**Generate PDF File from MySQL Database Using PHP**

Last Updated On: [March 5, 2018](https://www.phpflow.com/php/generate-pdf-file-mysql-database-using-php/)| By: [Parvez](https://www.phpflow.com/author/admin/)

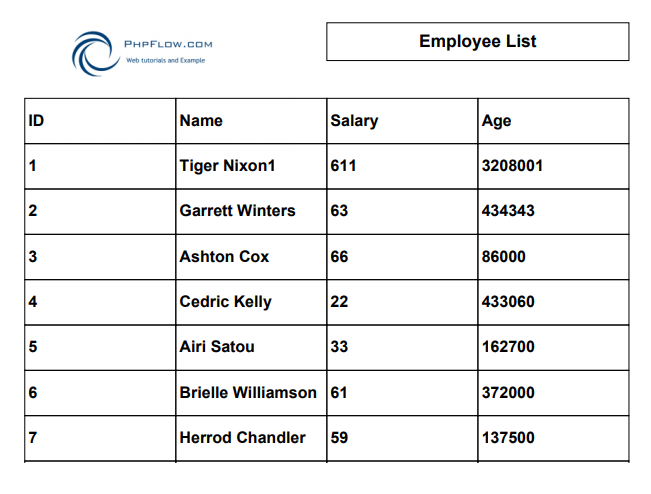
This is another export feature php tutorials, I will create simple PHP script to fetch data from MySQL and create pdf file using php. We will use third party [FPDF](http://www.fpdf.org/).The FPDF is very awesome PHP class to generate PDF using PHP from MySQL database.This is open source php library to generate pdf file using PHP.

PDF is very common and popular file format to read,view and write documents.PDF format is independent of application software, hardware, and operating systems.

**FPDF has following main features**

* Choice of measure unit, page format and margins.
* Page header and footer management.
* Automatic page and line break with text justification
* Image support (JPEG, PNG and GIF).
* Colors
* Links
* TrueType, Type1 and encoding support
* Page compression

There are dependency php extension which is **Zlib** to enable compression and GD for GIF image support. The latest version requires at least PHP 5.1.



Checkout other php export tutorials,

* [Exporting Data to Excel with PHP and MySQL](https://www.phpflow.com/php/exporting-data-to-excel-with-php-and-mysql/)
* [Export Data to CSV and Download Using PHP and MySQL](https://www.phpflow.com/php/export-data-to-csv-and-download-using-php-and-mysql/)
* [Import CSV File Into MySql Using PHP](https://www.phpflow.com/php/import-csv-file-into-mysql/)
* [Export HTML Table Data to Excel, CSV, PNG and PDF using jQuery Plugin](https://www.phpflow.com/php/export-html-table-data-to-excel-csv-png-and-pdf-using-jquery-plugin/)
* [Export the jQuery Datatable data to PDF,Excel,CSV and Copy](https://www.phpflow.com/php/how-to-export-the-jquery-datatable-data-to-pdfexcelcsv-and-copy/)

**We will follow following steps to Generate PDF**

* Download the FPDF library from [fpdf.org](http://www.fpdf.org/)
* We will featch data from MySQL database into the page.
* We will use FPDF libs function to generate pdf file with header and footer.

**Step 1:** We will create employee table into mysql database.



|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11 | --  -- Table structure for table `employee`  --    CREATE TABLE IF NOT EXISTS `employee` (    `id` int(11) NOT NULL AUTO\_INCREMENT COMMENT 'primary key',    `employee\_name` varchar(255) NOT NULL COMMENT 'employee name',    `employee\_salary` double NOT NULL COMMENT 'employee salary',    `employee\_age` int(11) NOT NULL COMMENT 'employee age',    PRIMARY KEY (`id`)  ) ENGINE=InnoDB  DEFAULT CHARSET=latin1 COMMENT='datatable demo table' AUTO\_INCREMENT=64 ; |

We will generate some sample data and insert into employee table.



