

Sign up for our free weekly [Web Dev Newsletter](#).

✕



Search for articles, questions, tips


🔍

- articles
- quick answers
- discussions
- features
- community
- help

Articles » Web Development » PHP » General

🔖 🖨

Step-by-Step PHP Tutorials for Beginners - Creating your PHP program FROM SCRATCH: Basic Authentication, Membership and CRUD functionalities



Kristian Guevara, 23 Aug 2016

★★★★★

4.56 (78 votes)

Rate this:

★★★★★

PHP Tutorials for beginners

Introduction

Interested in creating your own website? Ever wondered how cool is it to have a website that has a log-in/log-out functionality? Want to learn how to Create, Read, Update or Delete(CRUD) records in a database? Have you lost track on your previous tutorials? Well, I'm going to teach you how to create one *from scratch* where in you will know every single details on how the code works. If you are new to back-end web development, this tutorial is for you. **I'll explain everything in detail** so that you won't have to research some particular methods being used. We won't be using any frameworks to keep things simple. Also I won't be focusing on the websites design since were after the functionalities though, it's easy to implement the design. What will be doing is a simple *item list* when the users is logged in.

Shameless plug

You can also checkout my *ASP.NET MVC 5 from scratch* version right [here](#).

I also have my *NodeJS/Express with MongoDB* version right [here](#).

Pre-requisites

In this tutorial, we will be using some languages that will enhance our development. *If you really don't know any of the things I will mention, it's ok. Just try to go with the flow.* It's just easy to comprehend them since I'm not into using advanced methods for this one. Off course since this is creating a back-end service for a website, you must know at least the very very basics of the following:

1. ~~HTML/CSS~~ — Off course this will never get off the list when it comes to web development. Why did I even bother place this? LOL.

2. **JavaScript** - For a redirecting script. No jQuery needed for now

3. **mySQL** - the most basic language for querying. We will be using this for our CRUD(Create, Read, Update, Delete) functionalities.

For the software, you will be needing the following:

1. **Text Editor** - [Sublime Text](#)/[Notepad++](#) will do. This will serve as our coding environment.

2. **XAMPP** - This will be our web server. This is were we will be saving our website files and save our data to the database.

3. **Web Browser** - You can use any browser for as long as it doesn't eat much of your RAM.

Make sure to have those 3 installed. Do not proceed if your not done installing them. I already provided a hyperlink to make things faster.

For our agenda in this topic:

1. **Utilize CRUD statements in SQL** - Create, read, update and delete records in a database.

2. **Simple authentication and security** - Logging in and out, handling unauthorized users and restricting access.

3. **Simple time and date manipulation** - Displaying time and date postings and monitoring your posts.

4. **Displaying public and private data** - Simply display information for logged-in users only and for the public

5. **De-briefing and Summary** - Final output and some notes. FAQ's are also place here.

Topic Contents:

Note: In case you stopped on a certain part or want to search quickly , you can simply resume by pressing ctrl + f and look for the number where you last stopped

1. **Setting up your server**
2. **Creating the public HTML Pages**
3. **Creating the database and it's tables**
4. **Adding users to the database**
5. **User log-in: Authentication**
6. **Setting up the home page for Logged-in users and Logging-out**
7. **Testing Page Security**
8. **Adding data to the list - User Access Only**
9. **Displaying data in the home page**
10. **Editing Data**
11. **Deleting data**
12. **Displaying public data**
13. **De-briefing and Summary**

Make sure you follow chronologically. Don't proceed without accomplishing the previous part not unless you know the code.

Some reminders before we start:

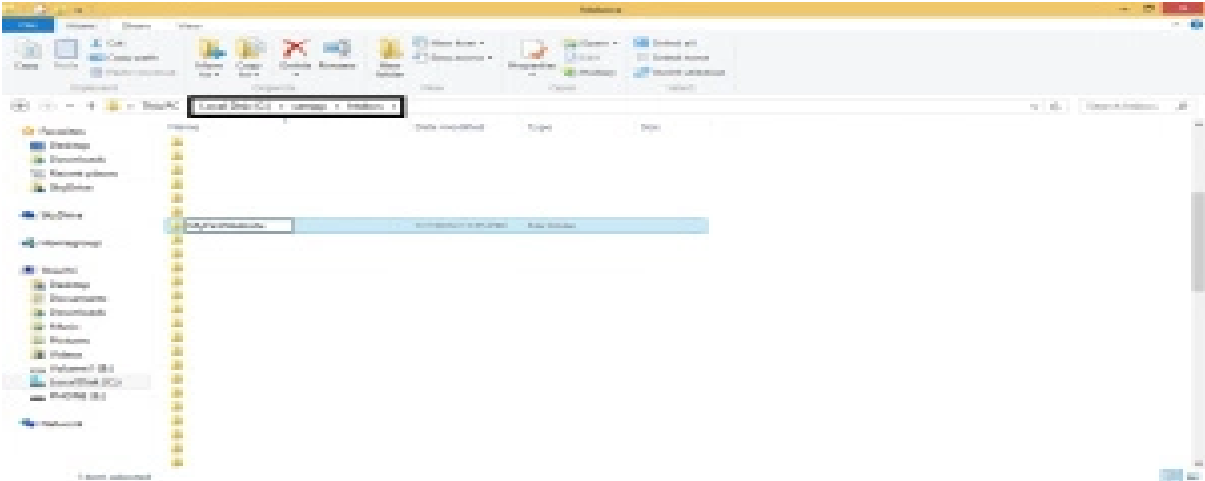
Note: If the picture seems unclear/blurry, you can click it to view it on it's larger size. Also, pardon me for using images to display the codes rather than showing you a raw text because it can't format properly on my WordPress subscription but don't worry, I'll be uploading the finished product on the end of this tutorial to double-check your work.

-----Real Tutorials Begin Here-----

1.) Setting up your server

Now that your all set and go, as the basics of programming goes, let's start by creating a simple shoutout of "hello world" in our server.

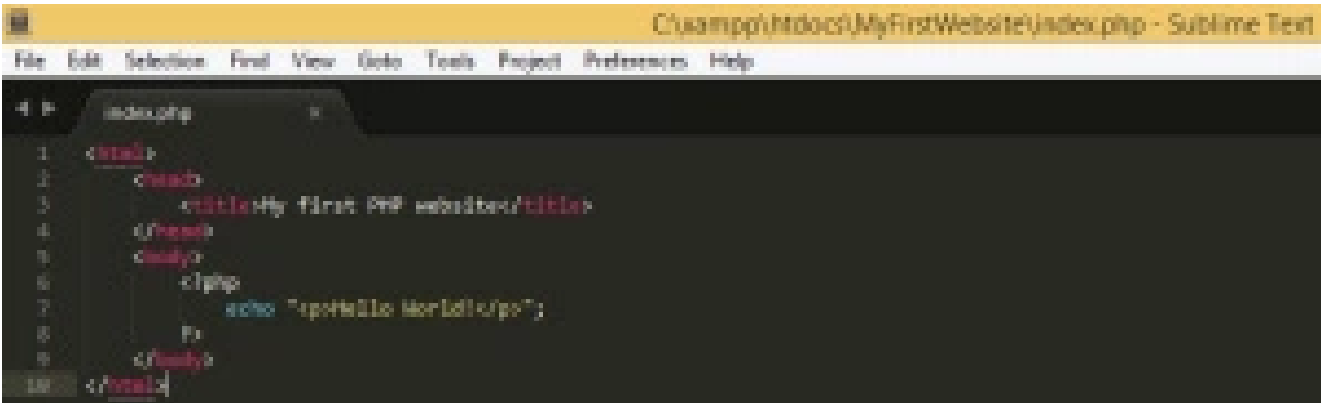
First of, go to the directory where you installed your XAMPP (Commonly in C:\xampp). From there go to the *htdocs* folder (Commonly in C:\xampp\htdocs) and create a folder named "MyFirstWebsite".



From that part, you have now created a Local URL for your website. That folder will be used to supply in all website files (.html, .php, .css, .js, etc.). Open up your text editor and let's get started!

