

# Przydatne strony

<http://www.php5-tutorial.com/mysql/connecting-to-mysql/>

[http://www.w3schools.com/php/php\\_mysql\\_intro.asp](http://www.w3schools.com/php/php_mysql_intro.asp)

# What is MySQL?

- MySQL is a database system used on the web
- MySQL is a database system that runs on a server
- MySQL is ideal for both small and large applications
- MySQL is very fast, reliable, and easy to use
- MySQL uses standard SQL
- MySQL compiles on a number of platforms
- MySQL is free to download and use
- MySQL is developed, distributed, and supported by Oracle Corporation

# Bazy danych

## PHP Connect to MySQL

PHP 5 and later can work with a MySQL database using:

- **MySQLi extension** (the "i" stands for improved)
- **PDO (PHP Data Objects)**

Earlier versions of PHP used the MySQL extension. However, this extension was deprecated in 2012.

- MySQLi (object-oriented)
- MySQLi (procedural)
- PDO

**MySQLi** – rozszerzenie języka **PHP** służące do obsługi bazy danych **MySQL** za pomocą składni obiektowej. Pojawiło się ono w PHP 5 i wymaga MySQL w wersji 4.1.3 lub wyższej.

## Biblioteka PDO

PDO to skrót od *PHP Data Objects*. Jest to zupełnie nowy interfejs języka PHP przeznaczony do komunikacji z bazami danych, po raz pierwszy napisany wyłącznie w OOP. Jego najważniejszą zaletą jest to, że możemy za jego pomocą łączyć się zarówno z bazą danych MySQL, jak i z bazą danych PostgreSQL (o innych systemach DB nie wspominając). Wersji beta PDO można było używać już w PHP 5.0, natomiast stabilna wersja pojawiła się wraz z PHP 5.1.


**PHP OOP (ang. Object-Oriented Programming)** to sposób projektowania aplikacji, w którym programy definiuje się za pomocą obiektów. Podejście obiektowe, w odróżnieniu od proceduralnego daje programistom więcej możliwości przy projektowaniu dużych i złożonych systemów. Obiekty realizują niezależnie pewne zadania a sposób ich wykonania nie jest istotny dla innych elementów systemu.

# Sprawdzić ustawienia

C:\WebServ 64\php\php.ini

```
1163 [MySQLi]
1164
1165 ; Maximum number of persistent links. -1 means no limit.
1166 ; http://php.net/mysql.max-persistent
1167 mysqli.max_persistent = -1
1168
1169 ; Allow accessing, from PHP's perspective, local files with
1170 ; http://php.net/mysql.allow_local_infile
1171 ;mysqli.allow_local_infile = On
1172
1173 ; Allow or prevent persistent links
1174 ; http://php.net/mysql.allow-persistent
1175 mysqli.allow_persistent = On
1176
1177 ; Maximum number of links. -1 means no limit.
1178 ; http://php.net/mysql.max-links
1179 mysqli.max_links = -1
1180
1181 ; If mysqlnd is used: Number of cache buffers per host
1182 ; http://php.net/mysql.cache_size
1183 mysqli.cache_size = 2000
1184
1185 ; Default port number for mysqli_connect(). If unset, mysqli_connect() will use
1186 ; the $MYSQL_TCP_PORT or the mysql-tcp entry in /etc/services or the
1187 ; compile-time value defined MYSQL_PORT (in that order). Win32 will only look
1188 ; at MYSQL_PORT.
1189 ; http://php.net/mysql.default-port
1190 mysqli.default_port = 3306
1191
1192 ; Default socket name for local MySQL connects. If empty, uses the built-in
1193 ; MySQL defaults.
1194 ; http://php.net/mysql.default-socket
1195 mysqli.default_socket =
1196
1197 ; Default host for mysql_connect() (doesn't apply in safe mode).
1198 ; http://php.net/mysql.default-host
```

