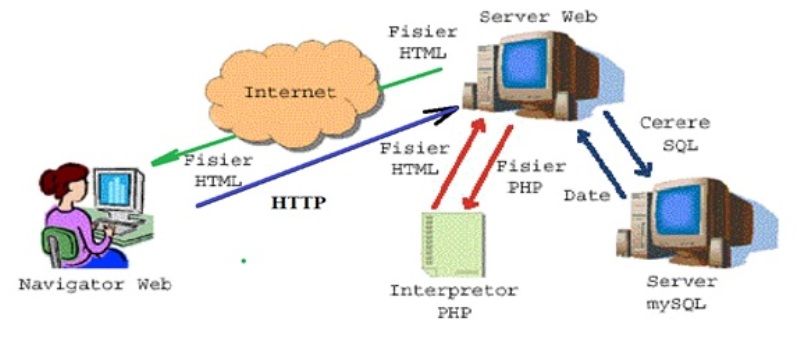
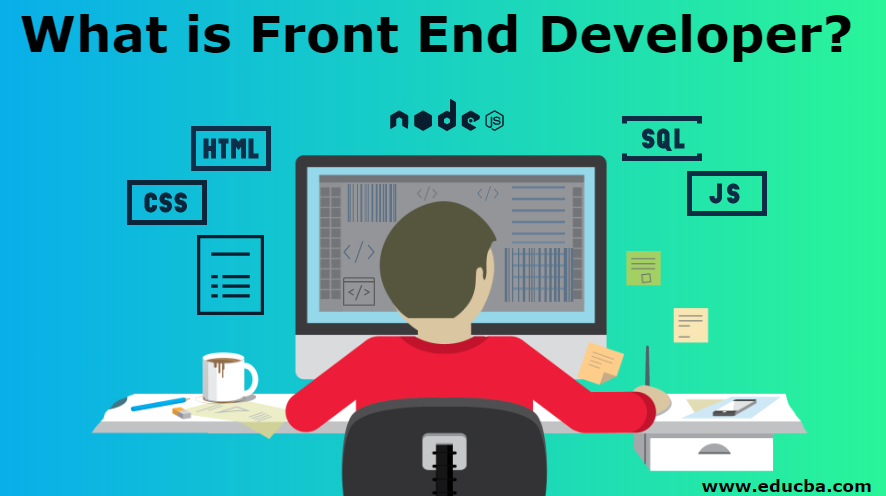
**LUCRARE DE LABORATOR NR 4**

**PROIECTAREA BD PENTRU SARCINA/PROIECTUL PRIMIT. LUCRUL CU MYSQL, HTML, CSS, JS (după caz) ȘI PHP PENTRU CREAREA COMPONENTEI *BACK-END* A APLICAȚIEI WEB.**

