

ΕΡΓΑΣΤΗΡΙΑΚΗ ΑΣΚΗΣΗ 3

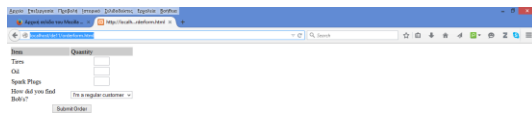
Η γλώσσα PHP

1. Φάση 1- PHP και HTML

Βήμα 1

1. Ενεργοποιήστε από το xampp control panel τον apache και τον mysql server.
2. Δημιουργήστε στο φάκελο C:/xampp/htdocs ένα νέο φάκελο για την ομάδα σας ακολουθώντας το πρότυπο: Pa9 (για το τμήμα της Παρασκευής 9.00πμ). Στο φάκελο αυτό αντιγράψτε τα αρχεία orderform.html, processorder.php, freight.html και freight.php.
3. Ανοίξτε ένα φυλλομετρητή και οδηγήστε τον στο localhost/pa9/orderform.html

Πρέπει να βλέπετε αυτό που ακολουθεί.

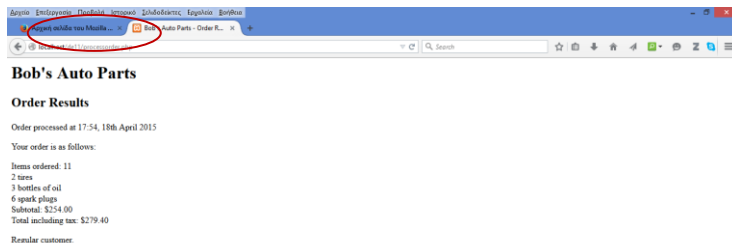


4. Ανοίξτε το αρχείο orderform.html με οποιονδήποτε κειμενογράφο. Είναι αρχείο html ή PHP? Βλέπετε κανένα tag της html που δεν γνωρίζετε?

(υπόδειξη: τι γνωρίζεται για το «action»? παρατηρήστε τα ονόματα των πεδίων της φόρμας)

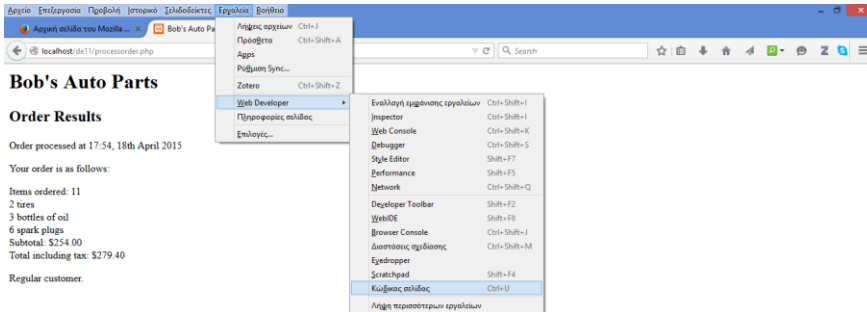
Βήμα 2

5. Συμπληρώστε τη φόρμα και πατήστε submit order. Θα δείτε κάτι σαν το παρακάτω:



6. Παρατηρήστε ότι το αποτέλεσμα που βλέπετε στο φυλλομετρητή σας έχει παραχθεί από το αρχείο processorder.php. Από το φυλλομετρητή σας, προσπαθήστε να δείτε τον κώδικα της σελίδας. Αν

χρησιμοποιείτε Mozilla Firefox, τότε για να δείτε τον κώδικα επιλέγετε όπως φαίνεται παρακάτω εργαλεία/web developer/κώδικας σελίδας.



Ο κώδικας αυτός ακολουθεί:

```
<html>
```

```
<head>
```

```
<title>Bob's Auto Parts - Order Results</title>
```

```
</head>
```

```
<body>
```

```
<h1>Bob's Auto Parts</h1>
```

```
<h2>Order Results</h2>
```

```
<p>Order processed at 17:54, 18th April 2015</p><p>Your order is as follows: </p>Items ordered: 11<br />2 tires<br />3 bottles of oil<br />6 spark plugs<br />Subtotal: $254.00<br />Total including tax: $279.40<br /><p>Regular customer.</p></body>
```

```
</html>
```

Όπως βλέπετε, δεν περιέχει κανένα στοιχείο PHP.

7. Ανοίξτε το αρχείο processorder.php με οποιονδήποτε κειμενογράφο. Τι παρατηρείτε?