880	extension=php_mbstring.dll
881	extension=php_exif.dll
882	extension=php_mysql.dll
883	extension=php_mysqli.dll
884	;extension=php_oci8.dll
885	;extension=php_oci8_11g.dll
886	extension=php_openssl.dll
887	;extension=php_pdo_firebird.dll



# PDO

PDO (PHP Data Objects) to rozszerzenie PHP o architekturze obiektowej, udostępniające jednolity, uniwersalny interfejs do komunikacji z bazami danych. Jest oparte na sterownikach, z których każdy udostępnia takie same metody, aczkolwiek może również dodawać własne, specyficzne dla konkretnej bazy danych (na przykład sterownik dla PostgreSQL udostępnia trzy dodatkowe metody do obsługi dużych obiektów - large object). PDO jest dostępne w PHP 5.1 i nowszych wersjach i nie wymaga dodatkowej konfiguracji (poza dodaniem sterowników dla wykorzystywanych baz danych).

Największą zaletą PDO jest jednolity interfejs. PHP posiada osobne rozszerzenia dla każdej bazy danych. Jeśli chcemy skorzystać z bazy MySQL, wywołujemy funkcje z prefiksem `mysql_` lub `mysqli_`, jeśli korzystamy z PostgreSQL - funkcje z prefiksem `pg_`. Dzięki PDO nasze skrypty będą korzystać z tych samych metod, niezależnie od wykorzystywanej bazy. Trzeba jednak pamiętać, że PDO nie uniezależnia nas w stu procentach od wykorzystywanej bazy danych - jeśli w zapytaniu skorzystamy z funkcji specyficznej dla danego silnika, to zapytanie nie zadziała na innej bazie. Drugą zaletą PDO jest walidacja danych wysyłanych w zapytaniach oraz zwolnienie programisty z konieczności filtrowania tych danych pod kątem zabezpieczeń przed atakami typu SQL Injection.

# Połączenie z serverem MySQL

The `mysqli_connect()` function opens a new connection to the MySQL server.

Returns an object representing the connection to the MySQL server

//proceduralnie

```
$servername = "localhost";
```

```
$username = "root";
```

```
$password = "";
```

```
// tworzymy połączenie z MySQL
```

```
$conn = mysqli_connect($servername, $username, $password);
```

```
// sprawdzamy połączenie
```

```
if (!$conn) {
```

```
    die("Błąd połączenia : " . mysqli_connect_error());
```

```
}
```

```
echo "Połączono z serwerem MySQL ";
```

```
mysqli_connect(host, username, password, dbname, port, socket);
```

opcjonalnie

//obiekto

```
$servername = "localhost";
```

```
$username = "root";
```

```
$password = "";
```

```
// tworzymy połączenie z MySQL
```

```
$conn = new mysqli($servername, $username, $password);
```

```
// sprawdzamy połączenie
```

```
if ($conn->connect_error) {
```

```
    die("Błąd połączenia: " . $conn->connect_error);
```

```
}
```

```
echo "Połączono z serwerem MySQL ";
```

# Tworzymy bazę danych

```
mysqli_query(connection, query, resultmode);
```

```
// Tworzymy bazę danych (proceduralnie)
$sql = "CREATE DATABASE UczeńP";
//$sql = "CREATE DATABASE IF NOT EXISTS UczeńP";
if (mysqli_query($conn, $sql)) {
    echo "Baza danych utworzona pomyślnie";
} else {
    echo "Błąd podczas tworzenia bazy: " . mysqli_error($conn);
}
```

```
// Tworzymy bazę danych (obiektoowo)
$sql = "CREATE DATABASE UczeńO";
//$sql = "CREATE DATABASE IF NOT EXISTS UczeńO";
if ($conn->query($sql) === TRUE) {
    echo "Baza danych utworzona pomyślnie";
} else {
    echo "Błąd podczas tworzenia bazy: " . $conn->error;
}
```

# mysqli\_query(connection,query,resultmode);

The mysqli\_query() function performs a query against the database.

```
mysqli_query(connection,query,resultmode);
```