I use sublime text as my text editor. If your using Notepad++ or any others, it's ok. It's not really a big factor but it's just a preference on which one would you like to use.

What we will do is a basic HTML page and display "hello world" from the server using a basic PHP syntax. We then type the following syntax:

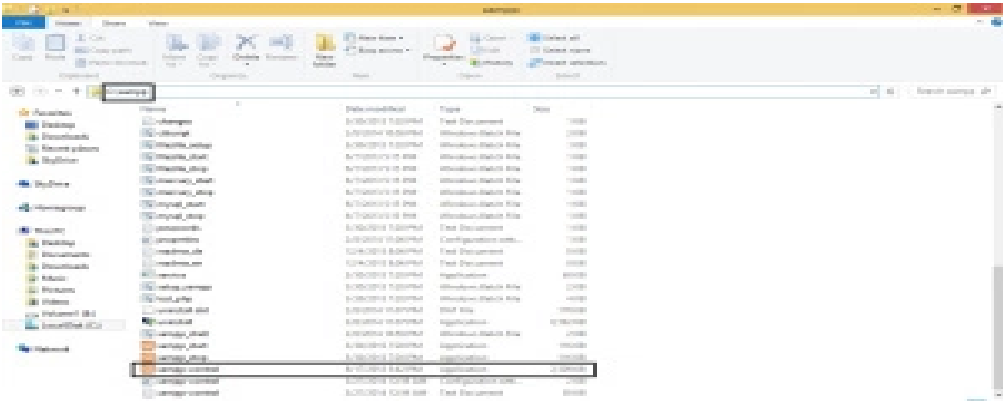


Hide Copy Code

```
<html>
<head>
<title>My first PHP Website</title>
</head>
<body>
<?php
    echo "<p>Hello World!</p>";
</body>
</html>
```

Save the file to the "MyFirstWebSite" Folder and name it as "index.php". (Directory as seen on the top part of the image)

Now that you have the file. Let's now open your XAMPP control panel. In case it doesn't appear on your desktop, it is located in your XAMPP folder as seen on the image:



Now that it's there, Run your Apache and mySQL by clicking the "Start" button on the actions column. You should see a random PID(s) and the default port number. **Apache** is the name of our web server wherein it will handle all the files as well as serve as the communication to the web browser and **MySQL** is our database which will store all of our information.

Open up your web browser and in the address bar, type **localhost**. You should see the menu of your XAMPP.

If it's the first time you run it, it will ask what language would you prefer, just simply pick and it will lead you to the menu. If you will noticed the directory is localhost/xampp, it's where the default page leads you even if you type in localhost.



If that appears, then it means that your XAMPP server is now running. Now let's try running the website you just placed. Try typing **localhost/MyFirstWebsite**. It should be the same as the picture below.



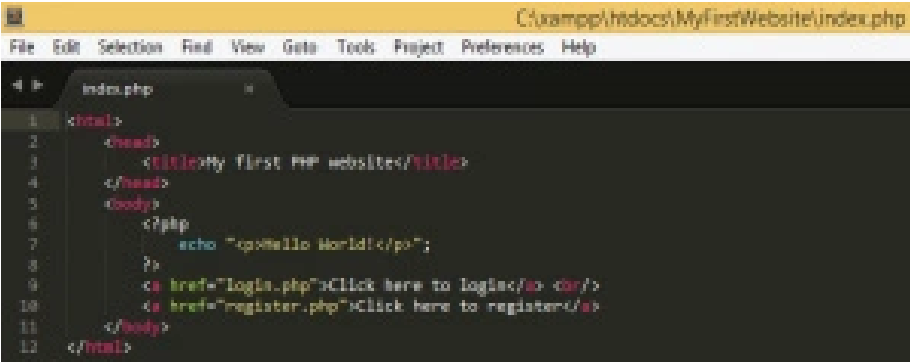
If you will notice that the URL is *MyFirstWebsite*, it is derived from the htdocs folder and it automatically reads files that are named "index"(Be it index.html, index.aspx, etc), which serve as the default page. Typing **localhost/MyfirstWebsite/index.php** is just the same. You can also create your custom name for the URL by simply renaming the folder but let's just stick to *MyFirstWebsite*.

*Note: If you don't have a file named index and you entered the URL, you will receive an error 404 for not having the file on the server. If you do have different files that are not named index.<extention>, you have to specify the specific file name. E.x: **localhost/MyfirstWebsite/page.php***

2.) Creating the public HTML Pages

Next, let's re-modify our website and add a registration link where our users can register as well as a Log-in page right after getting registered. Let's modify our home page with the following code:

index.php

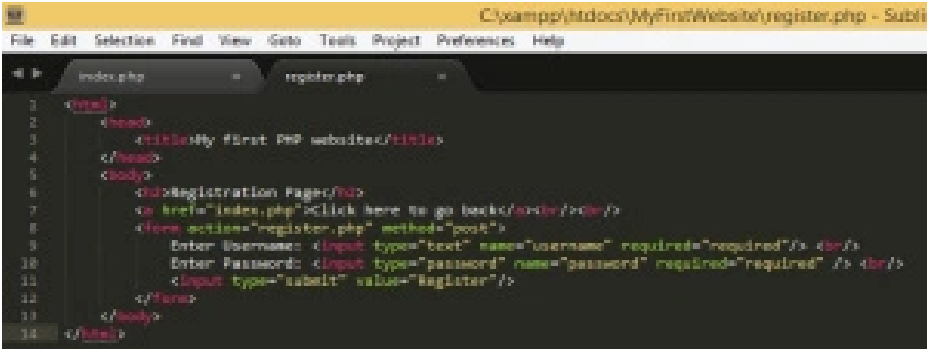


Hide Copy Code

```
<html>
  <head>
    <title>My first PHP Website</title>
  </head>
  <body>
    <?php
      echo "<p>Hello World!</p>";
    <?>
    <a href="login.php"> Click here to login
    <a href="register.php"> Click here to register
  </body>
</html>
```

As you can see, we only added 2 links which are for the Login and register. Let's create the registration page first:

register.php



Hide

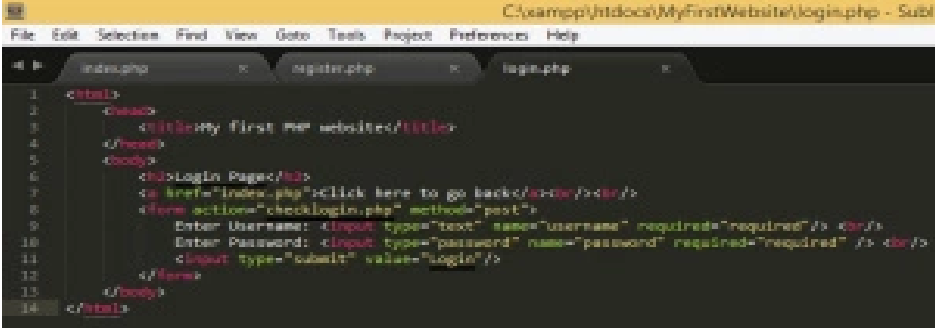
Copy Code

```
<html>
  <head>
    <title>My first PHP Website</title>
  </head>
  <body>
    <h2>Registration Page</h2>
    <a href="index.php">Click here to go back<br/><br/>
    <form action="register.php" method="POST">
      Enter Username: <input type="text" name="username" required="required" /> <br/>
      Enter password: <input type="password" name="password" required="required" /> <br/>
      <input type="submit" value="Register"/>
    </form>
  </body>
</html>
```

As you can see, it's just a basic form where the user can input his/her credentials. Now for the login page:

Hint: Just copy-paste the same code to make things faster.

login.php



Hide

Copy Code

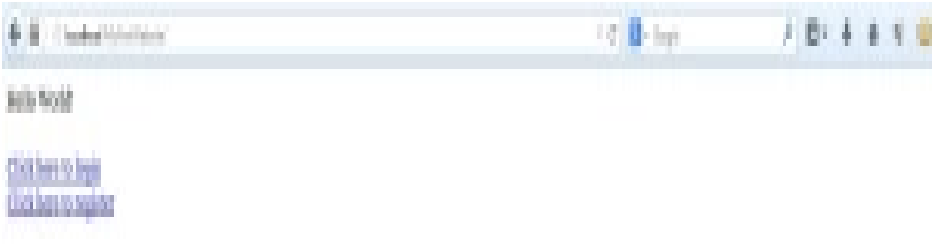
```
<html>
  <head>
    <title>My first PHP Website</title>
  </head>
  <body>
    <h2>Login Page</h2>
    <a href="index.php">Click here to go back<br/><br/>
    <form action="checklogin.php" method="POST">
      Enter Username: <input type="text" name="username" required="required" /> <br/>
      Enter password: <input type="password" name="password" required="required" /> <br/>
      <input type="submit" value="Login"/>
    </form>
  </body>
</html>
```

Basically it's just the same code as from the *register.php* but the changes made were the ones underlined.