|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21 | --  -- Dumping data for table `employee`  --    INSERT INTO `employee` (`id`, `employee\_name`, `employee\_salary`, `employee\_age`) VALUES  (1, 'Tiger Nixon', 320800, 61),  (2, 'Garrett Winters', 170750, 63),  (3, 'Ashton Cox', 86000, 66),  (4, 'Cedric Kelly', 433060, 22),  (5, 'Airi Satou', 162700, 33),  (6, 'Brielle Williamson', 372000, 61),  (7, 'Herrod Chandler', 137500, 59),  (8, 'Rhona Davidson', 327900, 55),  (9, 'Colleen Hurst', 205500, 39),  (10, 'Sonya Frost', 103600, 23),  (11, 'Jena Gaines', 90560, 30),  (12, 'Quinn Flynn', 342000, 22),  (13, 'Charde Marshall', 470600, 36),  (14, 'Haley Kennedy', 313500, 43),  (15, 'Tatyana Fitzpatrick', 385750, 19),  (16, 'Michael Silva', 198500, 66); |

**Step 2:** Connect MySQL database with PHP. We will create connection.php file and add below code.



|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20 | Class dbObj{  /\* Database connection start \*/  var $dbhost = "localhost";  var $username = "root";  var $password = "";  var $dbname = "test";  var $conn;  function getConnstring() {  $con = mysqli\_connect($this->dbhost, $this->username, $this->password, $this->dbname) or die("Connection failed: " . mysqli\_connect\_error());    /\* check connection \*/  if (mysqli\_connect\_errno()) {  printf("Connect failed: %s\n", mysqli\_connect\_error());  exit();  } else {  $this->conn = $con;  }  return $this->conn;  }  } |

Above file is used to connect and select database using PHP and MySQL.You need to change **$dbhost**, **$username**, **$password**, and **$dbname** variable’s value with your database credentials.

**Step 3:** We will create generate\_pdf.php file and add below code.



|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56 | <?php  //include connection file  include\_once("connection.php");  include\_once('libs/fpdf.php');    class PDF extends FPDF  {  // Page header  function Header()  {      // Logo      $this->Image('logo.png',10,-1,70);      $this->SetFont('Arial','B',13);      // Move to the right      $this->Cell(80);      // Title      $this->Cell(80,10,'Employee List',1,0,'C');      // Line break      $this->Ln(20);  }    // Page footer  function Footer()  {      // Position at 1.5 cm from bottom      $this->SetY(-15);      // Arial italic 8      $this->SetFont('Arial','I',8);      // Page number      $this->Cell(0,10,'Page '.$this->PageNo().'/{nb}',0,0,'C');  }  }    $db = new dbObj();  $connString =  $db->getConnstring();  $display\_heading = array('id'=>'ID', 'employee\_name'=> 'Name', 'employee\_age'=> 'Age','employee\_salary'=> 'Salary',);    $result = mysqli\_query($connString, "SELECT id, employee\_name, employee\_age, employee\_salary FROM employee") or die("database error:". mysqli\_error($connString));  $header = mysqli\_query($connString, "SHOW columns FROM employee");    $pdf = new PDF();  //header  $pdf->AddPage();  //foter page  $pdf->AliasNbPages();  $pdf->SetFont('Arial','B',12);  foreach($header as $heading) {  $pdf->Cell(40,12,$display\_heading[$heading['Field']],1);  }  foreach($result as $row) {  $pdf->Ln();  foreach($row as $column)  $pdf->Cell(40,12,$column,1);  }  $pdf->Output();  ?> |

We will include connection and pdf libs file, for customization of header and footer of pdf files, we will override header and footer methods and define our css design.

**Step 4:** We will create index.php file and added below code.



|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20 | <!DOCTYPE html>  <html lang="en">  <head>  <meta charset="UTF-8">  <title>Simple Example of PDF file using PHP and MySQL</title>  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.5/css/bootstrap.min.css">  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.5/css/bootstrap-theme.min.css">  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/font-awesome/4.7.0/css/font-awesome.min.css"/>  </head>  <body>  <div class="container" style="padding-top:50px">  <h2>Generate PDF file from MySQL Using PHP</h2>  <form class="form-inline" method="post" action="generate\_pdf.php">  <button type="submit" id="pdf" name="generate\_pdf" class="btn btn-primary"><i class="fa fa-pdf"" aria-hidden="true"></i>  Generate PDF</button>  </form>  </fieldset>  </div>  </body>  </html> |

We have added HTML form tag and define action value generate\_pdf.php file.

**Conclusion:**

We have generated pdf file using php and MySQL database, You can generate pdf using other database as well.You just need to replace MySQL query string.