Parameter	Description
<i>connection</i>	Required. Specifies the MySQL connection to use
<i>query</i>	Required. Specifies the query string

*resultmode* Optional. A constant. Either:

- MYSQLI\_USE\_RESULT (Use this if we have to retrieve large amount of data)
- MYSQLI\_STORE\_RESULT (This is default)

**Return Value:** For successful SELECT, SHOW, DESCRIBE, or EXPLAIN queries it will return a mysqli\_result object. For other successful queries it will return TRUE. FALSE on failure



# Close the Connection

The connection will be closed automatically when the script ends. To close the connection before, use the following:

## Example (MySQLi Object-Oriented)

```
$conn->close();
```

## Example (MySQLi Procedural)

```
mysqli_close($conn);
```

## Example (PDO)

```
$conn = null;
```

## MySQL Date and Time Types

	DATETIME	DATE	TIMESTAMP	TIME	YEAR
"Zero" value	'0000-00-00 00:00:00'	'0000-00-00'	'0000-00-00 00:00:00'	'00:00:00'	0000
Storage requirement (Bytes)		8	3	4	3 1
Date information	Yes	Yes	Yes	No	Yes
Time information	Yes	No	Yes	Yes	No
Display format	'YYYY-MM-DD HH:MM:SS'	'YYYY-MM-DD'	'YYYY-MM-DD HH:MM:SS'	'HH:MM:SS'	YYYY
Display width (Characters)	19		19		

### ***Range of legal values***

Min	'1000-01-01 00:00:00'	'1000-01-01'	1970	'-838:59:59'	1901
Max	'9999-12-31 23:59:59'	'9999-12-31'	2037	'838:59:59'	2155

### ***Automatic ? to the current date and time***

Initialization	No	No	Yes	No	No
Updating	No	No	Yes	No	No

# Tworzymy tabelę

```
8 <?php
9 //proceduralnie
10 $servername = "localhost";
11 $username = "root";
12 $password = "";
13 $dbname = "uczenp";
14 // tworzymy połączenie z baza danych
15 $conn = mysqli_connect($servername, $username, $password, $dbname);
16 // sprawdzamy połączenie
17 if (!$conn) {
18     die("Błąd połączenia : " . mysqli_connect_error());
19 }
20 echo "Połączono z baza danych ";
21 // tworzymy table
22 $sql = "CREATE TABLE danep (
23     id INT(3) UNSIGNED AUTO_INCREMENT ,
24     imie VARCHAR(30) NOT NULL,
25     nazwisko VARCHAR(30) NOT NULL,
26     email VARCHAR(50),
27     data_utworzenia TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
28     data_modyfikacji  TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
29     PRIMARY KEY (id)
30 );";
31 //równoważne działanie: data_utworzenia TIMESTAMP DEFAULT now()
32 if (mysqli_query($conn, $sql)) {
33     echo "Tabela danep utworzona";
34 } else {
35     echo "Błąd podczas tworzenia tabeli: " . mysqli_error($conn);
36 }
37 mysqli_close($conn);
38 ?>
```

# Tworzymy tabelę

```
8 <?php
9 //obiekto
10 $servername = "localhost";
11 $username = "root";
12 $password = "";
13 $dbname = "uczeno";
14 // tworzymy połączenie z baza danych
15 $conn = new mysqli($servername, $username, $password,$dbname );
16 // sprawdzamy połączenie
17 if ($conn->connect_error) {
18     die("Błąd połączenia: " . $conn->connect_error);
19 }
20 echo "Połączono z baza danych ";
21 // tworzymy tabele
22 $sql = "CREATE TABLE daneo (
23 id INT(3) UNSIGNED AUTO_INCREMENT ,
24 imie VARCHAR(30) NOT NULL,
25 nazwisko VARCHAR(30) NOT NULL,
26 email VARCHAR(50),
27 data_utworzenia TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
28 data_modyfikacji TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
29 PRIMARY KEY (id)
30 )";
31 if ($conn->query($sql) === TRUE) {
32     echo "Tabela uczeno utworzona";
33 } else {
34     echo "Błąd podczas tworzenia tabeli: " . $conn->error;
35 }
36 $conn->close();
37 ?>
```