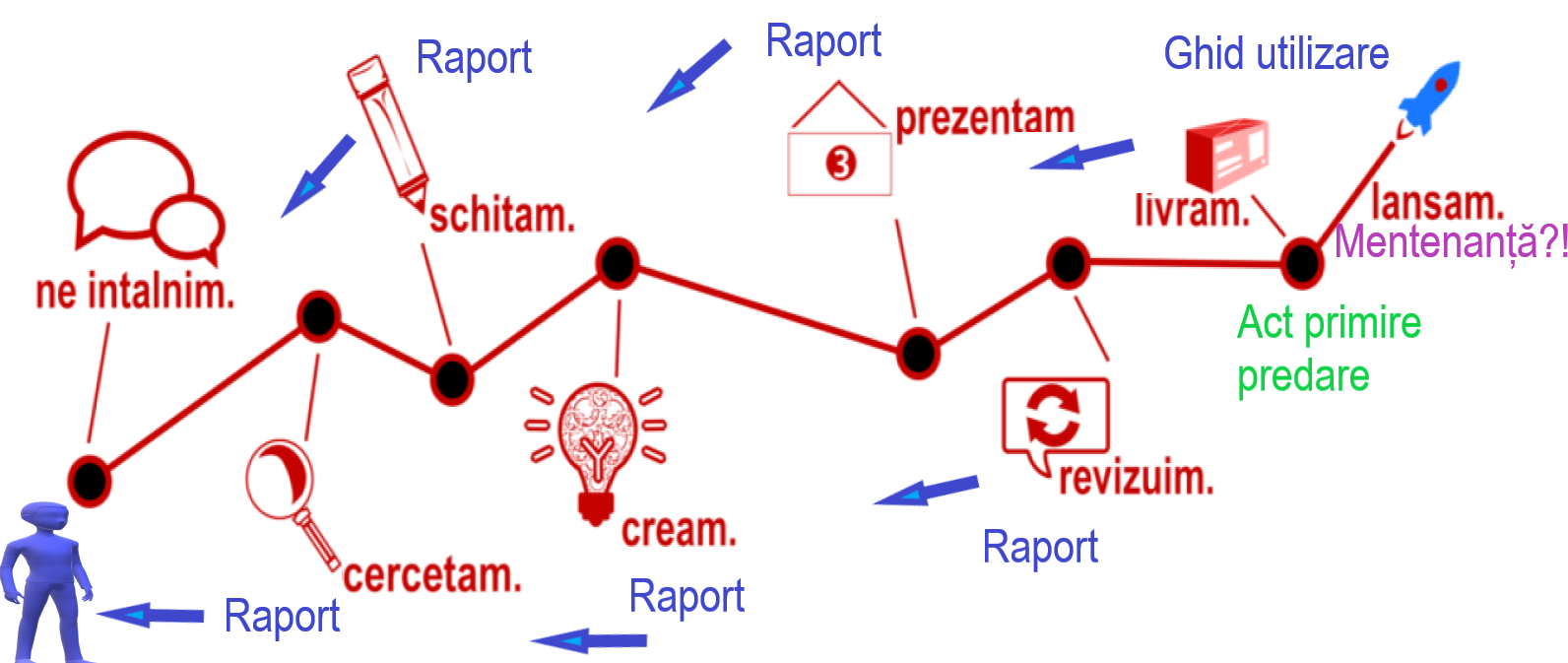
****

***PREZENTAREA COMPONENTEI "FRONT-END" A PROIECTULUI-SARCINĂ PRIMIT LA LL1, PASII 1-7/ REAMINTIM!!/***

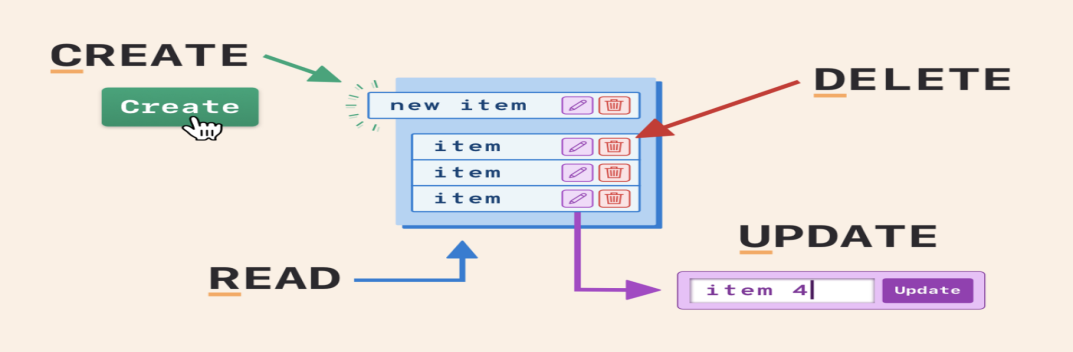


**LUCRARE DE LABORATOR NR 4 se realizează în practică prin intermediul a 3 componente:**

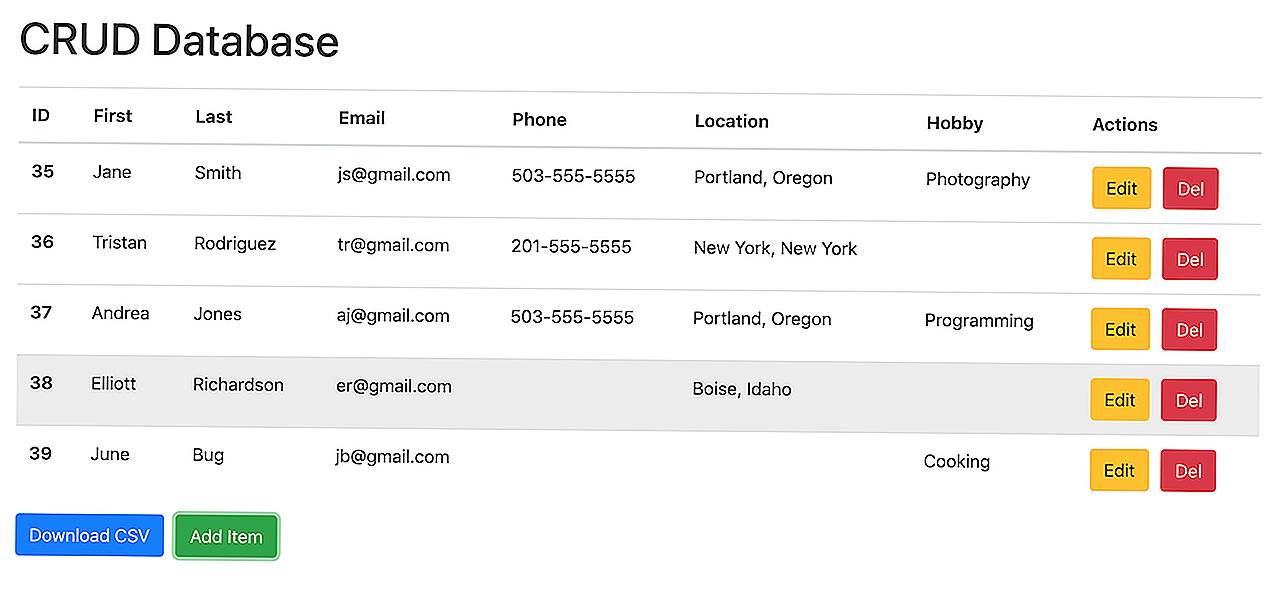
1. ***PROIECTAREA BD PENTRU PROIECTUL - SARCINĂ PRIMITĂ LA LL1 (UTILIZÎND ABORADAREA „INPUT-OUTPUT”) SI PREGĂTIREA STRUCTURII BD PENTRU LUCRUL CU MYSQL, HTML, CSS, JS (după caz) ȘI PHP PENTRU CREAREA COMPONENTEI BACK-END A APLICAȚIEI WEB***



1. ***UTILIZAREA  XAMPP ȘI MYSQL, PENTRU A CREA BD PROIECTATĂ A ROIECTULUI - SARCINĂ PRIMITĂ LA LL1 ÎN REGIM CONSOLĂ ȘI EFECTUAREA ÎN EA A MANIPULĂRILOR CU DATELE PRIN INTERMEDIUL OPERAȚIILOR DE TIP CRUD***