Ο κώδικας PHP εκτελείται στην πλευρά του server και αυτό που αποστέλλεται στον φυλλομετρητή είναι καθαρή html σελίδα.

Βήμα 3^ο

8. Ποιο είναι το ποσοστό του φόρου που υπολογίζεται στο συγκεκριμένο παράδειγμα?
9. Αλλάξτε το σε 23% (ισχύον ΦΠΑ στην Ελλάδα).
10. Αλλάξτε την τιμή σε ένα εκ των ειδών και επιβεβαιώστε την ορθή εκτέλεση των υπολογισμών.
11. Προσθέστε ένα ακόμα είδος, τόσο στη φόρμα όσο και στη σελίδα PHP.
12. Δημιουργήστε ένα νέο αρχείο PHP το οποίο τυπώνει στην οθόνη «Hello World!», την ώρα και «Goodmorning TEI» αν είναι πρωί ή «Goodevening TEI» αν είναι απόγευμα.
13. Εισάγετε μέσα στο αρχείο που φτιάξατε την εντολή rhrinfo(); Ποιο είναι το αποτέλεσμα?

-
14. Δημιουργήστε μια φόρμα σε html και ένα αρχείο PHP σχετικό με την εφαρμογή που σας έχει ανατεθεί να αναπτύξετε.

2. Φάση 2 Προσπέλαση βάσης δεδομένων MySQL από το διαδίκτυο με γλώσσα Ph

2.1. ΘΕΩΡΗΤΙΚΟ ΥΠΟΒΑΘΡΟ

Έστω ότι υπάρχει ένα ηλεκτρονικό βιβλιοπωλείο (BookOrama) και ότι για τις ανάγκες αναζήτησης βιβλίων και αποστολής παραγγελιών έχει οργανωθεί η βάση δεδομένων books. Το παράδειγμα στο οποίο στηριζόμαστε στην άσκηση αυτή αντλείται από το βιβλίο PHP and MySQL development, L. Welling, L. Thomson, Addison Wesley ed.

Τα βήματα που ακολουθούνται όταν ένας χρήστης χρειάζεται να προσπελάσει τη βάση δεδομένων φαίνονται παρακάτω.

1. A user's web browser issues an HTTP request for a particular web page. This happens through an html form. For example, the user might have requested a search for all the books written by Michael Morgan at Book-O-Rama, using an HTML form. The search results page is called results.php.
2. The web server receives the request for results.php, retrieves the file, and passes it to the PHP engine for processing.
3. The PHP engine begins parsing the script. Inside the script is a command to connect to the database and execute a query (perform the search for books). PHP opens a connection to the MySQL server and sends on the appropriate query.
4. The MySQL server receives the database query, processes it, and sends the results—a list of books—back to the PHP engine.
5. The PHP engine finishes running the script. This usually involves formatting the query results nicely in HTML. It then returns the resulting HTML to the web server.
6. The web server passes the HTML back to the browser, where the user can see the list of books she requested.

Στα πλαίσια του πειραματικού μέρους θα δούμε πως εκτελούνται τα βήματα αυτά.

2.2. ΠΕΙΡΑΜΑΤΙΚΟ ΜΕΡΟΣ

Βήμα 1: Δημιουργήστε τη βάση δεδομένων

1. Δημιουργήστε τη βάση δεδομένων books η οποία περιλαμβάνει του πίνακες customers, orders, books, order_items, book_reviews.

Για να το κάνετε αυτό,

- ενεργοποιήστε από το xampp κατά τα γνωστά τους Apache και MySQL servers



Πανεπιστήμιο Δυτικής Αττικής

Τμ. Μηχανικών Βιομηχανικής Σχεδίασης και Παραγωγής

Μάθημα: Σχεδίαση Διαδικτυακών Εφαρμογών

- ο με χρήση του PHPMyAdmin, δημιουργήστε τη βάση δεδομένων
- ο χρησιμοποιώντας τον κώδικα SQL που φαίνεται παρακάτω δημιουργήστε τους παραπάνω πίνακες

```
create table customers
(
  customerid int unsigned not null auto_increment primary key,
  name char(50) not null,
  address char(100) not null,
  city char(30) not null
);

create table orders
(
  orderid int unsigned not null auto_increment primary key,
  customerid int unsigned not null,
  amount float(6,2),
  date date not null
);

create table books
(
  isbn char(13) not null primary key,
  author char(50),
  title char(100),
  price float(4,2)
);

create table order_items
(
  orderid int unsigned not null,
  isbn char(13) not null,
  quantity tinyint unsigned,

  primary key (orderid, isbn)
);

create table book_reviews
(
  isbn char(13) not null primary key,
  review text
);
```