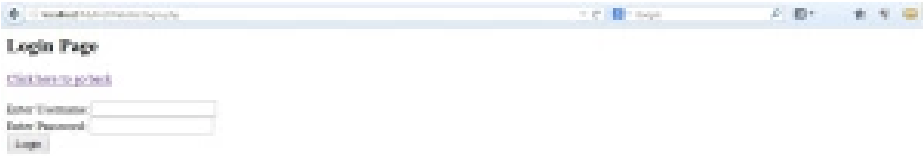
Click [here](#) for the complete login.php code

Try running *localhost/MyFirstWebsite* again and your pages should look like this:

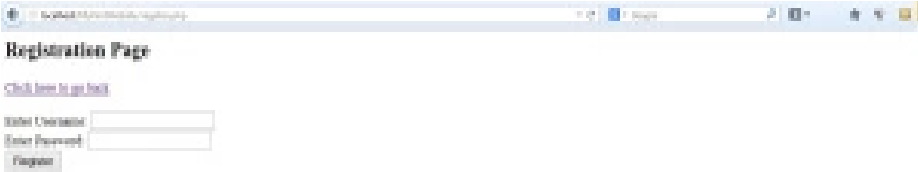
index.php



login.php



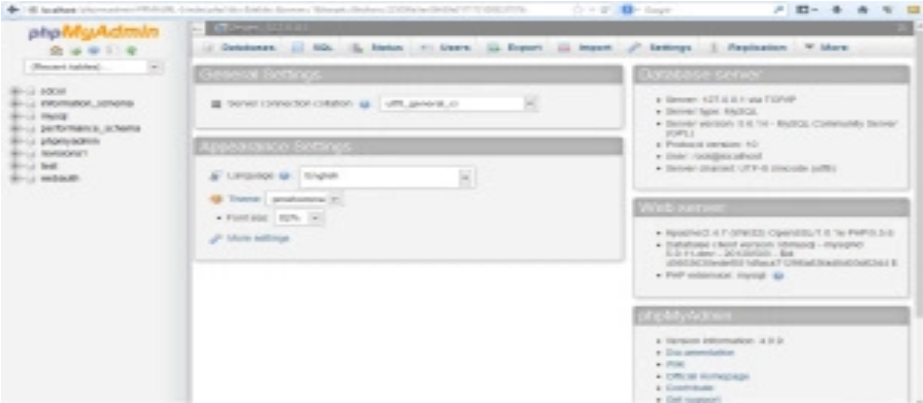
register.php



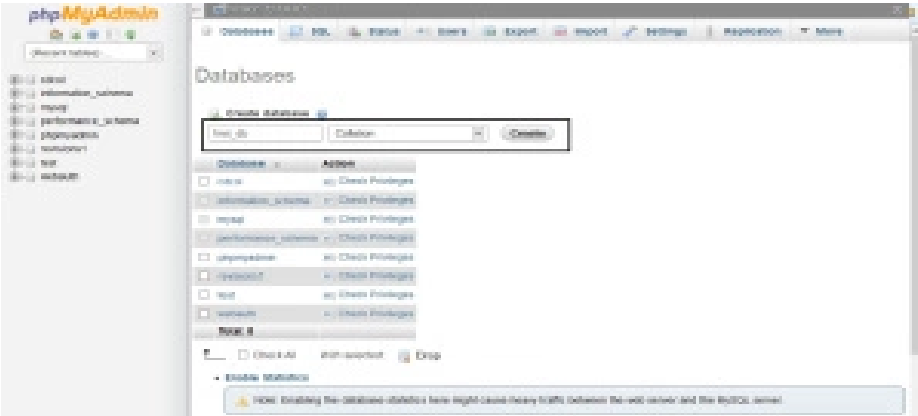
3.) Creating the database and it's tables

Now that we have our basic page for the public. Let's proceed to the database. First, type **localhost/phpmyadmin**. This will lead you to the phpmyadmin homepage:

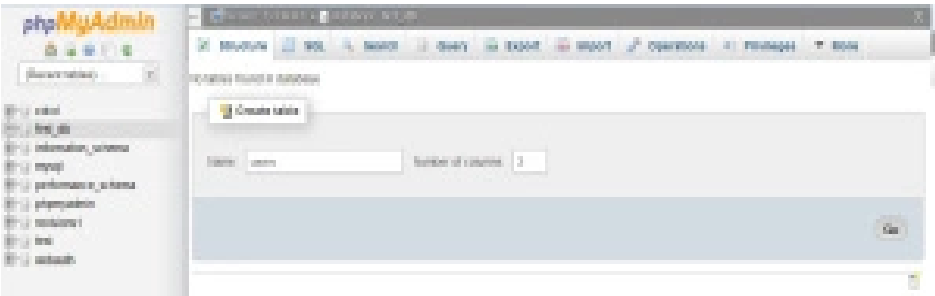
localhost/phpmyadmin



From there, go to the Databases tab located on top then from the text box in the middle, type *first_db* then click on create. Just leave the *Collation* as is.



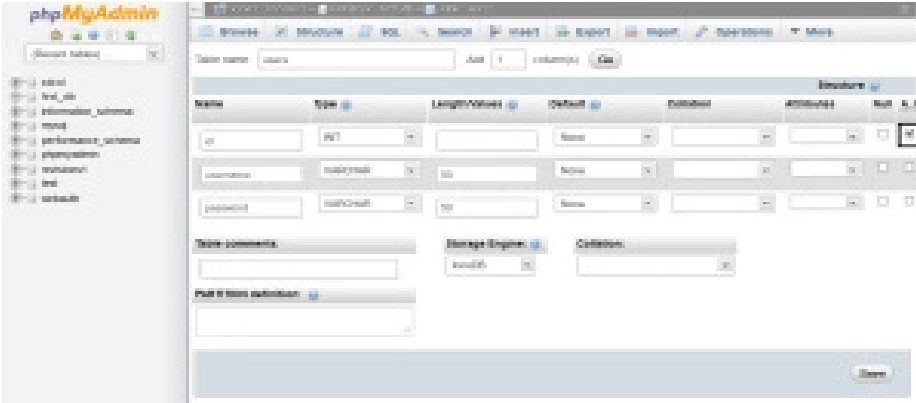
You have just successfully created your first database. Now from there, let's create a table wherein we can register our users and display information. First, click on *first_db* located on the left side and create a table named *users* with 3 columns then click on *Go*.