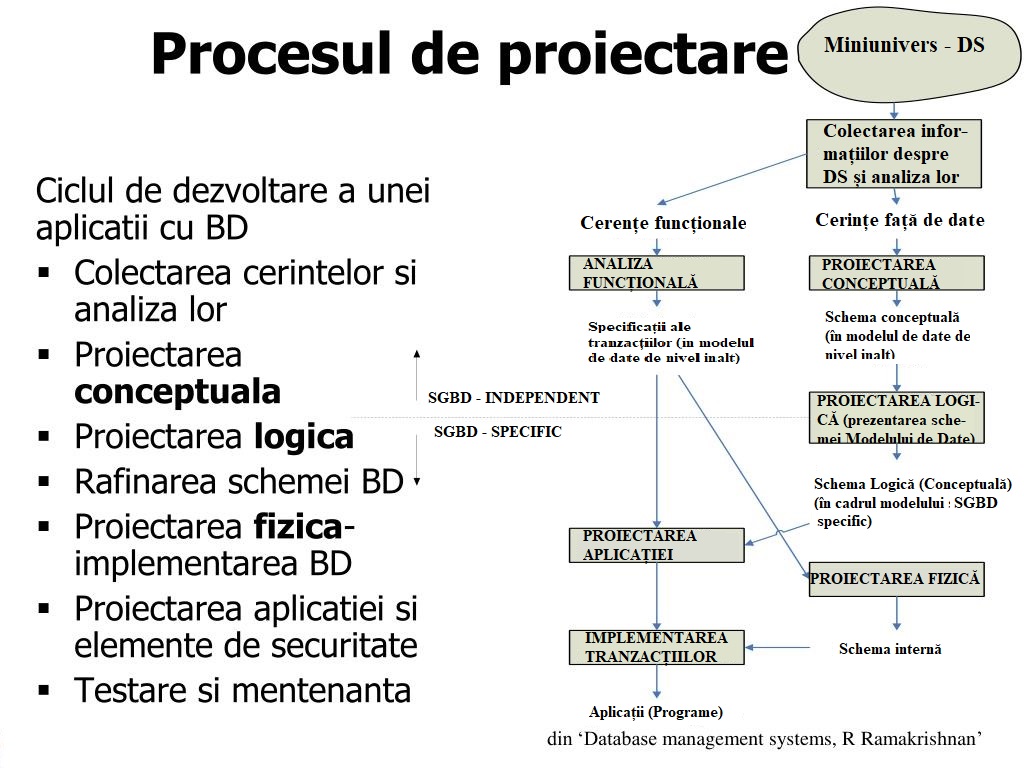


1. ***UTILIZAREA  XAMPP ȘI MYSQL, PENTRU A CREA BD PROIECTATĂ A ROIECTULUI - SARCINĂ PRIMIT LA LL1 ÎN REGIM MENIU SQL DIN PHPMYADMIN ȘI EFECTUAREA ÎN EA A MANIPULĂRILOR CU DATELE, A OPERAȚIILOR DE TIP CRUD UTILIZÂND* *HTML, CSS, JS (după caz) ȘI PHP PENTRU CREAREA COMPONENTEI BACK-END A APLICAȚIEI WEB***



**LABORATOR NR\_\_4 , MYSQL (BD) PARTEA I**

***PROIECTAREA BD PENTRU PROIECTUL - SARCINĂ PRIMIT LA LL1 (UTILIZÎND ABORADAREA „INPUT-OUTPUT”) SI PREGĂTIREA STRUCTURII BD PENTRU LUCRUL CU MYSQL, HTML, CSS, JS (după caz) ȘI PHP PENTRU CREAREA COMPONENTEI BACK-END A APLICAȚIEI WEB***



**EXEMPLU**

***Sarcina: Producătorii de lapte din Republica Moldova***

**I. Proiectarea *Input-Output*:**

***1.1. Modelul Conceptual al Domeniului de Studiu***

***1.2. Modelul Logic***

**„Locații”**

|  |  |
| --- | --- |
| Id\_loc | Nume\_loc |

**„Producători”**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| IdProd | NumeProd | Data\_fondare | Director | Id\_loc |

**„Cumpărători”**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| IdCump | NumeCump | Id\_loc | Adresa | Tip |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| IdProdus | NumePr | IdProd | Data\_pr | Termen\_val | Val\_energ |

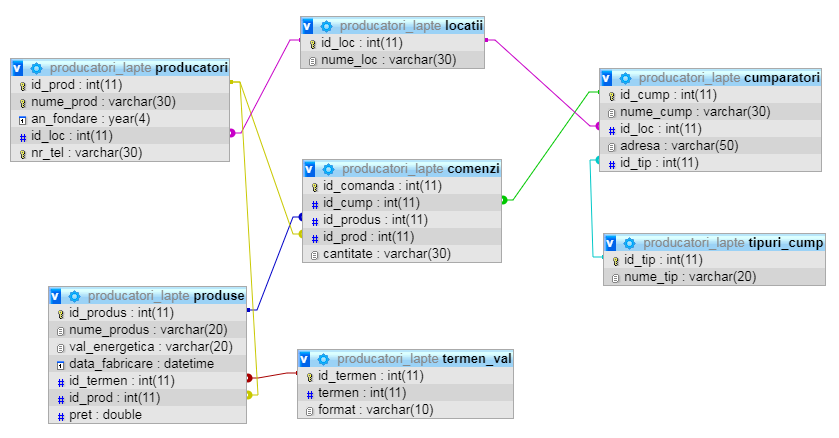
**„Produse„ „Comenzi”**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| IdCom | IdCump | IdProdus | Cant | Preț |