- ο Εισάγετε δεδομένα στους παραπάνω πίνακες χρησιμοποιώντας τον παρακάτω κώδικα SQL

```
use books;

insert into customers values
(3, "Julie Smith", "25 Oak Street", "Airport West");
```



Πανεπιστήμιο Δυτικής Αττικής

Τμ. Μηχανικών Βιομηχανικής Σχεδίασης και Παραγωγής

Μάθημα: Σχεδίαση Διαδικτυακών Εφαρμογών

```
(4, "Alan Wong", "1/47 Haines Avenue", "Box Hill"),  
(5, "Michelle Arthur", "357 North Road", "Yarraville");
```

insert into orders values

```
(NULL, 3, 69.98, "2007-04-02"),  
(NULL, 1, 49.99, "2007-04-15"),  
(NULL, 2, 74.98, "2007-04-19"),  
(NULL, 3, 24.99, "2007-05-01");
```

insert into books values

```
("0-672-31697-8", "Michael Morgan", "Java 2 for Professional Developers", 34.99),  
("0-672-31745-1", "Thomas Down", "Installing Debian GNU/Linux", 24.99),  
("0-672-31509-2", "Pruitt, et al.", "Teach Yourself GIMP in 24 Hours", 24.99),  
("0-672-31769-9", "Thomas Schenk", "Caldera OpenLinux System Administration Unleashed", 49.99);
```

insert into order_items values

```
(1, "0-672-31697-8", 2),  
(2, "0-672-31769-9", 1),  
(3, "0-672-31769-9", 1),  
(3, "0-672-31509-2", 1),  
(4, "0-672-31745-1", 3);
```

insert into book_reviews values

```
("0-672-31697-8", "Morgan's book is clearly written and goes well beyond most of the basic Java books out there.");
```

2. Δημιουργήστε ένα νέο χρήστη με πλήρη δικαιώματα στην παραπάνω βάση. Ο χρήστης θα έχει όνομα bookorama και κωδικό (password) bookorama123.

Ερώτημα:

1. Ο πίνακας books από ποια πεδία αποτελείται?
2. Πόσες εγγραφές περιέχει ο πίνακας books?

Βήμα 2: Δημιουργήστε τη φόρμα στην οποία ο χρήστης θα κάνει την αναζήτηση βιβλίων

Δημιουργήστε το αρχείο search.html το οποίο περιέχει τον παρακάτω κώδικα html και αποθηκεύστε το στο C:/xampp/htdocs/ tmima/Bookorama (όπου τμήμα Pa9, Pa13, Pa14 ανάλογα με τη μέρα και ώρα του εργαστηριακού σας τμήματος).

```
<html>  
<head>  
<title>Book-O-Rama Catalog Search</title>  
</head>  
<body>  
<h1>Book-O-Rama Catalog Search</h1>  
<form action="results.php" method="post">  
Choose Search Type:<br />  
<select name="searchtype">
```

```
<option value="author">Author</option>
<option value="title">Title</option>
<option value="isbn">ISBN</option>
</select>
<br />
Enter Search Term:<br />
<input name="searchterm" type="text" size="40"/>
<br />
<input type="submit" name="submit" value="Search"/>
</form>
</body>
</html>
```

Ερώτημα: με βάση τον παραπάνω κώδικα, ποιο script PHP θα επεξεργαστεί τα δεδομένα που θα εισάγει ο χρήστης?

Βήμα 3: Ανάπτυξη του αρχείου PHP

In any script used to access a database from the Web, **you follow some basic steps:**

1. Check and filter data coming from the user.
2. Set up a connection to the appropriate database.
3. Query the database.
4. Retrieve the results.
5. Present the results back to the user.