For the table's structure, make sure to have the following fields then click on save:

- Format: Column Name - Type - Length - Null Property - Other Properties
- id - INT - N/A - Not Null - Auto Increment
- username - varchar - 50 - Not null
- password - varchar - 50 - Not null

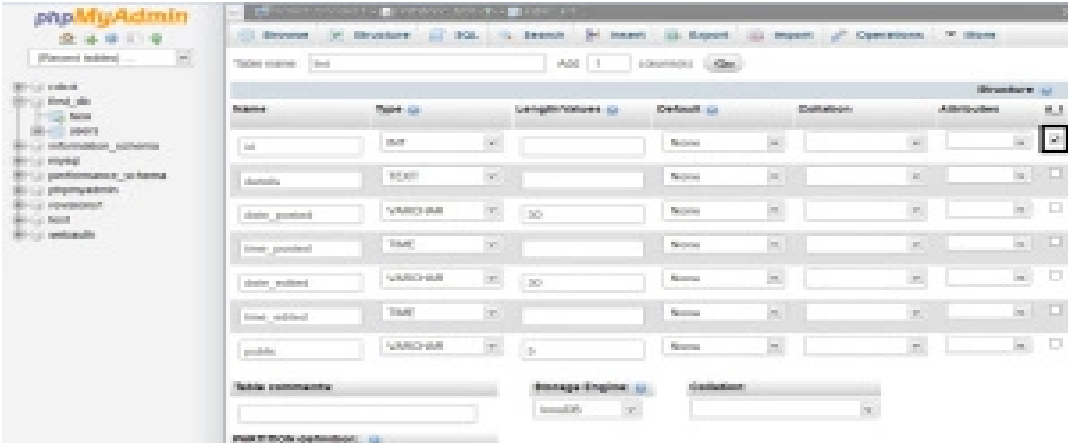
Leave everything by default if not specified.



Note: You need to scroll right for the auto_increment. I just edited the picture to fit the A_I field

Next, create another table named *list* with 7 columns and for the table's structure:

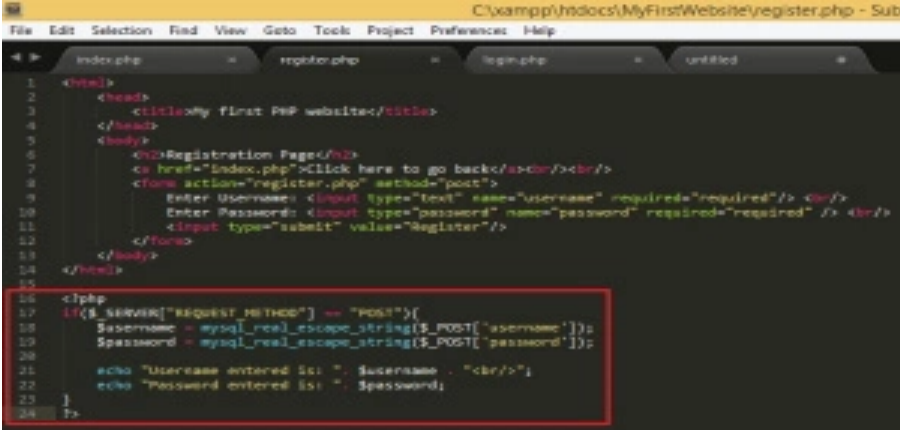
- id - INT - N/A - Not Null - Auto Increment
- details - text - Not null
- date_posted - varchar - 30 - Not null
- time_posted - Time - Not null
- date_edited - varchar - 30 - Not null
- time_edited - Time - Not null
- public -varchar - 5 - Not null



4.)Adding users to the database

Now that we have our tables. Let's move on to the fun part, getting your registration page into action. From your *registration.php*, add the following below the html codes:

register.php



Hide Copy Code

```
<html>
  <head>
    <title>My first PHP Website</title>
  </head>
  <body>
    <h2>Registration Page</h2>
    <a href="index.php">Click here to go back<br/><br/>
    <form action="checklogin.php" method="POST">
      Enter Username: <input type="text" name="username" required="required" /> <br/>
      Enter password: <input type="password" name="password" required="required" /> <br/>
      <input type="submit" value="Register"/>
    </form>
  </body>
</html>
```

Here's the explanation to the code:

- `$_SERVER["REQUEST_METHOD"] == "POST"` - checks if the form has received a POST method when the *submit* button has been clicked. The *POST* method is created in the html from the ***method="POST"***. Click [here](#) for the form method reference.

- `$_POST['']` - gets the name coming from a POST method. This just simply gets the input based on the name from the form. In our case it's username and password.
- `mysql_real_escape_string()` - encapsulates the input into a string to prevent inputs from SQL Injections. This ensures that your strings don't escape from unnecessary characters. Click [here](#) to learn more about SQL Injections.

Now try to go to your `register.php` and try to input anything then click on "Register". In my case I placed in the username `xtian` and password as `123456`. It should display the inputs below. Here's my sample:

localhost/111704/Step-by-Step-WebSite/register.php - Sublime Text

Registration Page

[Click here to go back](#)

Enter Username:

Enter Password:

Register

Username entered is: xtian

Password entered is: 123456

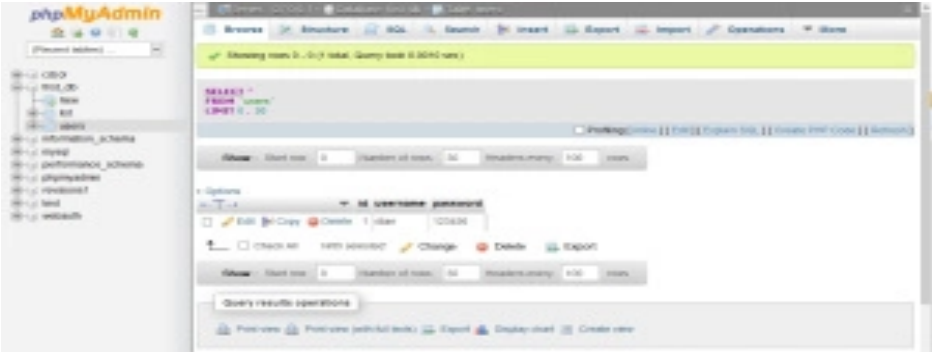
On this part, you should have understood on how to get input from the form. Now to add it to the database. On your `register.php`, add the following code:

register.php

```
1 <?php
2 //
3 //
4 <!--Registration Page-->
5
6 <a href="index.php">Click here to go back</a><br/>
7
8 <form action="register.php" method="post">
9     Enter Username: <input type="text" name="username" required="required" /> <br/>
10    Enter Password: <input type="password" name="password" required="required" /> <br/>
11    <input type="submit" value="Register"/>
12 </form>
13
14 <?php
15
16
17 <?php
18
19 <?php
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2550
2551
255
```


- **mysql_fetch_array('query')** - fetches all queries in the table to display or manipulate data. It is placed in a while loop so that it would query all rows. *Take note that only 1 row is queried per loop that's why a while loop is necessary.*
- **\$row['row name']** - the value of the column in the current query. It is represented as an array. In our case \$row is the name of the variable for our row in the loop.

Try the inputs that you have made earlier and see what happens. It should prompt that you have successfully registered. Try going to *phpmyadmin* and see your *users table*:



Congratulations! Now you know how to add data into the database with data validations.

5.)User log-in: Authentication

Now for the login page. Let's create a new file called *checklogin.php*. The reason is going back to our *login.php*, our form has an action called "*checklogin.php*", particularly <form **action = "checklogin.php"** method="POST">. If you will notice on the *register.php*, it's also on *register.php* because the back-end is done on the same file as well.