# Dodajemy rekord

```
8 <?php
9 //proceduralnie
10 $servername = "localhost";
11 $username = "root";
12 $password = "";
13 $dbname = "uczenp";
14 // tworzymy połączenie z baza danych
15 $conn = mysqli_connect($servername, $username, $password, $dbname);
16 // sprawdzamy połączenie
17 if (!$conn) {
18     die("Błąd połączenia : " . mysqli_connect_error());
19 }
20 echo "Połączono z z baza danych ";
21 //dodajemy rekord do tabeli:
22 $sql = "INSERT INTO danep(imie, nazwisko, email)
23 VALUES ('Agnieszka', 'Nowakowska', 'aga@example.com')";
24
25 if (mysqli_query($conn, $sql)) {
26     echo "dodano rekord do tabeli";
27 } else {
28     echo "Błąd podczas dodawania rekordu do tabeli:" . mysqli_error($conn);
29 }
30 mysqli_close($conn);
31 ?>
```

# Dodajemy rekord

```
8 <?php
9 // obiektowo
10 $servername = "localhost";
11 $username = "root";
12 $password = "";
13 $dbname = "uczeno";
14 // tworzymy połączenie z baza danych
15 $conn = new mysqli($servername, $username, $password, $dbname );
16 // sprawdzamy połączenie
17 if ($conn->connect_error) {
18     die("Błąd połączenia: " . $conn->connect_error);
19 }
20 echo "Połączono z z baza danych ";
21 ///dodajemy rekord do tabeli:
22 $sql = "INSERT INTO daneo(imie, nazwisko, email)
23 VALUES ('Agnieszka', 'Nowakowska', 'aga@example.com')";
24
25 if ($conn->query($sql) === TRUE) {
26     echo "dodano rekord do tabeli";
27 } else {
28     echo "Błąd podczas dodawania rekordu do tabeli:: " . $conn->error;
29 }
30 $conn->close();
31 ?>
```

# Dodajemy wiele rekordów

```
8 <?php
9 //proceduralnie
10 $servername = "localhost";
11 $username = "root";
12 $password = "";
13 $dbname = "uczenp";
14 // tworzymy połączenie z baza danych
15 $conn = mysqli_connect($servername, $username, $password, $dbname);
16 // sprawdzamy połączenie
17 if (!$conn) {
18     die("Błąd połączenia : " . mysqli_connect_error());
19 }
20 echo "Połączono z baza danych ";
21 //dodajemy kilka rekordów do tabeli:
22 $sql = "INSERT INTO danep(imie, nazwisko, email)
23 VALUES ('Agnieszka', 'Nowakowska', 'aga@example.com'),
24 ('Joanna', 'Wójcicka', 'jwojcicka@example.com'),
25 ('Dorota', 'Maciejewska', 'dm@example.com)";
26
27 if (mysqli_multi_query($conn, $sql)) {
28     echo "dodano rekordy do tabeli";
29 } else {
30     echo "Błąd podczas dodawania rekordów do tabeli:" . mysqli_error($conn);
31 }
32 mysqli_close($conn);
33 ?>
```

