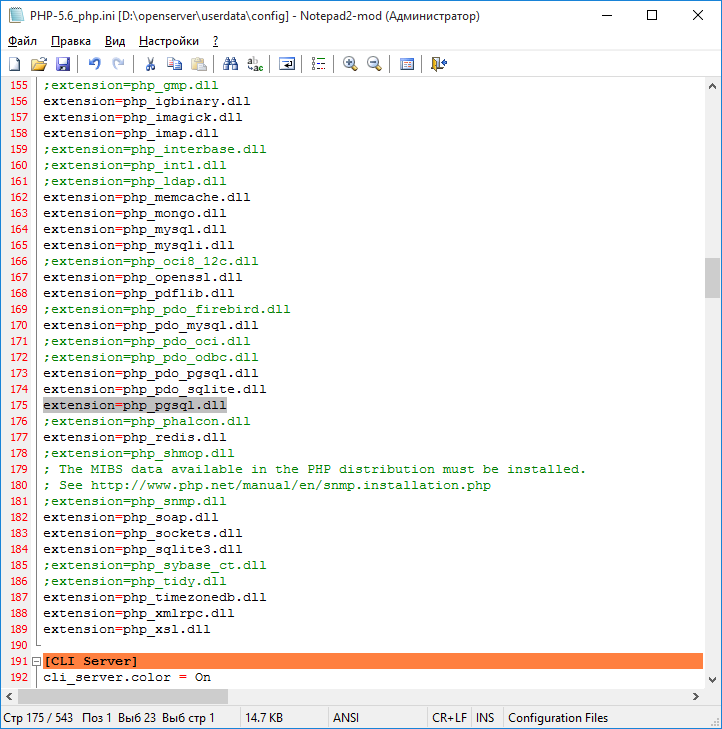
Для работы с PostgreSQL на PHP в файле настроек php.ini нужно раскомментировать (убрать точку с запятой вначале) строку

extension=php\_pgsql.dll



**Соединение с базой**

Для соединения использовать функцию **pg\_connect(string $connection\_string [, int $connect\_type ])**

Список основных ключевых слов **connection\_string**: host, port, dbname, user, password, connect\_timeout.

Пример **connection\_string:**

$connection\_string = "host=localhost port=5432 dbname=test user=postgres password=admin";

**connect\_type** - если в функцию передана константа **PGSQL\_CONNECT\_FORCE\_NEW**, будет создаваться новое подключение, даже если connection\_string идентична строке существующего подключения.

**pg\_close(**[ resource $connection ]**)** закрывает соединение с базой данных PostgreSQL

<?php

$connection\_string = "host=localhost port=5432 dbname=test user=postgres password=admin";

$dbconn = pg\_connect($connection\_string);

if ($dbconn)

echo "Connection successful";

else

die("Connection error");

pg\_close($dbconn);

?>

**Выполнение запросов**

**pg\_query()** – простой запрос

**pg\_query** ([ resource $connection ], string $query )

**connection** – соединение с БД. Если не передать параметр, то используется последнее созданное соединение.

**query** – SQL запрос. Может содержать несколько запросов разделённых точкой с запятой

**pg\_query\_params()** – параметризированный запрос

**pg\_query\_params** ([ resource $connection ], string $query , array $params )

**connection** – соединение с БД. Если не передать параметр, то используется последнее созданное соединение.

**query** - SQL запрос. Может содержать только один запрос. Если в запрос будут передаваться параметры, то они заменят псевдопеременные $1, $2 и т.д.

**params** - Массив значений параметров запроса для замены псевдопеременных $1, $2 и т.д. в исходной строке запроса. Количество элементов массива должно точно совпадать с количеством псевдопеременных.

$result = pg\_query\_params($dbconn, 'SELECT \* FROM "TEST1" WHERE "NAME" LIKE $1 and "ID" BETWEEN $2 and $3', array('%'.'asd'.'%', 1, 5));

Этот запрос выбирает все записи, в имени которых содержится «asd» и значение их идентификатора между 1 и 5.

**Получение результатов**

**pg\_num\_rows()**

**Функция pg\_num\_rows(**resource $result**) возвращает количество строк в результате запроса:**

$rows = pg\_num\_rows($result);

echo 'Returned '.$rows.' rows.<br>';

**pg\_fetch\_row()**

**Функция pg\_fetch\_row(**resource $result [, int $row ]**) возвращает одну строку из результата запроса**

**result** - ресурс результата запроса PostgreSQL, возвращаемый функциями pg\_query(), pg\_query\_params()

**row** - Номер выбираемой из результата запроса строки. Нумерация начинается с нуля. Если аргумент опущен или равен **NULL**, берется следующая по очереди строка.

Результат - численно индексированный массив значений из строки результата запроса. Индексация начинается с нуля. Значения представляются в виде строк. Значения *NULL* базы данных преобразуются в PHP **NULL**. **FALSE**, когда row превышает число строк в результате запроса, когда строк в результате не осталось, и при прочих ошибках.

while ($row = pg\_fetch\_row($result))

{

echo "ID: $row[0] NAME: $row[1]";

echo "<br/>";

}

**Другие функции**

**pg\_num\_fields(**resource $result**) – возвращает количество колонок из результата запроса**

**pg\_field\_name(**resource $result , int $field\_number**) – возвращает название колонки.**

**pg\_field\_num(**resource $result , string $field\_name**) – возвращает номер колонки по названию**

<html>

<head>

<title>

PostgreSQL

</title>

<style>

table, tr, td

{

border: 1px solid black;

}

</style>

</head>

<body>

<?php

$connection\_string = "host=localhost port=5432 dbname=test user=postgres password=admin";

$dbconn = pg\_connect($connection\_string);

if (!$dbconn)

die("Connection error");

$result = pg\_query\_params($dbconn, 'SELECT \* FROM "TEST1" WHERE "NAME" LIKE $1 and "ID" BETWEEN $2 and $3', array('%'.'asd'.'%', 1, 5));

$rows = pg\_num\_rows($result);

$cols = pg\_num\_fields($result);

echo 'Returned '.$rows.' rows.<br>';

echo '<table>';

echo '<tr>';

for ($j = 0; $j < $cols; $j++)

echo '<td>'.pg\_field\_name($result, $j).'</td>';

echo '</tr>';

while ($row = pg\_fetch\_row($result))

{

echo '<tr>';

for ($j = 0; $j < $cols; $j++)

echo '<td>'.$row[$j].'</td>';

echo '</tr>';

}

echo '</table>';

pg\_close($dbconn);

?>

</body>

</html>

Результат:

