# DATABASES

Date:       30/06/2016

Student:    Kasia Pieniazek

Teacher:    Alexandru Gheorghiasa

# ABOUT

General idea: Admin Panel based on the CRUD app.
Usability: saving data about members (in this case
about international participats in some event).

In my project administrator can:
- login to his own panel
- create: first name, last name, age  and nationality
- read: using the table administrator has a clear view
        on the data
- updte: opportunity to change name, last name or age
- delete: remove user from the database

# MY ACHIVEMENTS

1. What I wanted to achieve?
First of all, I wanted to gather my knowledge which
I gained during this semester. And in the result build
a functional and working system which will cover all
project requirements.

My assumed achievementy:
- understanding of relations beetween php and sql
  (possibility of saving,displaying, deleting etc.
   new member in the database)
- get ready with project before deadline

2. Why I wanted to build it?
There is no special excuse about topic of my project.
I wanted to came up with something simple and clear
which will cover all requirements and at the same time
does not complicate my project. And I think that this
system is very common so I am pretty much sure that i
can use it somewhere in the future.

# RISK ANALYSIS

1. The time constraint
2. Lack ofknowledge
3. Another project - time preassure

What could go wrong?
That I will not finish my project.
Another thing - that I could get some probles in case
of debugging my code.

# TIME TABLE:

I've started working on the project right after my Web
Development exam. So I splited my work between days.

19-20.06  - creating database consists of 3 tables and
            making relations

21-24.06 - all neccessary php files, connecting with
           dataase

26-30.06 - saving data in the database, deleting,
           receiving data from the database, update

31.06    - last changes, report

1.06     - hand in before deadline

# USED TOOLS

MAMP :

- standard for Mac OS X operating system
- stands for Mac, Apache, MySQL, PHP - can be easy enable or disable
- everything is in a one folder,
this makes backing up very easy

SUBLIME TEXT 2:

- Multiple cursors which let user write or edit in multiple places in a docu ment at the same time
- Easy way to intend your code
- Package control: This add-on let's
user install plugins within seconds directly from the editor

MySQL WORKBENCH:

- allows to edit db schema as a diagram and the export as sql code
- vizualizing DB  structure (good especially for beginners)

# CODE ANALYSIS

DB connection: I used both: PDO and MYSQLI

I was not sure which connection will suits me more that's why I practised both of them. I assumed that for the project purpose it will be totally fine. And I realize that in the future during other projects I will stick till one.
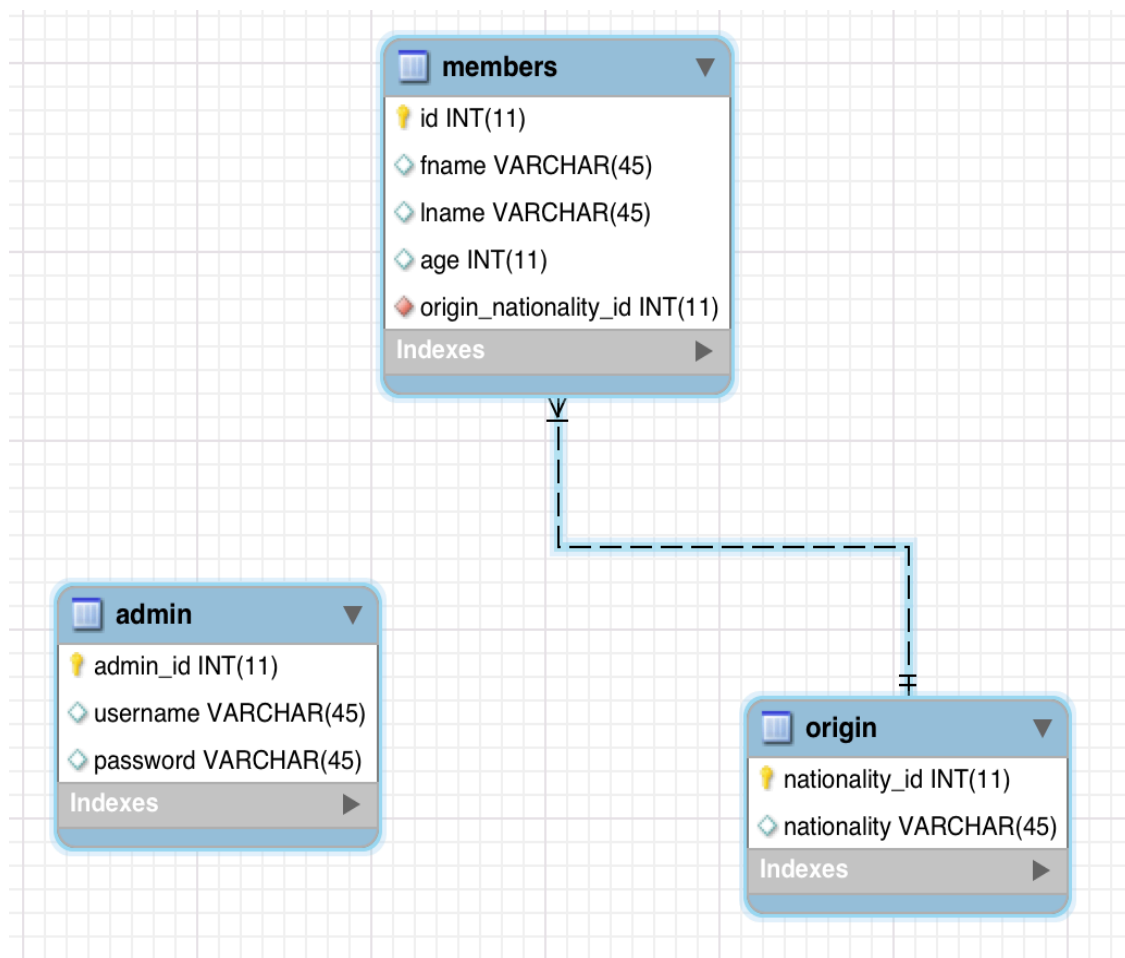
What could be done better?
I think my code structure. Sometimes it is a little bit messy so I have someting to imporove.

What was the most difficult part of the system?

I think the most part was to combine everything togeth-
er, to make it work. Or even harder was debugging my
own code, cause sometimes I could not find my own mis-
takes. The hard part for me was to insert data into ta-
ble with foreign key.

# DATABASE DESIGN

**members**
- id INT(11)
- fname VARCHAR(45)
- lname VARCHAR(45)
- age INT(11)
- origin_nationality_id INT(11)

Indexes

**admin**
- admin_id INT(11)
- username VARCHAR(45)
- password VARCHAR(45)

Indexes

**origin**
- nationality_id INT(11)
- nationality VARCHAR(45)

Indexes

# STRUCTURE OF THE DB

Database is normalized and contains 3 separated
tables: admin, origin and members. Each table is
implemented with AUTO_INCREMENT -
reason: to add next rows with different ids.

1. admin:
does not have any connections. It is only for the
admin to have a view inside the page after login

2. origin:
store nationalities

3. members:
store information about members and their's origin

Relation: origin -> members : one to many.
I decided to use this connection because following the
logic each member can have only one nationality.

# SQL

```php
$oQuery = $oDb->prepare("SELECT * FROM origin");
$oQuery->execute();
$aResults = $oQuery->fetchAll(PDO::FETCH_ASSOC);
for( $i = 0; $i < count($aResults); $i++ )
{
  echo ( "<option value=".$aResults[$i]['nationality_id'].">
  <p class='options'>".$aResults[$i]['nationality']."</p>
  </option>" );

}
```

-------------------------------------------------------------

1st part :
make a selector which display all data /nationalities
/ from database. First of all, I create a query which
connects with db and prepare statment to wchi is gon-
na be executed. Then I make a variable which wil store
all data from database as an array.Then I loop through
an array to make it possible to display all data.
At the end i put all data inside the option. I set up
also a value then I will be able to insert into
database.

--------------------------------------------------------

```php
<?php
include('connect.php');
//if (isset($_GET["page"])) { $page  = $_GET["page"]; } else { $page=1; };
//$start_from = ($page-1) * 6;
$result = $db->prepare("SELECT * FROM members INNER JOIN origin ON members.
origin_nationality_id = origin.nationality_id ");
result->execute();
for($i=0; $row = $result->fetch(); $i++){
?>
      <tr class="record">
      <td><?php echo $row['fname']; ?></td>
      <td><?php echo $row['lname']; ?></td>
      <td><?php echo $row['age']; ?></td>
      <td><?php echo $row['nationality']; ?></td>
      <td><a href="editform.php?id=<?php echo $row['id']; ?>"> edit </a> |
<a href="delete.php?id=<?php echo $row['id']; ?>"> delete </a></td>
                  </tr>
                  <?php
                      }
                  ?>
```

I had to select from table which has one foreig key to make it work I used INNER JOIN.

"SELECT * FROM members INNER JOIN origin ON members. origin_nationality_id = origin.nationality_id "

First of all I select all data from table called members. It contains (id, fname, age, origin_nationality_id). I did not want to display origin_nationality_id but the value which is hidden behind. To make it, I joined another table (using INNER JOIN) called origin and I needed to assing member's id to nationality's id. It is all done by using 'ON'. SO I take a name of the table which has a foreign key and it's id and then assign to another table.

# CONCLUSION

What I have reached?
- First of all, useful knowledge which I will be able to use in the future
- working system
- satisfaction and motivation for the
next projects
- experience
- stubbornness, even if I had some
dif culties I did not give up

What can you do better?
Organize my code better, and do not the mess there. I can improve it by coding more clearly, using comments, indentation and correct structure.

Was it successful?
I can truly say that yes. I have done what I planned. Maybe I could do more but since I have just started my web devel- opment adventure I glad and satisfy. Most things are working without any errors and colidations. But I think in the future I need to set the bar higher ;)