***1.3. Machetul Bazei de Date:***

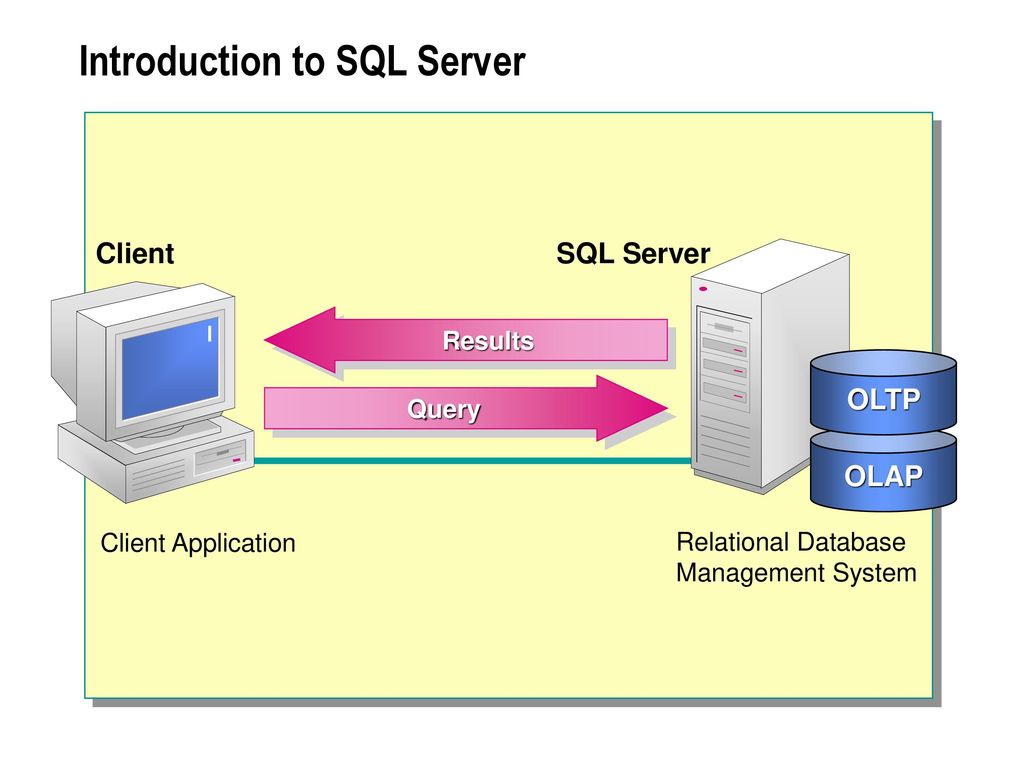
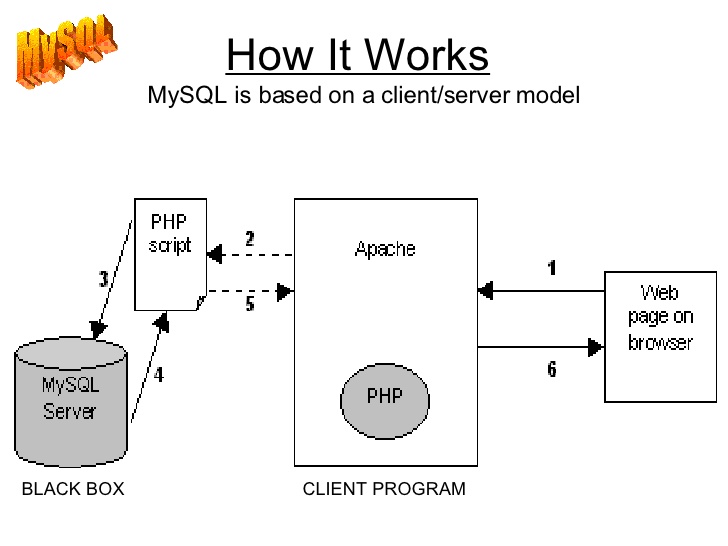
|  |  |  |  |
| --- | --- | --- | --- |
| **Cumpărători** | IdCump |  | PRIMARY KEY NOT NULL |
| NumeCump | VARCHAR(20) |  |
| Id\_loc | INT | FOREIGN KEY |
| Adresa | VARCHAR(30) |  |
| Tip | VARCHAR(10) |  |
| **Locații** | Id\_loc | INT | PRIMARY KEY NOT NULL |
| Nume\_loc | VARCHAR(10) |  |
| **Producători** | IdProd | INT | PRIMARY KEY NOT NULL |
| NumeProd | VARCHAR(30) |  |
| Data\_fondare | DATE |  |
| Director | VARCHAR(30) |  |
| Id\_loc | INT | FOREIGN KEY |
| **Produs** | **Id\_produs** | INT | PRIMARY KEY NOT NULL |
| Nume\_produs | VARCHAR(20) |  |
| IdProd | INT | FOREIGN KEY |
| Data\_pr | DATE |  |
| Termen\_val | VARCHAR(15) |  |
| Val\_energ | VARCHAR(15) |  |
| **Comenzi** | IdCom | INT | PRIMARY KEY NOT NULL |
| IdCump | INT | FOREIGN KEY |
| **IdProdus** | INT | FOREIGN KEY |
| Cant | FLOAT |  |
| Preț | Money |  |

***1.4. Schema Bazei de Date:***



**LABORATOR NR\_\_4 ,  MYSQL (BD) PARTEA II**

***UTILIZAREA  XAMPP ȘI MYSQL, PENTRU A CREA BD PROIECTATĂ A ROIECTULUI - SARCINĂ PRIMITĂ LA LL1 ÎN REGIM CONSOLĂ ȘI EFECTUAREA ÎN EA A MANIPULĂRILOR CU DATELE PRIN INTERMEDIUL OPERAȚIILOR DE TIP CRUD***

