL database, you could have a look at related Wiki articles for reference. After that, we will look into how to output the data entered in those tables on a RSS feed format. Finally we will create an Ovi App out of the feed with the online wizard.

RSS feed structure

The <channel> element has three required child elements:

- <title> - Defines the title of the channel (e.g. your home page)
- <link> - Defines the hyperlink to the channel (e.g. <http://www.yourpage.com>)
- <description> - Describes the channel (e.g. My apps);

Each <channel> element can have one or more <item> elements. Each <item> element defines an article or "story" in the RSS feed.

The <item> element has three required child elements:

- <title> - Defines the title of the item (e.g. RSS Tutorial)
- <link> - Defines the hyperlink to the item (e.g. <http://www.w3schools.com/rss/>)
- <description> - Describes the item (e.g. New RSS tutorial on W3Schools)

Finally, the two last lines close the "channel" and "rss" elements.

Create a new database for RSS feeds

Note:

- you may use your previously created database, tables and data for this
- you may change the naming for database tables and fields at will
- if you do so, please remember also to modify other code accordingly.
- these steps require you to issue the password for user "root"
- you may prefer to use e.g. PHPMYADMIN front-end to perform this,
- this article uses Windows command prompt, MySQL command console and the mysqladmin command-line tool

The new database can be created with the following command:

```
<path_to_mysql_bin_directory>\bin>mysqladmin -u root -p create RSS_feeds
Enter password: <enter your password for root here>
```

After the database has been created, take it in use

```
mysql> use RSS_feeds
Database changed
```

Create table "details"

When creating a table it needs to have at least one field in it. The SQL sentence below creates the "details" table with initial "id"

http://developer.nokia.com/community/wiki/How_to_create_your_own_RSS_feed_using_MySQL_and_PHP

field, which is of integer type, cannot be null, and is automatically increased (counter). It will also serve as the primary key for this table.

```
mysql>create table details(id INT NOT NULL AUTO_INCREMENT, PRIMARY KEY(id));
```

Next we will create rest of the fields or columns for this table. To add into a table we use the SQL command ALTER TABLE:

```
mysql>alter table details add column title TEXT NOT NULL;
mysql>alter table details add column descr MEDIUMTEXT NOT NULL;
mysql>alter table details add column url TEXT;
```

Output after adding one column:

```
mysql> alter table details add column title text NOT NULL;
Query OK, 0 rows affected (0.11 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

You can view the columns created using the DESC-command in SQL.

```
mysql> desc details;
+-----+-----+-----+-----+-----+-----+
| Field | Type      | Null | Key | Default | Extra      |
+-----+-----+-----+-----+-----+-----+
| id    | int(11)   | NO   | PRI | NULL    | auto_increment |
| title | text      | NO   |     | NULL    |              |
| descr | mediumtext | NO   |     | NULL    |              |
| url   | text      | YES  |     | NULL    |              |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.02 sec)
```

The "id" field is not part of the produced RSS feed later, but to keep track and counter on the records added.

The first table is now ready, so let's create the second one.

- Note: text types in MySQL are: TINYTEXT, TEXT, MEDIUMTEXT, and LONGTEXT.

Create table "items"

The steps for creating this table follow what described previously, but with some different column names.

```
mysql> create table items(id INT NOT NULL AUTO_INCREMENT, PRIMARY KEY(id));
Query OK, 0 rows affected (0.05 sec)
```

Add the fields (or columns) "id", "title", "descr", "url" and "details" to this table the similar way you did create them for the "details" table. Use the SQL command ALTER TABLE, just replace the table name with "items" this time.

After you have created the columns, view the table structure using DESC command. It should look similar to this:

```
mysql> desc items;
+-----+-----+-----+-----+-----+-----+
| Field | Type      | Null | Key | Default | Extra      |
+-----+-----+-----+-----+-----+-----+
| id    | int(11)   | NO   | PRI | NULL    | auto_increment |
| title | text      | NO   |     | NULL    |              |
| descr | mediumtext | NO   |     | NULL    |              |
| url   | text      | NO   |     | NULL    |              |
| details | int(8)   | NO   |     | NULL    |              |
+-----+-----+-----+-----+-----+-----+
```

The "url" field is supposed to reference to the page with the original text, could be e.g. a discussion forum posted message or blog page.

The details is used to link the items sub-level items table with the main details table. The linking field is "details". You could experiment by entering rows to each table according to their data types. For the items table into the "details" field use a number that increases (e.g. 1-4). The id fields are automatically increased. Use either INSERT INTO clause in SQL or plain PHPMyAdmin program, if you want to edit the tables faster. You could also create an input form to pick the data from user, or have the database filled data from elsewhere, as you wish.

PHP scripts

The implementation consists of two PHP scripts:

- index_rss.php to be saved in web server root folder
- mysql_connect.php to be saved in non-web web visible folder

PHP script - mysql_connect

- Important note: you should save MySQL connection parameters file in a place on your web server
- The path should only be accessible internally, not web visible.