# Dodajemy wiele rekordów

```
8 <?php
9 //obiektoowo
10 $servername = "localhost";
11 $username = "root";
12 $password = "";
13 $dbname = "uczeno";
14 // tworzymy połączenie z baza danych
15 $conn = new mysqli($servername, $username, $password,$dbname );
16 // sprawdzamy połączenie
17 if ($conn->connect_error) {
18     die("Błąd połączenia: " . $conn->connect_error);}
19 echo "Połączono z baza danych ";
20 ///dodajemy kilka rekordów do tabeli:
21 $sql = "INSERT INTO daneo(imie, nazwisko, email)
22 VALUES ('Agnieszka', 'Nowakowska', 'aga@example.com'),
23 ('Joanna', 'Wójcicka', 'jwojcicka@example.com'),
24 ('Dorota', 'Maciejewska', 'dm@example.com)";
25
26 if ($conn->multi_query($sql) === TRUE) {
27     echo "rekordy dodano do tabeli";
28 } else {
29     echo "Błąd podczas dodawania rekordów do tabeli:: " . $conn->error;
30 }
31 $conn->close();
32 ?>
```



# Pobieramy dane z bazy

## proceduralnie

```
mysqli_num_rows(result);
```

The `mysqli_num_rows()` function returns the number of rows in a result set.

*result* Required. Specifies a result set identifier returned by `mysqli_query()`, `mysqli_store_result()` or `mysqli_use_result()`

```
array mysqli_fetch_assoc ( mysqli_result result )
```

The `mysqli_fetch_assoc()` function is used to return an associative array representing the next row in the result set for the result represented by the `result` parameter, where each key in the array represents the name of one of the result set's columns.

## obiektoowo

# Pobieramy dane z bazy

```
8 <?php
9 //proceduralnie
10 $servername = "localhost";
11 $username = "root";
12 $password = "";
13 $dbname = "uczenp";
14 // tworzymy połączenie z baza danych
15 $conn = mysqli_connect($servername, $username, $password, $dbname);
16 // sprawdzamy połączenie
17 if (!$conn) {
18     die("Błąd połączenia : " . mysqli_connect_error());
19 }
20 echo 'Połączono z baza danych<br> ';
21 //odczytujemy dane z tabeli:
22 $sql = "SELECT * from danep ";
23
24 $wynik=mysqli_query($conn, $sql);
25 if (mysqli_num_rows($wynik)>0) {
26     echo 'dane zostały odczytane <br>';
27     echo '<table border="2">';
28     echo '<tr><th>id</th><th>imie</th><th>nazwisko</th><th>email</th><th>data_utworzenia</th><th>data_modyfikacji</th></tr>';
29     // output data of each row
30     while($row = mysqli_fetch_assoc($wynik)) {
31         echo '<tr><td>' . $row['id']. ' </td><td>' . $row['imie']. ' </td><td>' . $row['nazwisko']. ' </td><td>' . $row['email']
32         . ' </td><td>' . $row['data_utworzenia']. ' </td><td>' . $row['data_modyfikacji']. ' </td></tr>';
33     }
34 } else {
35     echo "Błąd podczas odczytywania danych z tabeli:" . mysqli_error($conn);
36 }
37 mysqli_close($conn);
38 -?>
```

# Pobieramy dane z bazy

```
8 <?php
9 //obiektoowo
10 $servername = "localhost";
11 $username = "root";
12 $password = "";
13 $dbname = "uczeno";
14 // tworzymy połączenie z baza danych
15 $conn = new mysqli($servername, $username, $password, $dbname);
16 // sprawdzamy połączenie
17 if ($conn->connect_error) {
18     die("Błąd połączenia : " . $conn->connect_error());
19 }
20 echo 'Połączono z baza danych<br> ';
21 //odczytujemy dane z tabeli:
22 $sql = "SELECT * from daneo ";
23
24 $wynik=$conn->query($sql);
25 if ($wynik->num_rows>0) {
26     echo 'dane zostały odczytane <br>';
27     echo '<table border="2">';
28     echo
29     '<tr><th>id</th><th>imie</th><th>nazwisko</th><th>email</th><th>data_utworzenia</th><th>data_modyfikacji
30     </th></tr>';
31     // output data of each row
32     while($row = $wynik->fetch_assoc()) {
33         echo '<tr><td>' . $row['id']. ' </td><td>' . $row['imie']. ' </td><td>' . $row['nazwisko'].
34         '</td><td>' . $row['email']. ' </td><td>' . $row['data_utworzenia']. ' </td><td>' . $row[
35         'data_modyfikacji']. ' </td></tr>';
36     }
37 } else {
38     echo "Błąd podczas odczytywania danych z tabeli:" . mysqli_error($conn);
39 }
40 $conn->close();
41 ?>
```