****

**DESCRIERE SUCCINĂ A PROIECTULUI**

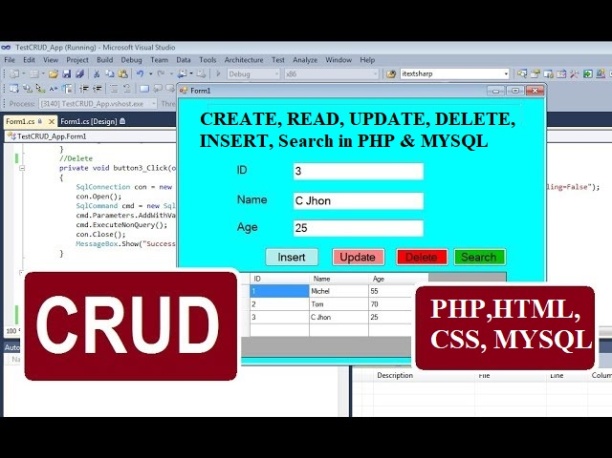
Pentru LL4, fiecare student are un set de **3 sarcini** la compartimentul MYSQL, pentru Tema/subiectul selectată/selectat în lucrarea precedentă.

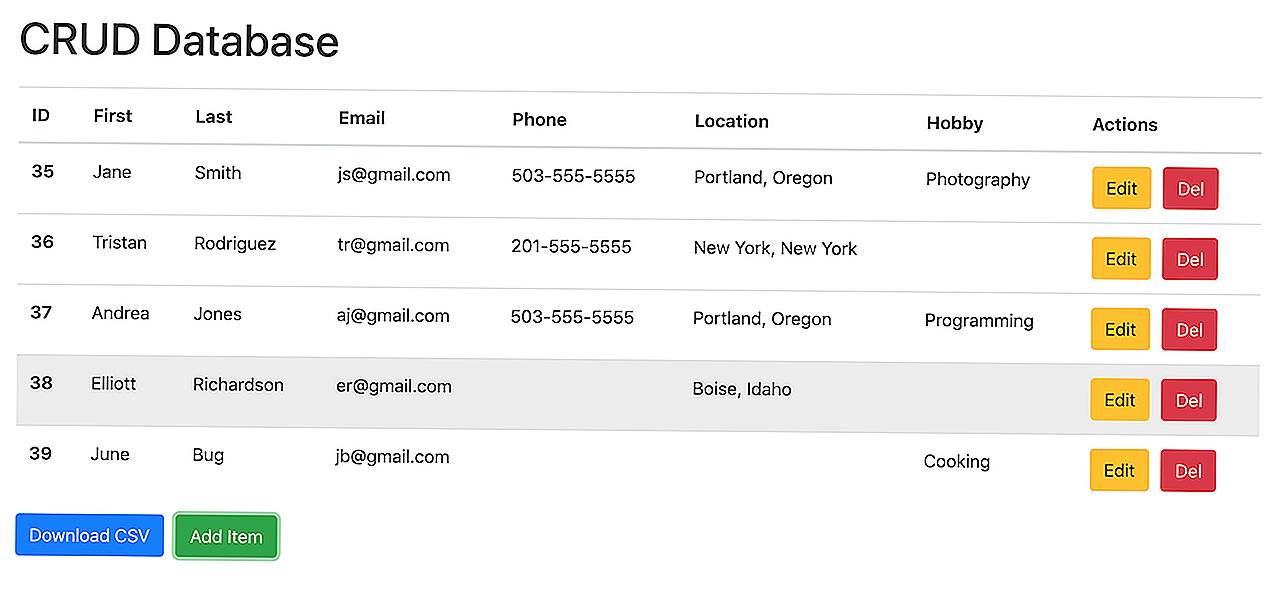
Fiecare sarcină din cele **3**, este un mod de executare a ceea ce trebuie să fie efectuat la etapa curentă, cu referință la subiectul Lucrării de laborator date.

**Notă:** **Se va aprecia in mod DEOSEBIT, propria viziune și prezentare a Modelului Conceptual, Modelului Logic și macheta BD (*elaborate prin paradigma Input -> Output*) și deci a schemei ER a BD, precum și realizarea unui model cât mai “larg/relevant” de descriere prin date a subiectului ales si deci si pregătirea suportului pentru a obține posibilitatea generării unui spectru “larg/relevant” de informații la procesarea lor prin intermediul aplicației Web, realizată in cele ce urmează prin instrumentele HTML, CSS (JS după caz) și PHP.**

**LABORATOR NR\_\_4 ,  MYSQL (BD) PARTEA III**

***UTILIZAREA  XAMPP ȘI MYSQL, PENTRU A CREA BD PROIECTATĂ A ROIECTULUI - SARCINĂ PRIMITĂ LA LL1 ÎN REGIM MENIU SQL DIN PHPMYADMIN ȘI EFECTUAREA ÎN EA A MANIPULĂRILOR CU DATELE, A OPERAȚIILOR DE TIP CRUD UTILIZÂND* *HTML, CSS, JS (după caz) ȘI PHP PENTRU CREAREA COMPONENTEI BACK-END A APLICAȚIEI WEB***