Let's now code the *checklogin.php* with the following syntax:

checklogin.php

```
1 <?php
2 session_start();
3 $username = mysql_real_escape_string($_POST['username']);
4 $password = mysql_real_escape_string($_POST['password']);
5
6 mysql_connect("localhost", "root", "") or die(mysql_error()); //Connect to server
7 mysql_select_db("first_db") or die ("Cannot connect to database"); //Connect to database
8 $query = mysql_query("Select * from users WHERE username='$username'"); //Query the users table IF there are existing rows equal to $username
9 $exists = mysql_num_rows($query); //Checks if username exists
10 $table_users = "";
11 $table_password = "";
12 if($exists > 0) //IF there are no returning rows or no existing username
13 {
14     while($row = mysql_fetch_assoc($query)) //display all rows from query
15     {
16         $table_users = $row['username']; // the first username row is passed on to $table_users, and so on until the query is finished
17         $table_password = $row['password']; // the first password row is passed on to $table_password, and so on until the query is finished
18     }
19     if(($username == $table_users) && ($password == $table_password)) // checks if there are any existing fields
20     {
21         //password == $table_password
22         $_SESSION['user'] = $username; //set the username in a session. This serves as a global variable
23         header("location: home.php"); // redirects the user to the authenticated home page
24     }
25     else
26     {
27         Print "Incorrect Username/Password!"; //Outputs the user
28         Print "Incorrect Username/Password!"; //Outputs the user
29     }
30 }
31 else
32 {
33     Print "Incorrect Username/Password!"; //Outputs the user
34     Print "Incorrect Username/Password!"; //Outputs the user
35 }
```

Hide Shrink ▲ Copy Code

```
<?php
session_start();
$username = mysql_real_escape_string($_POST['username']);
$password = mysql_real_escape_string($_POST['password']);
$bool = true;

mysql_connect("localhost", "root", "") or die (mysql_error()); //Connect to server
mysql_select_db("first_db") or die ("Cannot connect to database"); //Connect to database
$query = mysql_query("Select * from users WHERE username='$username'"); // Query the users table
$exists = mysql_num_rows($query); //Checks if username exists
$table_users = "";
$table_password = "";
if($exists > 0) //IF there are no returning rows or no existing username
{
    while($row = mysql_fetch_assoc($query)) // display all rows from query
    {
        $table_users = $row['username']; // the first username row is passed on to $table_users, and so
on until the query is finished
        $table_password = $row['password']; // the first password row is passed on to $table_password,
and so on until the query is finished
    }
    if(($username == $table_users) && ($password == $table_password))// checks if there are any
matching fields
    {
        if($password == $table_password)
        {
            $_SESSION['user'] = $username; //set the username in a session. This serves as a global
variable
            header("location: home.php"); // redirects the user to the authenticated home page
        }
    }
    else
```

```
{
    Print '<script>alert("Incorrect Password!");</script>'; // Prompts the user
    Print '<script>window.location.assign("login.php");</script>'; // redirects to login.php
}
else
{
    Print '<script>alert("Incorrect username!");</script>'; // Prompts the user
    Print '<script>window.location.assign("login.php");</script>'; // redirects to login.php
}
?>
```

Click [here](#) for the complete checklogin.php code

Here's the explanation of the code (Some are explained earlier so there's no need to reiterate):

- **session_start()** - Starts the session. This is usually done on authenticated pages. The reason why we used this is because this is required for the `$_SESSION[""]`.
- **mysql_num_rows()** - This returns an integer. This counts all the rows depending on the query.
- **\$_SESSION["name"]** - Serves as the session name for the entire session. This is relatively similar to public variables in object-oriented programming. We will be using this for validating whether the user is authenticated or not.

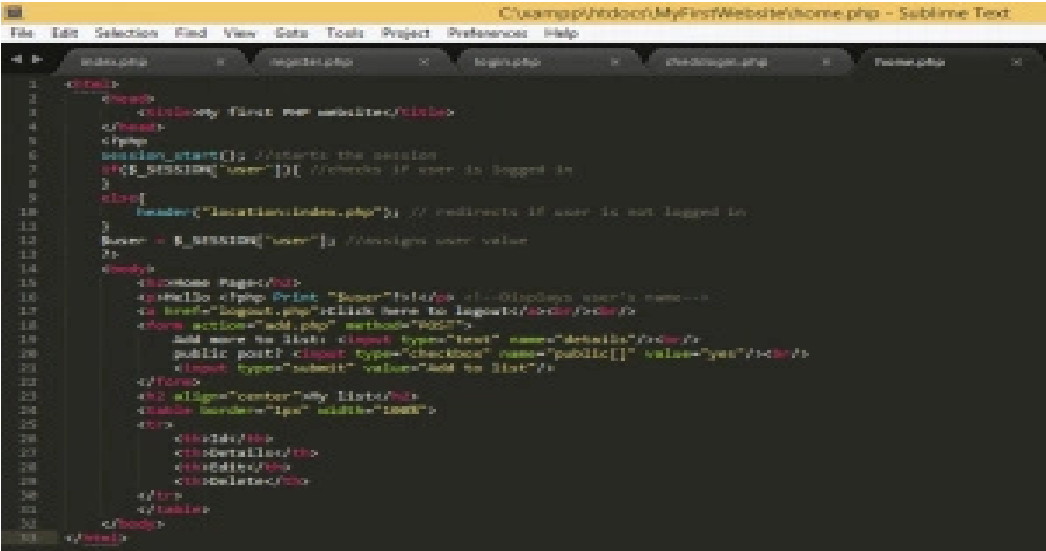
Now try to test your input with a wrong username and password. It should return the desired prompt. After testing, try inputting the correct values. It should lead you to *home.php*.

Note: home.php does not exist yet so it will produce an error 404.

6.)Setting up the home page for Logged-in users and Logging-out

Now that were authenticated, let now create our home page (*home.php*) with this following syntax:

home.php



Hide Copy Code

```
<html>
<head>
    <title>My first PHP Website</title>
</head>
<?php
session_start(); //starts the session
if($_SESSION['user']){ // checks if the user is logged in
}
else{
    header("location: index.php"); // redirects if user is not logged in
}
$user = $_SESSION['user']; //assigns user value
?>
<body>
    <h2>Home Page</h2>

<hello>!
<!--Display's user name-->
<a href="logout.php">Click here to go logout</a><br/><br/>
<form action="add.php" method="POST">
    Add more to list: <input type="text" name="details" /> <br/>
    Public post? <input type="checkbox" name="public[]" value="yes" /> <br/>
    <input type="submit" value="Add to list"/>
</form>
<h2 align="center">My list</h2>
```

Id	Details	Edit	Delete
</body> </html>			

Here's the explanation to the code:

- **session_start()** - Basically starts the session. Required for `$_SESSION[""]`.
- **header()** - redirects the user.

Try refreshing your browser and it should look like this:



Now that we have our home, let's try creating our `logout.php` and test if the user's session is off. What we would do is that if the user is logged-out, the user shouldn't access `home.php`. So here's the simple syntax to `logout.php`:

logout.php



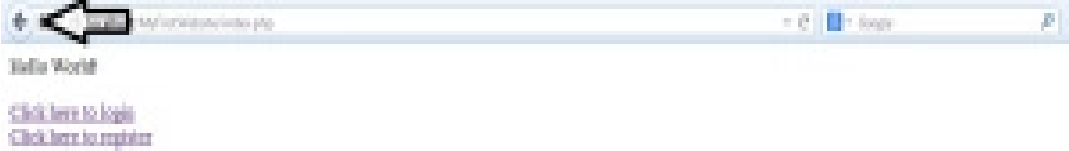
Hide Copy Code

```
<?php
    session_start();
    session_destroy();
    header("location:index.php");
?>
```

The syntax is simple. **session_destroy()** simply remove's all session's meaning, the value of `$_SESSION[""]` will be removed and **header()** will simply redirect it to the home page.

7.)Testing Page Security

Now try refreshing home.php and click on logout. Now try Clicking on the back button of your browser and see what happens:



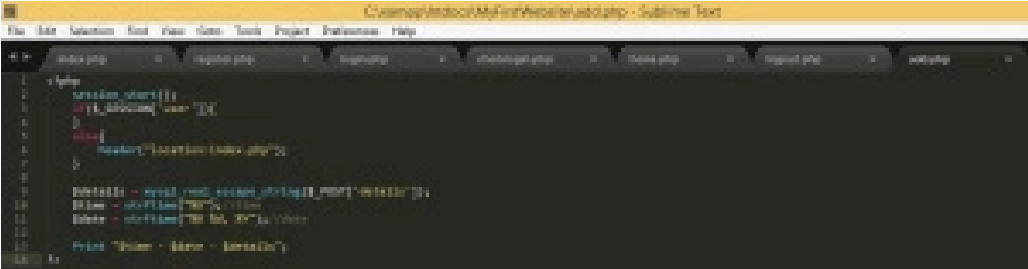
As you can see, it doesn't direct you to `home.php` because you are logged-out. Now for the second test, try manually inputting the address **localhost/MyFirstWebsite/home.php**. The same case should also happen. Since were logged-out, even a manual input of the address doesn't access an authorized page. What we have done is a simple security mechanism wherein we redirect back unauthorized users into a public page.