```

<?php

DEFINE ('DB_USER', 'root');
DEFINE ('DB_PASSWORD', '<your_root_password>');
DEFINE ('DB_HOST', '<your remote webserver URL or localhost>');
DEFINE ('DB_NAME', 'RSS_feeds');

$dbc = @mysql_connect (DB_HOST, DB_USER, DB_PASSWORD) OR die ('Could not connect to
MySQL: ' . mysql_error() );

mysql_select_db (DB_NAME) OR die ('Could not select the database: ' . mysql_error() );

?>

```

First the script will attempt connecting to the MySQL database according to the given credentials and then select the required database in use. In case any of these attempts fails, the script will exit with an error message.

This is a very generic and mostly used method to connect, if you want to implement this differently, you are free to do so.

PHP script – index_rss.php

This script first include the file "my_sq_connect" with the "require_once" method. Here PHP checks if the file has been already included, and if not will include / require it again.

The PHP method "header" is called before any actual output is sent, to ensure correct output format.

\$result_details variable is set to contain a query against the "details" table, from which all lines are fetched into an array. Then the variable \$row_details is used to get content in it from the database table using "mysql_fetch_array" method.

The variable \$xml is initialized to contain string for a valid XML RSS feed and the \$xml variable is built throughout the script with RSS data. It gets echoed at the end of the script.

The idea is first to go run the for next loop as long as it reaches the amount of rows fetched from the "details" table. The while clause inserts to \$row variable the content it gets using the query in \$result_items.

The final steps include closing the XML channel tag and the rss feed tag, then echoing the \$xml content on browser.

```

<?php

header("Content-Type: application/xml; charset=ISO-8859-1");

require_once ('mysql_connect.php');

$query_details = "SELECT * FROM details";

$result_details = mysql_query ($query_details) or die("Query failed with error:
".mysql_error());

$row_details = mysql_fetch_array($result_details);

$xml = '<?xml version="1.0" encoding="ISO-8859-1" ?><rss version="2.0"><channel>';

for ($counter = 1; $counter <= count($row_details); $counter += 1)
{
    $row = $result_details[$counter];
    $xml.='<title>'. $row['title'] .'</title><description>'.
    $row['descr'].'</description><link>'. $row['url'] .'</link>';
    $query_items = "SELECT * FROM items WHERE details = ".$counter;
    $result_items = mysql_query ($query_items) or die("Query failed with
error: ".mysql_error());

    while($row = mysql_fetch_array($result_items))
    {
        $xml .= '<item>
        <title>'. $row["title"] .'</title>
        <link>'. $row["url"] .'</link>
        <description>'. $row["descr"].'</description></item>';
    }
}

$xml .= '</channel>';
$xml .= '</rss></xml>';

echo $xml;

?>

```

You can use the character set you want instead of the one used. Feel free to experiment with this to get the desired output you want.

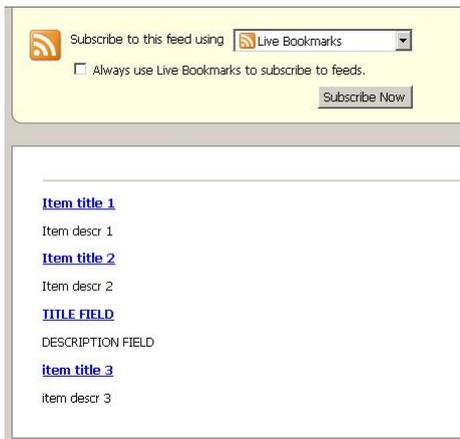
Example output

Note: Output here may differ fully on what you have, depending what you did insert database, and of the value of "details" field,

http://developer.nokia.com/community/wiki/How_to_create_your_own_RSS_feed_using_MySQL_and_PHP

(C) Copyright Nokia 2014. All rights reserved.

and if you use a different browser maybe.



Show result in a WRT widget

To show the results of the feed, you could parse thru it and create a display to the user. The snippet below shows the idea, and you can edit it depending on the tags used on the RSS feed created earlier.

- basic.js - code for WRT default JavaScript file

```
function XML_parsing_sequence(xml)
{
  $(xml).find("<replace_with_string_to_find>").each(function()
  {
    $("#output").append($(this).find("<replace_with_string_to_find>").text());

    //You may uncomment this, if you have added an attribute to the XML file, e.g. <(your
    id here)="1">
    //$("#output").append($(this).attr("id")+"<br />");

    //Add just one extra line in between each tag to form a list
    $("#output").append("<br>");

  });
}
```

- index.html - code for WRT default HTML-file

```
...
<script language="javascript" type="text/javascript" src="jquery.js"></script>
...

<body onLoad="javascript:init();">

<div id="output" align="center" ></div>

</body>
...
```

Tested with

- WAMP server on Windows XP (Includes MySQL)
- All services up and running
- Web Server online
- Firefox 3.6.3

Related Articles

- [Articles on How to install WAMP and MySQL server to create Mobile server use](#)