****



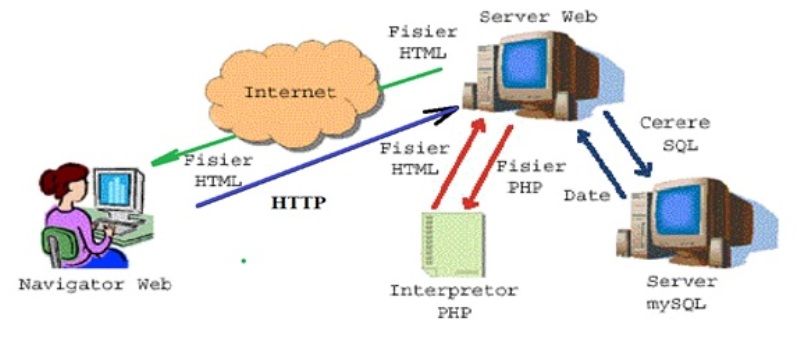
**REALIZAREA PARTEA III , A LUCRĂRII DE LABORATOR NR.4 ESTE EFECTUATĂ ÎN DOUĂ ETAPE CE URMEAZĂ:**

1. ***UTILIZAREA  XAMPP ȘI MYSQL, PENTRU A CREA BD PROIECTATĂ A ROIECTULUI - SARCINĂ PRIMIT LA LL1 ÎN REGIM MENIU SQL DIN PHPMYADMIN***

MYSQL este un  SGBD produs de compania suedeza MySql AB   și distribuit sub Licența Publică Generala GNU. Este cel mai popular SGBD open-source la ora actuală, fiind o componentă cheie a stivei LAMP  (Linux, Apache, MySQL, PHP).

Vom menționa că BD creată în regim consolă, poate fi creată și prin intermediul Phpmyadmin a pachetului integrat XAMPP, utilizînd ***punctul de meniu SQL*** al Phpmyadmin, precum și alte instrumente ale lui. Acest exercitiu poate fi realizat prin crearea unei BD ce o dublează pe precedenta, creată în regim consolă.

**ÎN CONTINUARE, luînd în considerare că BD este creată, indiferent de modalitățile utilizate, CONSOLĂ SAU MENIUL SQL DIN PHPMYADMIN, pentru a crea o APLICAȚIE WEB, ce ar asigura USER-ului acces la BD și i-ar oferi modalitatea de a obține informațiile necesare pentru managementul procesului decizional a sarcinii formulate, este necesar de utilizat limbajul de programare PHP, pentru a asigura legătura Serverului Web cu Serverul unde este stocată BD, adică serverul MySql.**

****

## Succint. Ce este PHP? PHP este un limbaj de script cu destinație generală a cărui funcție principală, dar nu și singura, este procesarea datelor pe un server web și implementarea lor în cod HTML. Permite procesarea rapidă și încărcarea paginilor și se execută pe aproape toate sistemele de operare. PHP funcționează fiind recunoscut de către serverul web și generând un cod HTML sau altă ieșire pe care clienții (în principal browserele web) le pot recunoaște.

**Avantajele utilizării programării PHP**

Acest limbaj de programare este foarte popular deoarece permite și programatorilor cu mai puțină experiență să obțină performanțe în muncă. Datorită numeroaselor avantaje și ușurinței cu care se utilizează, milioane de site-uri și aplicații web sunt create cu ajutorul acestui limbaj.

**Ușor de utilizat** – limbajul este ușor de învățat și de implementat, iar actualizarea site-ului și adăugarea noilor opțiuni este relativ simplă. O circumstanță atenuantă suplimentară este suportul pentru cele mai populare CMS-uri (Content management system), așa cum sunt WordPress, Joomla sau Drupal. 

**Este gratuit** – este vorba despre o soluție open source și nu este necesară cumpărarea unui software.

**Compatibilitate între platforme**– PHP suportă toate sistemele de operare mai cunoscute, iar site-urile construite cu ajutorul acestui program se pot integra ușor pe marea majoritate a serverelor.

**Popularitatea**– având în vedere că este momentan vorba despre cel mai popular limbaj de programare din lume, există un număr mare de persoane cu care vă puteți consulta. În acest mod, puteți căuta pe Google soluții pentru aproape orice problemă pe care o întâmpinați. Există o mare probabilitate ca cineva să fi avut aceeași problemă.

**Cum funcționează PHP?**

Deoarece PHP este un limbaj cu scop general, cu el se pot crea și scripturi care au un scop general. Acest limbaj este orientat în primul rând spre mediul internetului, ceea ce îl face ideal pentru crearea paginilor web dinamice.

Un astfel de tip de site a apărut odată cu dezvoltarea internetului, când a devenit o necesitate comunicarea activă cu vizitatorii. Deși există și alte limbaje de programare care ajută la crearea site-urilor dinamice, PHP a atins cel mai mare succes.

Având în vedere că este orientat spre web, acest limbaj conține multe opțiuni pentru crearea site-urilor. PHP permite interacționarea cu formele HTML, dar și procesarea informațiilor pe care utilizatorul le solicită. De asemenea, este posibilă comunicarea cu bazele de date în vederea stocării informațiilor despre utilizator. Pe lângă acest lucru, este posibil să se genereze și să se afișeze site-uri sigure prin introducerea numelui de utilizator și a parolei.