Now try logging in again and let's go back to `home.php`.

8.) Adding data to the list - User Access Only

In our next step, let's create the adding of item's to the list. As you will notice from the form, it is written as `<form action="add.php" method="POST">`, meaning our http post request goes to `add.php` and with that, we create our `add.php` with the following syntax:

add.php



Hide Copy Code

```
<?php
    session_start();
    if($_SESSION['user']) {
    }
    else{
        header("location:index.php");
    }
```

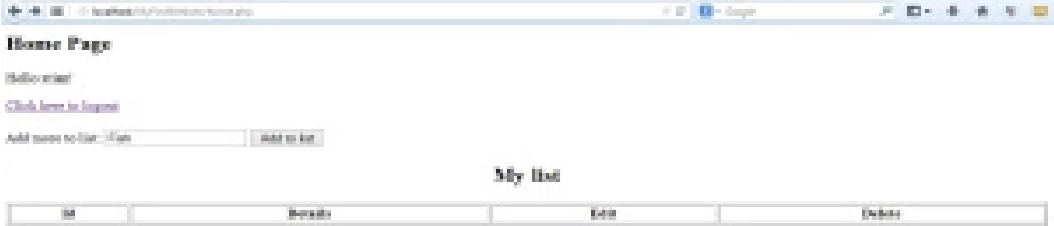
```
$details = mysql_real_escape_string($_POST['details']);
$time = strftime("%X"); //time
$date = strftime("%B %d, %Y"); //date

Print "$time - $date - $details";
```

?>

Take note that this ain't our official *add.php* syntax yet, I'm just going to demonstrate the time and date syntax and getting your input.

Now go back to your *home.php* and try to add an item then click on "Add to list".



This should be the following output on *add.php*:

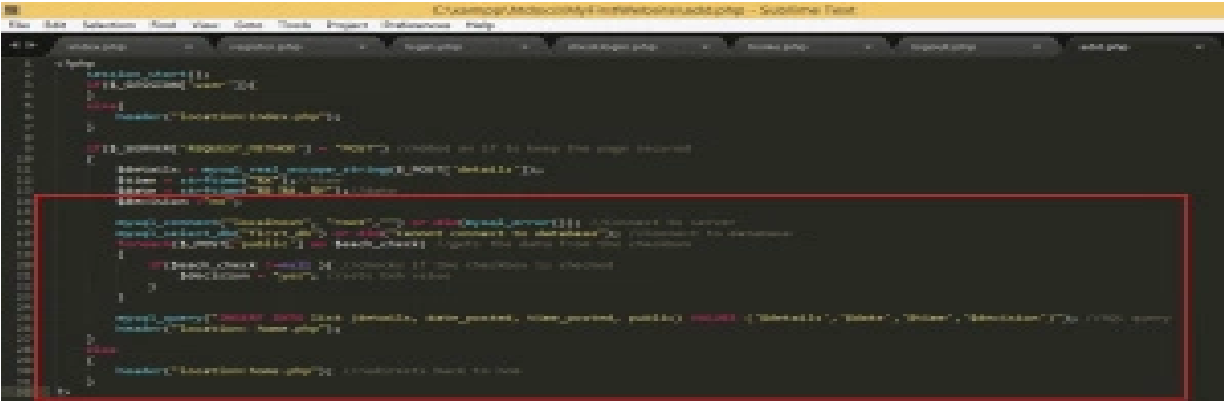


As you can see, we have our current time, date, and your input. Here's the explanation to the code:

- *strftime()* - get's the time based on what format your placed.
- *%X* - current system time.
- *%B* - current system month.
- *%d* - current system day.
- *%Y* - current system year.

Now let's modify our *add.php* and add the following data into the database together with the data from the checkbox:

add.php



Hide Shrink ▲ Copy Code

```
<?php
session_start();
if($_SESSION['user']){
}
else{
    header("location:index.php");
}

if($_SERVER['REQUEST_METHOD'] == "POST")
{
    $details = mysql_real_escape_string($_POST['details']);
    $time = strftime("%X"); //time
    $date = strftime("%B %d, %Y"); //date
    $decision = "no";

    mysql_connect("localhost","root","") or die(mysql_error()); //Connect to server
    mysql_select_db("first_db") or die("Cannot connect to database"); //Conect to database
    foreach($_POST['public'] in $each_check) //gets the data from the checkbox
    {
        if($each_check != null){ //checks if checkbox is checked
            $decision = "yes"; // sets the value
        }
    }

    mysql_query("INSERT INTO list(details, date_posted, time_posted, public) VALUES
('$details','$date','$time','$decision')"); //SQL query
    header("location:home.php");
}
else
{
    header("location:home.php");
}

?>
```

Click [here](#) for the complete add.php code

Here's a little explanation:

- foreach()** - gets the value of the checkbox. As you will notice, the checkbox format in the form is *name="checkbox[]"*. To get data from checkbox, it has to be instantiated as an array. Doing so would make it possible to get data from multiple checkboxes.

Now try entering some data and click "Add to list". In my case, I'll just use fish again. Let's go to our phpmyadmin and let's see if the data has been added. Here's the result of my case:

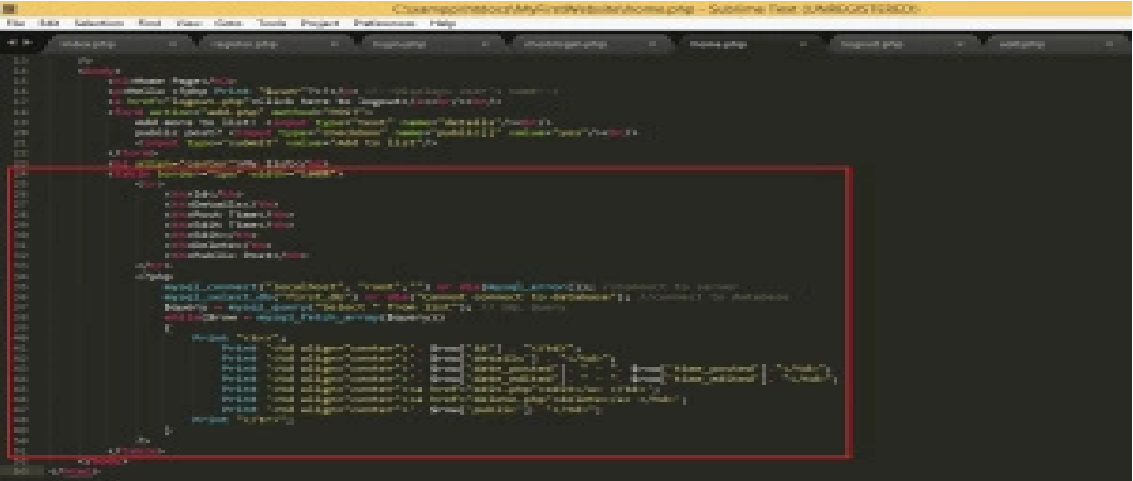
localhost/phpmyadmin



9.)Displaying data in the home page

Now that we have seen that the data has been successfully added. Let's now display the data in our home page. Let's modify our *home.php* and let's add some columns for the date:

home.php



Hide Shrink ▲ Copy Code

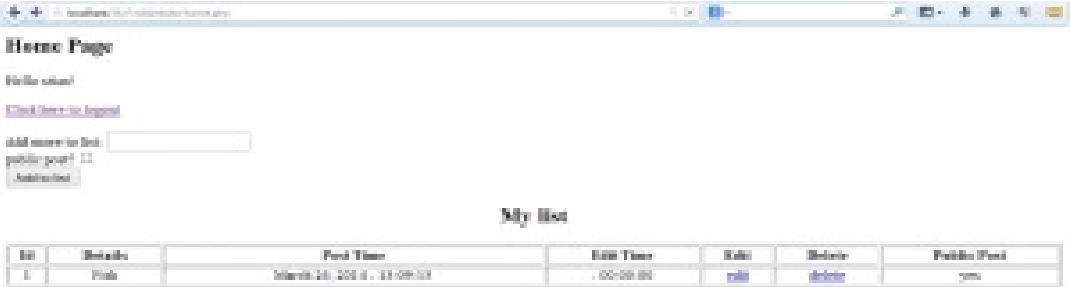
```
<html>
  <head>
    <title>My first PHP Website</title>
  </head>
  <?php
    session_start(); //starts the session
    if($_SESSION['user']){ // checks if the user is logged in
    }
    else{
      header("location: index.php"); // redirects if user is not logged in
    }
    $user = $_SESSION['user']; //assigns user value
  >
  <body>
    <h2>Home Page</h2>

    <hello>!
    <!--Display's user name-->
    <a href="logout.php">Click here to go logout</a><br/><br/>
    <form action="add.php" method="POST">
      Add more to list: <input type="text" name="details" /> <br/>
      Public post? <input type="checkbox" name="public[]" value="yes" /> <br/>
      <input type="submit" value="Add to list"/>
    </form>
    <h2 align="center">My list</h2>
    '
      Print '';
      Print '';
      Print '';
      Print '';
      Print '';
      Print '';
      Print '';
      Print '';
    }
  >
```

Id		Details	Post Time	Edit Time	Edit Delete	Public Post
. \$row['id']		. \$row['details']	. \$row['date_posted'] . " - " . \$row['time_posted'] . "	. \$row['date_edited'] . " - " . \$row['time_edited'] . "	edit delete	\$row['public']

</body> </html>

The explanation to the added code is quite simple. It just basically displays the data coming from the while loop. It has been explained earlier in our tutorial so I believe that by this point, you should have understood the process of getting the data in the query. Going back to the browser, try refreshing your *home.php* and see what happens:



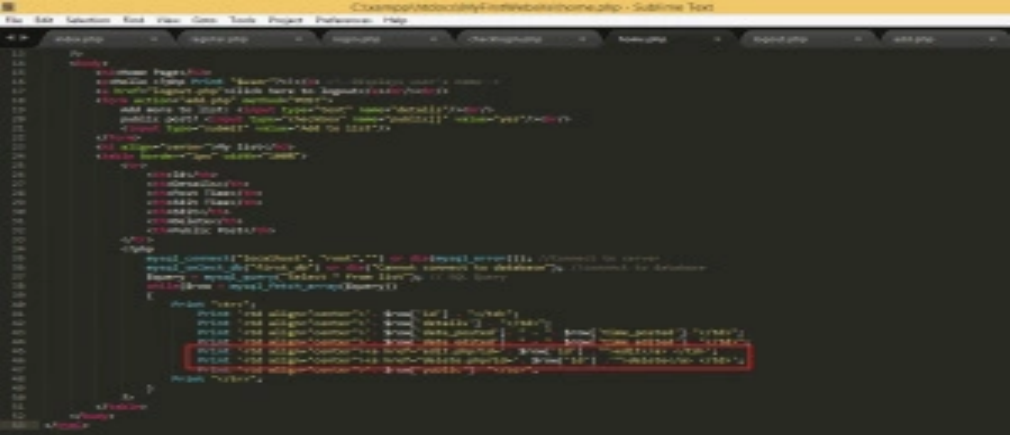
It should now display that data. From our CRUD checklist, we have now accomplished Create and Read. Next is to update(edit) and delete information. If you will notice we have edit and delete links displayed on the column. I'll add another data to the list named *"tuna"* to have another example and this time, it's privacy to *no*:



10.) Editing Data

Let's now try editing our data and to do that we will use a new functionality called *"GET"*. With our previous methods, we have been using *POST* as our http request but this time, let's use *GET* for editing and deleting records. To start of, let's modify our *home.php* and add a little code to 2 columns:

home.php



Hide Shrink ▲ Copy Code

```
<html>
  <head>
    <title>My first PHP Website</title>
  </head>
  <?php
    session_start(); //starts the session
    if($_SESSION['user']){ // checks if the user is logged in
    }
    else{
      header("location: index.php"); // redirects if user is not logged in
    }
    $user = $_SESSION['user']; //assigns user value
  >
  <body>
    <h2>Home Page</h2>

    <hello>!
    <!--Display's user name-->
    <a href="logout.php">Click here to go logout</a><br/><br/>
    <form action="add.php" method="POST">
      Add more to list: <input type="text" name="details" /> <br/>
      Public post? <input type="checkbox" name="public[]" value="yes" /> <br/>
      <input type="submit" value="Add to list"/>
    </form>
    <h2 align="center">My list</h2>
    '
      Print '';
      Print '';
      Print '';
      Print '';
      Print '';
      Print '';
      Print '';
```

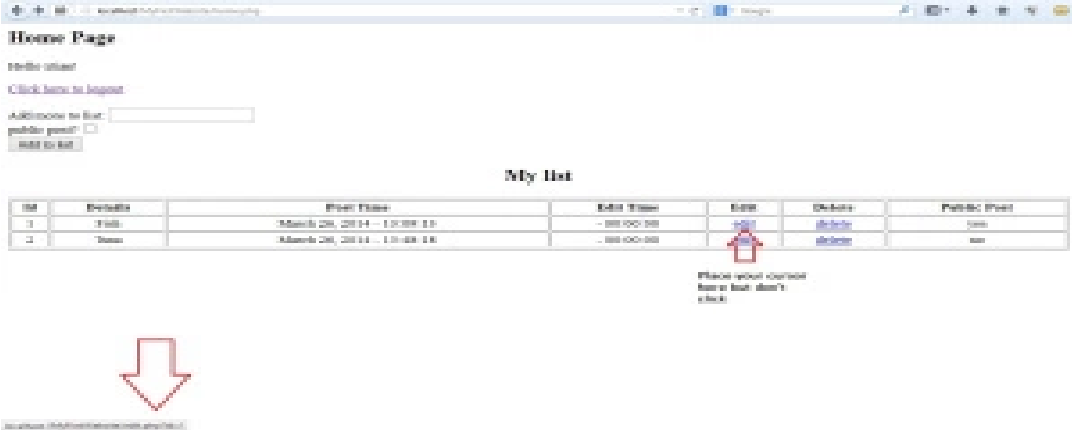
```
Print '';  
}  
?>
```

Id	Details	Post Time	Edit Time	Edit Delete	Public Post
' . \$row['id'] . "	' . \$row['details'] . "	' . \$row['date_posted'] . " - " . \$row['time_posted'] . "	' . \$row['date_edited'] . " - " . \$row['time_edited'] . "	edit delete	\$row['public']

</body> </html>

If you have noticed, we only added URL parameter for the edit and delete links namely *id*. We will be using this later to handle thee data. The reason why we use *id* is because it's a unique identifier. It is possible for the person to have enter the same data so it's not recommended to use the *details* as a mean for manipulation later on.