Το αρχείο PHP που κάνει την επιθυμητή επεξεργασία δίνεται παρακάτω:

```
<html>
<head>
<title>Book-O-Rama Search Results</title>
</head>
<body>
<h1>Book-O-Rama Search Results</h1>
<?php
// create short variable names
$searchtype=$_POST['searchtype'];
$searchterm=trim($_POST['searchterm']);
if (!$searchtype || !$searchterm) {
echo 'You have not entered search details. Please go back and try again.';
exit;
}
```

```
if (!get_magic_quotes_gpc()){
$searchtype = addslashes($searchtype);
$searchterm = addslashes($searchterm);
}
@ $db = new mysqli('localhost', 'bookorama', 'bookorama123', 'books');
if (mysqli_connect_errno()) {
echo 'Error: Could not connect to database. Please try again later.';
exit;
}
$query = "select * from books where ".$searchtype." like '%".$searchterm.%'";
$result = $db->query($query);
$num_results = $result->num_rows;
echo "<p>Number of books found: ".$num_results."</p>";
for ($i=0; $i <$num_results; $i++) {
$row = $result->fetch_assoc();
echo "<p><strong>".($i+1)." Title: ";
echo htmlspecialchars(stripslashes($row['title']));
echo "</strong><br />Author: ";
echo stripslashes($row['author']);
echo "<br />ISBN: ";
echo stripslashes($row['isbn']);
echo "<br />Price: ";
echo stripslashes($row['price']);
echo "</p>";
}
$result->free();
$db->close();
?>
</body>
</html>
```

Ονομάστε το results.php και αποθηκεύστε το στον ίδιο φάκελο με το search.html.

Τρέξτε το search.html στο φυλλομετρητή σας πληκτρολογώντας τη διεύθυνση:

localhost/tmima/bookorama/search.html (όπου τμήμα, το τμήμα σας όπως έχει διατυπωθεί παραπάνω).

Ανάλυση του κώδικα

1. Checking and Filtering Input Data

You begin the script by stripping any whitespace that the user might have inadvertently entered at the beginning or end of his search term. You do this by applying the function trim() to the value of \$_POST['searchterm'] when giving it a shorter name:

```
$searchterm=trim($_POST['searchterm']);
```

The next step is to verify that the user has entered a search term and selected a search type. Note that you check whether the user entered a search term after trimming whitespace from the ends of `$searchterm`. If you arrange these lines in the opposite order, you could encounter situations in which a user's search term is not empty and therefore does not create an error message; instead, it is all whitespace, so it is deleted by `trim()`:

```
if (!$searchtype || !$searchterm) {  
    echo "You have not entered search details. Please go back and try again."  
    exit;  
}
```

You check the `$searchtype` variable, even though in this case it's coming from an HTML SELECT. You might ask why you should bother checking data that has to be filled in. It's important to remember that there might be more than one interface to your database. Also, it's sensible to screen data in case of any security problems that can arise because of users coming from different points of entry.

When you plan to use any data input by a user, you need to filter it appropriately for any control characters. We described the functions `addslashes()`, `stripslashes()`, and `get_magic_quotes_gpc()`. You need to escape data when submitting any user input to a database such as MySQL.

In this case, you check the value of the `get_magic_quotes_gpc()` function. It tells you whether quoting is being done automatically. If it is not, you use `addslashes()` to escape the data:

```
if (!get_magic_quotes_gpc()) {  
    $searchtype = addslashes($searchtype);  
    $searchterm = addslashes($searchterm);  
}
```

2. Setting Up a Connection

The PHP library for connecting to MySQL is called `mysqli` (the *i* stands for improved). You use the following line in the script to connect to the MySQL server:

```
@ $db = new mysqli('localhost', 'bookorama', 'bookorama123', 'books');
```

This line instantiates the `mysqli` class and creates a connection to host `localhost` with username `bookorama`, and password `bookorama123`. The connection is set up to use the database called `books`.

The result of your attempt at connection is worth checking because none of the rest of code will work without a valid database connection. You do this using the following code:

```
if (mysqli_connect_errno()) {  
    echo 'Error: Could not connect to database. Please try again later.'  
    exit;  
}
```

Bear in mind that there is a limit to the number of MySQL connections that can exist at the same time. The MySQL parameter `max_connections` determines what this limit is. The purpose of this parameter and the related Apache parameter `MaxClients` is to tell the server to reject new connection requests instead of allowing machine resources to be completely used up at busy times or when software has crashed.

3. Choosing a Database to Use

Remember that when you are using MySQL from a command-line interface, you need to tell it which database you plan to use with a command such as `use books`; You also need to do this when connecting from the Web. The database to use is specified as a parameter to the `mysqli` constructor or the `mysqli_connect()` function. If you want to change the default database, you can do so with the `mysqli_select_db()` function.

It can be accessed as either

```
$db->select_db(dbname)
```

or as

```
mysqli_select_db(db_resource, db_name)
```

4. Querying the Database

To actually perform the query, you can use the `mysqli_query()` function. Before doing this, however, it's a good idea to set up the query you want to run:

```
$query = "select * from books where ".$searchtype." like '%" . $searchterm . "%'";
```

In this case, you search for the user-input value (`$searchterm`) in the field the user specified (`$searchtype`). Notice the use of `like` for matching rather than `equal`: it's usually a good idea to be more tolerant in a database search.

Remember that the query you send to MySQL does not need a semicolon at the end of it, unlike a query you type into the MySQL monitor.

You can now run the query:

```
$result = $db->query($query);
```

Or, if you want to use the procedural interface, you use

```
$result = mysqli_query($db, $query);
```

5. Retrieving the Query Results

A large variety of functions is available to break the results out of the result object or identifier in different ways. The result object or identifier is the key to accessing the rows returned by the query. In this example, you counted the number of rows returned and also used the `mysqli_fetch_assoc()` function.

When you use the object-oriented approach, the number of rows returned is stored in the `num_rows` member of the result object, and you can access it as follows:

```
$num_results = $result->num_rows;
```

It's useful to know this if you plan to process or display the results, because you now know how many there are and can loop through them:

```
for ($i=0; $i <$num_results; $i++) {
```

```
// process results
```

```
}
```

In each iteration of this loop, you call `$result->fetch_assoc()` (or `mysqli_fetch_assoc()`). The loop does not execute if no rows are returned. This is a function that takes each row from the resultset and returns the row as an array, with each key an attribute name and each value the corresponding value in the array:

```
$row = $result->fetch_assoc();
```

Given the array `$row`, you can go through each field and display it appropriately, as shown in this example:

```
echo "<br />ISBN: ";
```

```
echo stripslashes($row['isbn']);
```

Βήμα 4: Τρέξτε το παραπάνω παράδειγμα

Ανοίξτε στο φυλλομετρητή σας το `search.html` από το κατάλληλο μονοπάτι (για το τμήμα Παρασκευής 9 το μονοπάτι είναι: `localhost/ra9/BookOrama/search.html`).

Επιλέξτε στο πεδίο `search type` την τιμή `author` και στο πεδίο `search term` εισάγετε `Michael Morgan`. Πατώντας `search`, θα πρέπει να εμφανιστεί το παρακάτω:



Book-O-Rama Search Results

Number of books found: 1

- Title:** Java 2 for Professional Developers
Author: Michael Morgan
ISBN: 0-672-31697-8
Price: 34.99

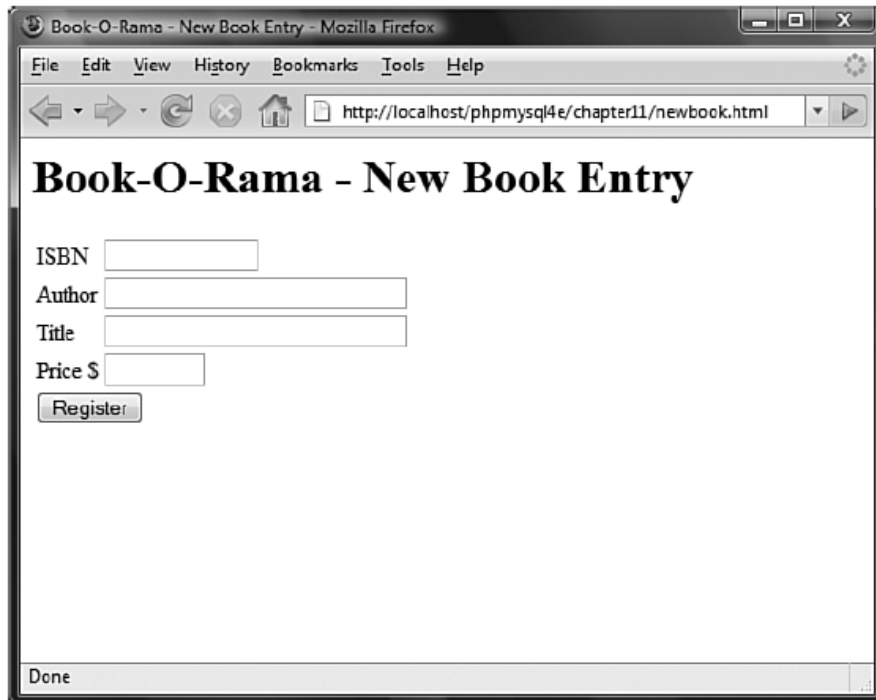
Εργασία

Κάντε τις κατάλληλες αλλαγές ώστε ο χρήστης να μπορεί να επιλέγει ποσότητα από το βιβλίο στην 1^η φόρμα και στην απάντηση εκτός από τα στοιχεία του βιβλίου να εμφανίζεται και το κόστος της παραγγελίας.