Trebuie avut în vedere că activitatea PHP se execută direct pe server, ceea ce înseamnă că este imposibilă interacțiunea cu computerul. De aceea, anumite opțiuni legate de spațiu sunt imposibil de realizat. Asta înseamnă că prin PHP nu puteți să influențați rezoluția ecranului, comportamentul mouse-ului… 

De aceea, pe lângă PHP se utilizează deseori și JavaScript. Acest script nu poate accesa serverele, dar de aceea poate face lucruri pe care PHP nu le poate face. Există mai multe domenii în care acest limbaj de programare s-a dovedit ca fiind excelent pentru:

* **Bazele de date** – pentru site-uri care se bazează pe bazele de date, PHP este un instrument de lucru ideal. Scripturile create cu acest limbaj pot găzdui date în bază și pot permite utilizatorului accesul la ele.
* **Pentru mai multe detalii cu referință la PHP și MySql, accesați acest Link,**

[**https://web.ceiti.md/lesson.php?id=7**](https://web.ceiti.md/lesson.php?id=7)

* + ***EFECTUAREA ÎN BD A MANIPULĂRILOR CU DATELE, A OPERAȚIILOR DE TIP CRUD UTILIZÂND* *HTML, CSS, JS (după caz) ȘI PHP PENTRU CREAREA COMPONENTEI BACK-END A APLICAȚIEI WEB***

Un exemplu similar de componente CRUD poate fi precăutat aici:

<https://phpgurukul.com/crud-operation-using-php-and-mysqli/>

**Continutul acestui material este prezentat în Anexa 1 de mai jos.**

**Anexa 1**

**CRUD operation using PHP and MySQLi**

BY [ANUJ KUMAR](https://phpgurukul.com/author/anujk305/)

In this tutorial, we will learn how to create CRUD operation using [PHP](https://phpgurukul.com/php-basics/)and [MySQLi](https://phpgurukul.com/mysqli-procedural-functions/).

CRUD Stands for create, read, update and delete record in the database.

[View Demo](https://phpgurukul.com/demos/php-mysqli-crud/)

**Files includes in this tutorials**

* **phpcrud.sql:** Contain the database table structure.
* **dbconnection.php:** Used for database connection.
* **index.php**: Used to fetch the record from database.
* **read.php:**Used to fetch the record of particular user.
* **edit.php:** Used to edit the record.
* **insert.php:** Used to insert the new record.

Step 1– Create a database

Open browser type http://localhost/phpmyadmin, create a database named ‘phpcrud’. After creating database run the SQL script or import the SQL file.

[MySQL Table](https://phpgurukul.com/how-to-create-database-and-tables-in-mysql/) **tblusers**structure

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9 | CREATE TABLE `tblusers` (    `ID` int(10) NOT NULL,    `FirstName` varchar(200) DEFAULT NULL,    `LastName` varchar(200) DEFAULT NULL,    `MobileNumber` bigint(10) DEFAULT NULL,    `Email` varchar(200) DEFAULT NULL,    `Address` mediumtext DEFAULT NULL,    `CreationDate` timestamp NOT NULL DEFAULT current\_timestamp() ON UPDATE current\_timestamp()  ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4; |

**Step 2– Create a database connection file(dbconnection.php)**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7 | <?php  $con=mysqli\_connect("localhost", "root", "", "phpcrud");  if(mysqli\_connect\_errno())  {  echo "Connection Fail".mysqli\_connect\_error();  }  ?> |

**Step 3– Create a**[**HTML form**](https://phpgurukul.com/html-forms/)**for data insertion**

|  |  |
| --- | --- |
| 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 | <form  method="POST">  <h2>Fill Data</h2>  <p class="hint-text">Fill below form.</p>  <div class="form-group">    <div class="row">  <div class="col"><input type="text" class="form-control" name="fname" placeholder="First Name" required="true"></div>  <div class="col"><input type="text" class="form-control" name="lname" placeholder="Last Name" required="true"></div>  </div>  </div>    <div class="form-group">  <input type="text" class="form-control" name="contactno" placeholder="Enter your Mobile Number" required="true" maxlength="10" pattern="[0-9]+">          </div>    <div class="form-group">  <input type="email" class="form-control" name="email" placeholder="Enter your Email id" required="true">    </div>    <div class="form-group">  <textarea class="form-control" name="address" placeholder="Enter Your Address" required="true"></textarea>  </div>    <div class="form-group">  <button type="submit" class="btn btn-success btn-lg btn-block" name="submit">Submit</button>  </div>  </form> |

**Step 4– Now write PHP Code for data insertion** and put this code at the top of the insert.php file.

Once the user filled all the data and click on the submit button then data will be saved in the database using the 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 | <?php  //Databse Connection file  include('dbconnection.php');  if(isset($\_POST['submit']))    {     //getting the post values      $fname=$\_POST['fname'];      $lname=$\_POST['lname'];      $contno=$\_POST['contactno'];      $email=$\_POST['email'];      $add=$\_POST['address'];      // Query for data insertion       $query=mysqli\_query($con, "insert into tblusers(FirstName,LastName, MobileNumber, Email, Address) value('$fname','$lname', '$contno', '$email', '$add' )");      if ($query) {      echo "<script>alert('You have successfully inserted the data');</script>";      echo "<script type='text/javascript'> document.location ='index.php'; </script>";    }    else      {        echo "<script>alert('Something Went Wrong. Please try again');</script>";      }  }  ?> |

**Step 5– Read / Fetch the record from the database(index.php)**

In this step we will fetch all data from the database.