# Usuwanie rekord

```
8 <?php
9 //proceduralnie
10 $servername = "localhost";
11 $username = "root";
12 $password = "";
13 $dbname = "uczenn";
14 // tworzymy połączenie z baza danych
15 $conn = mysqli_connect($servername, $username, $password, $dbname);
16 // sprawdzamy połączenie
17 if (!$conn) {
18     die("Błąd połączenia : " . mysqli_connect_error());
19 }
20 echo 'Połączono z baza danych<br> ';
21 //usuwanie rekord
22 $sql = "DELETE from danep where id=1";
23 if(mysqli_query($conn,$sql)){
24     echo 'rekord został usuniety <br>';
25 } else
26 {echo 'błąd podczas usuwania rekordu ' .mysqli_error($conn);}
27 echo 'zawartość tabeli po usunięciu rekordu <br>';
28 //odczytujemy dane z tabeli:
29 $sql = "SELECT * from danep ";
30 $wynik=mysqli_query($conn, $sql);
31 if (mysqli_num_rows($wynik)>0) {
32     echo 'dane zostały odczytane <br>';
33     echo '<table border="2">';
34     echo '<tr><th>id</th><th>imie</th><th>nazwisko</th><th>email</th><th>data_utworzenia</th><th>data_modyfikacji</th></tr>';
35     // output data of each row
36     while($row = mysqli_fetch_assoc($wynik)) {
37         echo '<tr><td>' . $row['id']. ' </td><td>' . $row['imie']. ' </td><td>' . $row['nazwisko']. ' </td><td>' . $row['email'] .
38             '</td><td>' . $row['data_utworzenia']. ' </td><td>' . $row['data_modyfikacji']. ' </td></tr>';
39     }
40 } else {
41     echo "Błąd podczas odczytywania danych z tabeli:" . mysqli_error($conn);
42 }
43 mysqli_close($conn);
44 ?>
```

# Usuwamy rekord

```
8 <?php
9 //obiektoowo
10 $servername = "localhost";
11 $username = "root";
12 $password = "";
13 $dbname = "uczeno";
14 // tworzymy połączenie z baza danych
15 $conn = new mysqli($servername, $username, $password, $dbname);
16 // sprawdzamy połączenie
17 if ($conn->connect_error) {
18     die("Błąd połączenia : " . $conn->connect_error());
19 }
20 echo 'Połączono z baza danych<br> ';
21 //usuwamy rekord
22 $sql = "DELETE from daneo where id=5";
23 if($conn->query($sql)==TRUE){
24     echo 'rekord został usuniety';
25 } else
26 {echo 'błąd podczas usuwania rekordu ' . $conn->error;}
27 echo 'zawartość tabeli po usunięciu rekordu <br>';
28 //odczytujemy dane z tabeli:
29 $sql = "SELECT * from daneo ";
30 $wynik=$conn->query($sql);
31 if ($wynik->num_rows>0) {
32     echo 'dane zostały odczytane <br>';
33     echo '<table border="2">';
34     echo '<tr><th>id</th><th>imie</th><th>nazwisko</th><th>email</th><th>data_utworzenia</th><th>data_modyfikacji</th></tr>';
35     // output data of each row
36     while($row =$wynik->fetch_assoc()) {
37         echo '<tr><td>' . $row['id']. ' </td><td>' . $row['imie']. ' </td><td>' . $row['nazwisko']. ' </td><td>' . $row['email']
38             . ' </td><td>' . $row['data_utworzenia']. ' </td><td>' . $row['data_modyfikacji']. ' </td></tr>';
39     }
40 } else {
41     echo "Błąd podczas odczytywania danych z tabeli:" . mysqli_error($conn);
42 }
43 $conn->close();
44 ?>
```