Πόσες και ποιες επεμβάσεις πρέπει να κάνετε?

3. Φάση 3- Εισαγωγή στοιχείων σε βάση δεδομένων MySQL από το διαδίκτυο με γλώσσα PHP

Έστω ότι θέλουμε να είναι δυνατή η προσθήκη ενός βιβλίου στη βάση δεδομένων που δημιουργήσαμε στην εργαστηριακή άσκηση 6 και ότι η σχετική φόρμα θέλουμε να δείχνει όπως φαίνεται παρακάτω.



Βήμα 1: Δημιουργούμε το σχετικό αρχείο html το οποίο ονομάζομε newbook.html και το αποθηκεύουμε στον κατάλληλο φάκελο (για το τμήμα μας) μέσα στο C://xampp/htdocs

Το αρχείο newbook.html δίνεται παρακάτω:

```
<html>
<head>
<title>Book-O-Rama - New Book Entry</title>
</head>
<body>
<h1>Book-O-Rama - New Book Entry</h1>
<form action="insert_book.php" method="post">
<table border="0">
<tr>
<td>ISBN</td>
<td><input type="text" name="isbn" maxlength="13" size="13"></td>
</tr>
<tr>
<td>Author</td>
<td><input type="text" name="author" maxlength="30" size="30"></td>
</tr>
<tr>
<td>Title</td>
<td><input type="text" name="title" maxlength="60" size="30"></td>
</tr>
<tr>
```



```
<td>Price $</td>
<td><input type="text" name="price" maxlength="7" size="7"></td>
</tr>
<tr>
<td colspan="2"><input type="submit" value="Register"></td>
</tr>
</table>
</form>
</body>
</html>
```

Βήμα 2: Δημιουργούμε το αρχείο pph το οποίο θα εισάγει τα δεδομένα του καινούριου βιβλίου στη βάση μας, το ονομάζουμε insert_book.pph και το αποθηκεύουμε στον κατάλληλο φάκελο (για το τμήμα μας) μέσα στο C://xampp/htdocs.

```
<html>
<head>
<title>Book-O-Rama Book Entry Results</title>
</head>
<body>
<h1>Book-O-Rama Book Entry Results</h1>
<?php
// create short variable names
$isbn=$_POST['isbn'];
$author=$_POST['author'];
$title=$_POST['title'];
$price=$_POST['price'];
if (!$isbn || !$author || !$title || !$price) {
echo "You have not entered all the required details.<br
/>"
."Please go back and try again.";
exit;
}
if (!get_magic_quotes_gpc()) {
$isbn = addslashes($isbn);
$author = addslashes($author);
$title = addslashes($title);
$price = doubleval($price);
}
@ $db = new mysqli('localhost', 'bookorama',
'bookorama123', 'books');
if (mysqli_connect_errno()) {
echo "Error: Could not connect to database. Please try
again later.";
exit;
```

```
}  
$query = "insert into books values  
( '$isbn.', '$author.', '$title.',  
'$price.' )";  
$result = $db->query($query);  
if ($result) {  
echo $db->affected_rows." book inserted into  
database.";  
} else {  
echo "An error has occurred. The item was not added.";  
}  
$db->close();  
?>  
</body>  
</html>
```

Βήμα 3: τρέξτε ένα παράδειγμα εισαγωγής βιβλίου

1. Ανοίξτε σε ένα browser το newbook.html (πληκτρολογώντας π.χ. localhost/Pa9/bookorama/newbook.html).
2. Εισάγετε στοιχεία για ένα βιβλίο.
3. Ψάξτε στη βάση για να δείτε αν έχει καταχωρηθεί το βιβλίο χρησιμοποιώντας το search.html.

Βήμα 4: Εξάσκηση

1. Παρατηρήστε τι κοινά έχουν το results.php και το insert_book.php
2. Τα αρχεία results.php και insert_book.php κάνουν και τα δύο query στη βάση δεδομένων. Το μεν results.php κάνει query τύπου select και το δε insert_book.php κάνει query τύπου insert. Δημιουργήστε ένα παράδειγμα με query τύπου update.
3. Δημιουργήστε συνδέσμους από την σελίδα search.html στην σελίδα new_book.html και από τη σελίδα των αποτελεσμάτων εισαγωγής νέων βιβλίων στη σελίδα search.html.