|  |  |
| --- | --- |
| 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 | <table class="table table-striped table-hover">                  <thead>                      <tr>                          <th>#</th>                          <th>Name</th>                          <th>Email</th>                          <th>Mobile Number</th>                          <th>Created Date</th>                          <th>Action</th>                      </tr>                  </thead>                  <tbody>                       <?php  $ret=mysqli\_query($con,"select \* from tblusers");  $cnt=1;  $row=mysqli\_num\_rows($ret);  if($row>0){  while ($row=mysqli\_fetch\_array($ret)) {    ?>  <!--Fetch the Records -->                      <tr>                          <td><?php echo $cnt;?></td>                          <td><?php  echo $row['FirstName'];?> <?php  echo $row['LastName'];?></td>                          <td><?php  echo $row['Email'];?></td>                          <td><?php  echo $row['MobileNumber'];?></td>                          <td> <?php  echo $row['CreationDate'];?></td>                          <td>  <a href="read.php?viewid=<?php echo htmlentities ($row['ID']);?>" class="view" title="View" data-toggle="tooltip"><i class="material-icons">&#xE417;</i></a>  <a href="edit.php?editid=<?php echo htmlentities ($row['ID']);?>" class="edit" title="Edit" data-toggle="tooltip"><i class="material-icons">&#xE254;</i></a>                       <a href="index.php?delid=<?php echo ($row['ID']);?>" class="delete" title="Delete" data-toggle="tooltip" onclick="return confirm('Do you really want to Delete ?');"><i class="material-icons">&#xE872;</i></a>  </td>  </tr>  <?php  $cnt=$cnt+1;  } } else {?>  <tr>  <th style="text-align:center; color:red;" colspan="6">No Record Found</th>  </tr>  <?php } ?>    </tbody>  </table> |

**Step 6–Read / Fetch the particular record (read.php)**

create read.php file. For fetching a record we have to get the row id of that record and store in $vid. We access the $\_GET[‘viewid’] variable to do it.  
Code for gets a record based on the given id.

|  |  |
| --- | --- |
| 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 | <table cellpadding="0" cellspacing="0" border="0" class="display table table-bordered" id="hidden-table-info">    <tbody>  <?php  $vid=$\_GET['viewid'];  $ret=mysqli\_query($con,"select \* from tblusers where ID =$vid");  $cnt=1;  while ($row=mysqli\_fetch\_array($ret)) {    ?>  <tr>      <th>First Name</th>      <td><?php  echo $row['FirstName'];?></td>      <th>Last Name</th>      <td><?php  echo $row['LastName'];?></td>    </tr>    <tr>      <th>Email</th>      <td><?php  echo $row['Email'];?></td>      <th>Mobile Number</th>      <td><?php  echo $row['MobileNumber'];?></td>    </tr>    <tr>      <th>Address</th>      <td><?php  echo $row['Address'];?></td>      <th>Creation Date</th>      <td><?php  echo $row['CreationDate'];?></td>    </tr>  <?php  $cnt=$cnt+1;  }?>    </tbody>  </table> |

**Step 7 –Edit/ Update the particular record (edit.php)**

**Step 7.1 Fetch the data in the HTML Form.**

create edit.php file. For updating a record we have to get the row id of that record and store in $eid. We access the $\_GET[‘editid’] variable to do it.  
Code for gets a record based on the given id. Through this way, we can get data autofill-data in HTML Form.

|  |  |
| --- | --- |
| 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 | <form  method="POST">  <?php  $eid=$\_GET['editid'];  $ret=mysqli\_query($con,"select \* from tblusers where ID='$eid'");  while ($row=mysqli\_fetch\_array($ret)) {  ?>  <h2>Update </h2>  <p class="hint-text">Update your info.</p>  <div class="form-group">  <div class="row">  <div class="col"><input type="text" class="form-control" name="fname" value="<?php  echo $row['FirstName'];?>" required="true"></div>  <div class="col"><input type="text" class="form-control" name="lname" value="<?php  echo $row['LastName'];?>" required="true"></div>  </div>          </div>    <div class="form-group">  <input type="text" class="form-control" name="contactno" value="<?php  echo $row['MobileNumber'];?>" required="true" maxlength="10" pattern="[0-9]+">          </div>    <div class="form-group">  <input type="email" class="form-control" name="email" value="<?php  echo $row['Email'];?>" required="true">  </div>    <div class="form-group">  <textarea class="form-control" name="address" required="true"><?php  echo $row['Address'];?></textarea>  </div>  <?php  }?>  <div class="form-group">  <button type="submit" class="btn btn-success btn-lg btn-block" name="submit">Update</button>          </div>      </form> |

**Step 7.1 Code for update the particular record. Put this code on the top of the edit.php**

|  |  |
| --- | --- |
| 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 | <?php  //Database Connection  include('dbconnection.php');  if(isset($\_POST['submit']))    {     $eid=$\_GET['editid'];  //Getting Post Values      $fname=$\_POST['fname'];      $lname=$\_POST['lname'];      $contno=$\_POST['contactno'];      $email=$\_POST['email'];      $add=$\_POST['address'];        //Query for data updation       $query=mysqli\_query($con, "update  tblusers set FirstName='$fname',LastName='$lname', MobileNumber='$contno', Email='$email', Address='$add' where ID='$eid'");        if ($query) {      echo "<script>alert('You have successfully update the data');</script>";      echo "<script type='text/javascript'> document.location ='index.php'; </script>";    }    else      {        echo "<script>alert('Something Went Wrong. Please try again');</script>";      }  }  ?> |

**Step 8 – Code for data deletion from the database**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12 | <?php  //database connection  file  include('dbconnection.php');  //Code for deletion  if(isset($\_GET['delid']))  {  $rid=intval($\_GET['delid']);  $sql=mysqli\_query($con,"delete from tblusers where ID=$rid");  echo "<script>alert('Data deleted');</script>";  echo "<script>window.location.href = 'index.php'</script>";  }  ?> |

[View Demo](https://phpgurukul.com/demos/php-mysqli-crud/)

**Download Source Code (CRUD operation using PHP and MySQLi)**

Size: 9.31 KB

Version: V 1

[**Download Now!**](https://phpgurukul.com/?smd_process_download=1&download_id=12801)