# aktualizacja rekordu

```
8 <?php
9 //proceduralnie
10 $servername = "localhost";
11 $username = "root";
12 $password = "";
13 $dbname = "uczenn";
14 // tworzymy połączenie z baza danych
15 $conn = mysqli_connect($servername, $username, $password, $dbname);
16 // sprawdzamy połączenie
17 if (!$conn) {
18     die("Błąd połączenia : " . mysqli_connect_error());
19 }
20 echo 'Połączono z baza danych<br> ';
21 //aktualizujemy rekord
22 $sql = "UPDATE danep SET nazwisko='Andrzeiewska' WHERE id=3";
23 if(mysqli_query($conn,$sql)){
24     echo 'aktualizacja została wykonana <br>';
25 } else
26 {echo 'błąd podczas aktualizacji rekordu ' .mysqli_error($conn);}
27 echo 'zawartość tabeli po aktualizacji <br>';
28 //odczytujemy dane z tabeli:
29 $sql = "SELECT * from danep ";
30 $wynik=mysqli_query($conn, $sql);
31 if (mysqli_num_rows($wynik)>0) {
32     echo 'dane zostały odczytane <br>';
33     echo '<table border="2">';
34     echo '<tr><th>id</th><th>imie</th><th>nazwisko</th><th>email</th><th>data_utworzenia</th><th>data_modyfikacji</th></tr>';
35     // output data of each row
36     while($row = mysqli_fetch_assoc($wynik)) {
37         echo '<tr><td>' . $row['id']. ' </td><td>' . $row['imie']. ' </td><td>' . $row['nazwisko']. '</td><td>' . $row['email'] .
38             '</td><td>' . $row['data_utworzenia']. ' </td><td>' . $row['data_modyfikacji']. ' </td></tr>';
39     }
40 } else {
41     echo "Błąd podczas odczytywania danych z tabeli:" . mysqli_error($conn);
42 }
43 mysqli_close($conn);
44 ?>
```

# aktualizacja rekordu

```
8 <?php
9 //obiektoowo
10 $servername = "localhost";
11 $username = "root";
12 $password = "";
13 $dbname = "uczono";
14 // tworzymy połączenie z baza danych
15 $conn = new mysqli($servername, $username, $password, $dbname);
16 // sprawdzamy połączenie
17 if ($conn->connect_error) {
18     die("Błąd połączenia : " . $conn->connect_error());
19 }
20 echo 'Połączono z baza danvch<br> ';
21 //aktualizujemy rekord
22 $sql = "UPDATE dane SET nazwisko='Andrzejewska' WHERE id=3";
23 if($conn->query($sql)==TRUE) {
24     echo 'rekord został zaktualizowany';
25 } else
26 {echo 'błąd podczas aktualizacji ' . $conn->error;}
27 echo 'zawartość tabeli poaktualizacji rekordu <br>';
28 //odczytujemy dane z tabeli:
29 $sql = "SELECT * from daneo ";
30 $wynik=$conn->query($sql);
31 if ($wynik->num_rows>0) {
32     echo 'dane zostały odczytane <br>';
33     echo '<table border="2">';
34     echo '<tr><th>id</th><th>imie</th><th>nazwisko</th><th>email</th><th>data_utworzenia</th><th>data_modyfikacji</th></tr>';
35     // output data of each row
36     while($row =$wynik->fetch_assoc()) {
37         echo '<tr><td>' . $row['id']. ' </td><td>' . $row['imie']. ' </td><td>' . $row['nazwisko']. ' </td><td>' . $row['email'].
38             '</td><td>' . $row['data_utworzenia']. ' </td><td>' . $row['data_modyfikacji']. ' </td></tr>';
39     }
40 } else {
41     echo "Błąd podczas odczytywania danych z tabeli:" . mysqli_error($conn);
42 }
43 $conn->close();
44 ?>
```