You can download source code and see Demo from below link

# // EMAIL Simple Form to Email PHP PHP form to email explained

[Email Forms](http://form.guide/category/email-form), [Tutorials](http://form.guide/category/tutorial)

[email form in php](http://form.guide/tag/email-form-in-php), [email form with php](http://form.guide/tag/email-form-with-php), [php form to email](http://form.guide/tag/php-form-to-email)

It is a common requirement to have a form on almost any web site.

In this article, we will create a PHP script that will send an email when a web form is submitted.

There are two parts for the web form:

1. The HTML form code for the form. The HTML code below displays a standard form in the web browser. If you are new to HTML coding, please see: [HTML form tutorial](http://www.javascript-coder.com/html-form/html-form-tutorial-p1.phtml)
2. The PHP script for handling the form submission. The script receives the form submission and sends an email.

HTML code for the email form:

|  |  |  |
| --- | --- | --- |
| 1 | <form method="post" name="myemailform"action="form-to-email.php"> | |
| 2 |  |

|  |  |  |
| --- | --- | --- |
| 3 | Enter Name: <input type="text" name="name"> | |
| 4 |  |

|  |  |  |
| --- | --- | --- |
| 5 | Enter Email Address:    <input type="text" name="email"> | |
| 6 |  |

|  |  |  |
| --- | --- | --- |
| 7 | Enter Message:  <textarea name="message"></textarea> | |
| 8 |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 9 | <input type="submit" value="Send Form"> | | |
| 10 | | </form> |

The form contains the following fields:  
name, email and message.

name and email are single-line text input fields where as message is a text area field (multi-line text input).

You can have different types of input fields in a form. Please see the [HTML form input examples](http://form.guide/html-form/html-form-input.html) page for details.

On hitting the submit button, the form will be submitted to “form-to-email.php”. This form is submitted through the [POST method](http://form.guide/php-form/php-form-post.html)

Accessing the form submission data in the PHP script

Once your website visitor has submitted the form, the browser sends the form submission data to the script mentioned in the ‘action’ attribute of the form. (for the current form, the script is *form-to-email.php*)

Since we have the form submission method mentioned as POST in the form (method=’post’) we can access the form submission data through the $\_POST[] array in the PHP script.

The following code gets the values submitted for the fields: name, email and message.

|  |  |
| --- | --- |
| 1 | <?php |
| 2 | $name = $\_POST['name']; | |

|  |  |  |
| --- | --- | --- |
| 3 | $visitor\_email = $\_POST['email']; | |
| 4 | $message = $\_POST['message']; |

|  |  |
| --- | --- |
| 5 | ?> |

Composing the email message

Now, we can use the above PHP variables to compose an email message. Here is the code:

|  |  |
| --- | --- |
| 1 | <?php |
| 2 | $email\_from = 'yourname@yourwebsite.com'; | |

|  |  |
| --- | --- |
| 3 |  |
| 4 | $email\_subject = "New Form submission"; | |

|  |  |
| --- | --- |
| 5 |  |
| 6 | $email\_body = "You have received a new message from the user $name.\n". | |

|  |  |  |
| --- | --- | --- |
| 7 | "Here is the message:\n $message". | |
| 8 | ?> |

The ‘From’ address, the subject and the body of the email message are composed in the code above. Note the way the body of the message is composed using the variables.

If a visitor ‘Anthony’ submits the form, the email message will look like this:  
  
"You have received a new message from the user Anthony.  
Here is the message:  
Hi,  
Thanks for your great site. I love your site. Thanks and Bye.  
Anthony."

Sending the email

The PHP function to send email is mail().

|  |  |
| --- | --- |
| 1 | mail(to,subject,message,headers) |

For more details, see the [PHP mail()](http://www.php.net/manual/en/function.mail.php) page.

The *headers* parameter is to provide additional mail parameters ( like the from address, CC, BCC etc)

Here is the code to send the email:

|  |  |  |
| --- | --- | --- |
| 1 | <?php | |
| 2 |  |

|  |  |  |
| --- | --- | --- |
| 3 | $to = "yourname@yourwebsite.com"; | |
| 4 |  |

|  |  |  |
| --- | --- | --- |
| 5 | $headers = "From: $email\_from \r\n"; | |
| 6 |  |

|  |  |  |
| --- | --- | --- |
| 7 | $headers .= "Reply-To: $visitor\_email \r\n"; | |
| 8 |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 9 | mail($to,$email\_subject,$email\_body,$headers); | | |
| 10 | |  |

|  |  |
| --- | --- |
| 11 | ?> |

Notice that we put your email address in the ‘From’ parameter and the visitor’s email address in the ‘Reply-To’ parameter. The ‘From’ parameter should indicate the origin of the email. If you put the visitor’s email address in the ‘From’ parameter, some email servers might reject the email thinking that you are impersonating someone.

Sending the email to more than one recipients

If you want to send the email to more than one recipients, then you just need to add these in the “$to” variable.

|  |  |
| --- | --- |
| 1 | <?php |
| 2 | $to = "name1@website-name.com, name2@website-name.com,name3@website-name.com"; | |

|  |  |
| --- | --- |
| 3 |  |
| 4 | mail($to,$email\_subject,$email\_body,$headers); | |

|  |  |
| --- | --- |
| 5 | ?> |

You can use the CC (carbon copy) and BCC (Blind Carbon Copy) parameters as well. The CC and BCC emails are added in the *‘headers’* parameter.

**Sample code:**

|  |  |
| --- | --- |
| 1 | <?php |
| 2 | $to = "name1@website-name.com, name2@website-name.com,name3@website-name.com"; | |

|  |  |
| --- | --- |
| 3 |  |
| 4 | $headers = "From: $email\_from \r\n"; | |

|  |  |
| --- | --- |
| 5 |  |
| 6 | $headers .= "Reply-To: $visitor\_email \r\n"; | |

|  |  |
| --- | --- |
| 7 |  |
| 8 | $headers .= "Cc: someone@domain.com \r\n"; | |

|  |  |  |
| --- | --- | --- |
| 9 |  | |
| 10 | | $headers .= "Bcc: someoneelse@domain.com \r\n"; | |

|  |  |
| --- | --- |
| 11 |  |
| 12 | mail($to,$email\_subject,$email\_body,$headers); | |

|  |  |
| --- | --- |
| 13 | ?> |

Securing the form against email injection

Spammers are looking for exploitable email forms to send spam emails. They use the form handler script as a ‘relay’. What they do is to submit the form with manipulated form values. To secure our form from such attacks, we need to validate the submitted form data.

All the values that go in the ‘*headers*‘ parameter should be checked to see whether it contains \r or \n. The hackers insert these characters and add their own code to fool the function.

Here is the updated code:

|  |  |
| --- | --- |
| 1 | <?php |
| 2 | function IsInjected($str) | |

|  |  |
| --- | --- |
| 3 | { |
| 4 | $injections = array('(\n+)', | |

|  |  |
| --- | --- |
| 5 | '(\r+)', |
| 6 | '(\t+)', |

|  |  |
| --- | --- |
| 7 | '(%0A+)', |
| 8 | '(%0D+)', |

|  |  |  |
| --- | --- | --- |
| 9 | '(%08+)', | |
| 10 | | '(%09+)' |

|  |  |
| --- | --- |
| 11 | ); |
| 12 |  | |

|  |  |  |
| --- | --- | --- |
| 13 | $inject = join('|', $injections); | |
| 14 | $inject = "/$inject/i"; |

|  |  |
| --- | --- |
| 15 |  |
| 16 | if(preg\_match($inject,$str)) | |

|  |  |
| --- | --- |
| 17 | { |
| 18 | return true; | |

|  |  |
| --- | --- |
| 19 | } |
| 20 | else | |

|  |  |
| --- | --- |
| 21 | { |
| 22 | return false; | |

|  |  |  |
| --- | --- | --- |
| 23 | } | |
| 24 | } |

|  |  |
| --- | --- |
| 25 |  |
| 26 | if(IsInjected($visitor\_email)) | |

|  |  |
| --- | --- |
| 27 | { |
| 28 | echo "Bad email value!"; | |

|  |  |  |
| --- | --- | --- |
| 29 | exit; | |
| 30 | } |

|  |  |
| --- | --- |
| 31 | ?> |

In general, any value used in the header should be validated using the code above.

Better, complete validations could be done using the [PHP form validation script here](http://form.guide/php-form/php-form-validation.html).

PHP form to email complete code

The link below contains the complete form, validation and emailing code.