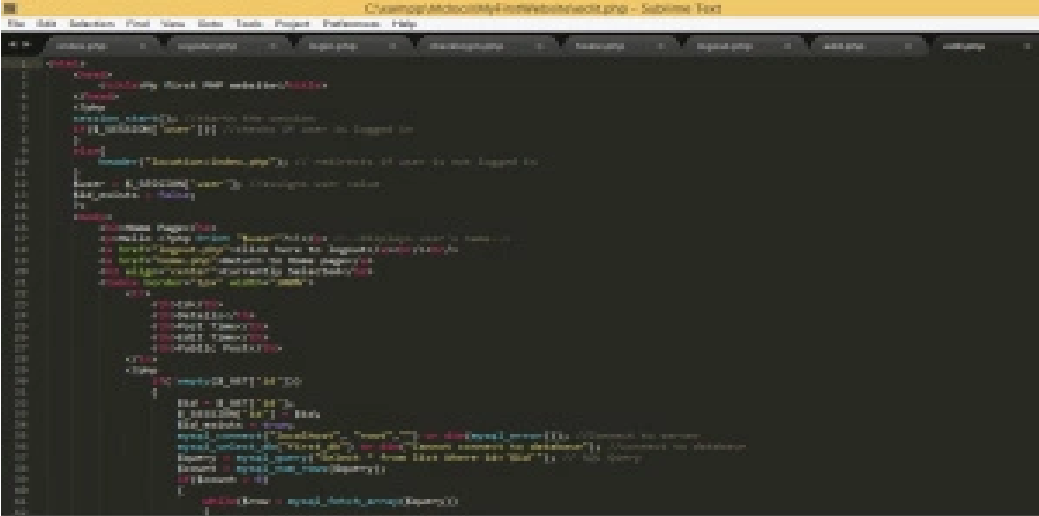
Try putting your cursor into the edit link and you will see the value of the id onthe lower left:



Now that we have that, let's try creating our *edit.php* and let's try to get the data from there:

edit.php

lines 1 - 42



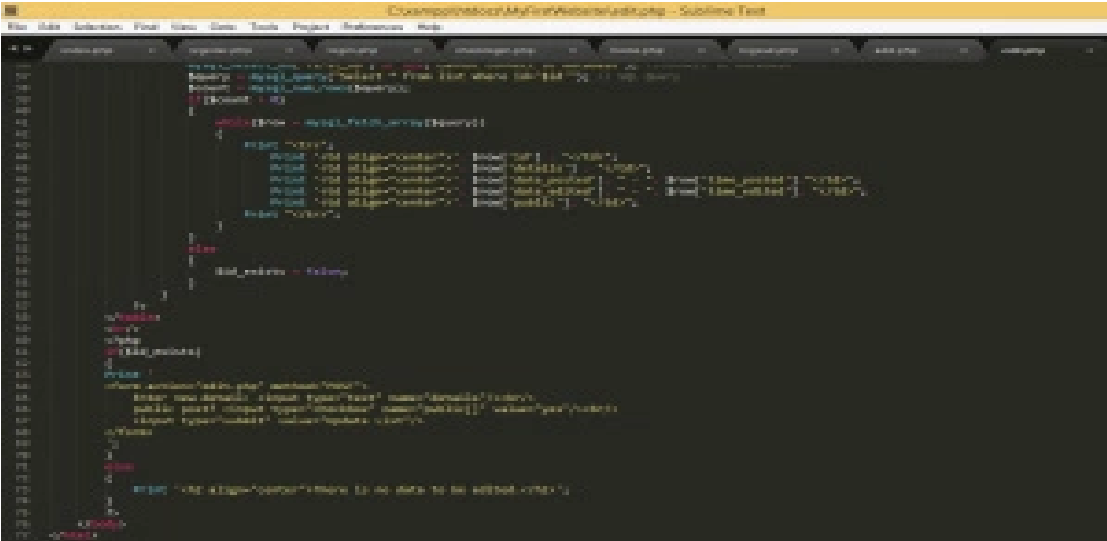
Hide Copy Code

```
<html>  
  <head>  
    <title>My first PHP Website</title>  
  </head>  
  <?php  
    session_start(); //starts the session  
    if($_SESSION['user']){ // checks if the user is logged in  
    }  
    else{  
      header("location: index.php"); // redirects if user is not logged in  
    }  
    $user = $_SESSION['user']; //assigns user value  
    ?>  
    <body>  
      <h2>Home Page</h2>  
  
    <hello>!  
    <!--Display's user name-->  
    <a href="logout.php">Click here to go logout</a><br/><br/>  
    <a href="home.php">Return to home page</a>  
    <h2 align="center">Currently Selected</h2>  
    0)  
    {  
      while($row = mysql_fetch_array($query))  
      {  
        Print '';  
      }  
    }  
  }  
}
```

Some explanations to the code:

- **!empty()** - a method that checks if the value is not empty. The syntax can be reversed if you want to check if it's empty by removing the explanation point (!), therefore it's syntax would be **empty()**.
- **\$_GET['id']** - Used to get the value from the parameter. In our case, we use *id* as our URL parameter so the syntax would be **\$_GET['id']**.

- **`$id_exists`** - the variable that checks whether the given id exists.
- **`$_SESSION['id']`** - we place the value of id into session to use it on another file.



lines 42 - 76

Hide Copy Code

```
Print "<tr>";
Print '

```

```

"; Print ""; Print ""; Print ""; Print ""; Print ""; } } else { $id_exists = false; } } ?>

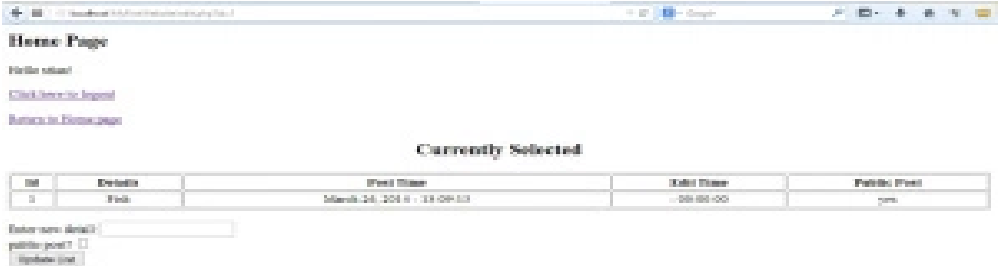
```

Id	Details	Post Time	Edit Time	Public Post
' . \$row['id'] . '	' . \$row['details'] . '	' . \$row['date_posted'] . ' - ' . \$row['time_posted'] . '	' . \$row['date_edited'] . ' - ' . \$row['time_edited'] . '	' . \$row['public'] . '

 Enter new detail: <input type="text" name="details"/>
 public post? <input type="checkbox" name="public[]" value="yes"/> <input type="submit" value="Update List"/> </form> ' ; } else { <h2 align="center">There is not data to be edited.</h2> } ?> </body> </html> [Click here for the complete edit.php code \(Only refer to the front-end part. Github Sourced\)](#)

The reason why we are putting the variable `$id_exists` is because in case the user modifies the URL parameter into a non-existing number(in our case we only have 2 rows/id's), we can display a prompt in which the data doesn't exist.

Now try clicking the edit link into the first row and it should display like this:



Complete edit.php code:

Hide Shrink Copy Code

```
<html>
<head>
<title>My first PHP Website</title>
</head>
<?php
session_start(); //starts the session
if($_SESSION['user']){ // checks if the user is logged in
}
else{
header("location: index.php"); // redirects if user is not logged in
}
$user = $_SESSION['user']; //assigns user value
?>
<body>
<h2>Home Page</h2>

<hello>!
<!--Display's user name-->
<a href="logout.php">Click here to go logout</a><br/><br/>
<a href="home.php">Return to home page</a>
<h2 align="center">Currently Selected</h2>
0)
{
while($row = mysql_fetch_array($query))
{
Print "";
Print '
Print '
Print '
Print '
Print '
Print '
Print '
}
}
}
```

```
        else
        {
            $id_exists = false;
        }
    }
    ?>
```

Id	Details	Post Time	Edit Time	Public Post
' . \$row['id'] . "	' . \$row['details'] . "	' . \$row['date_posted'] . " - " . \$row['time_posted'] . "	' . \$row['date_edited'] . " - " . \$row['time_edited'] . "	' . \$row['public'] . "

 Enter new detail: <input type="text" name="details"/>
 public post? <input type="checkbox" name="public[]" value="yes"/> <input type="submit" value="Update List"/> </form> '; } else { <h2 align="center">There is not data to be edited.</h2> } ?> </body> </html> [Click here for the complete edit.php code \(Only refer to the front-end code. Github sourced\)](#)

Let's try modifying the URL parameter by removing *?id=1* and now should result to *localhost/MyFirstWebsite/edit.php* and it should result like this:



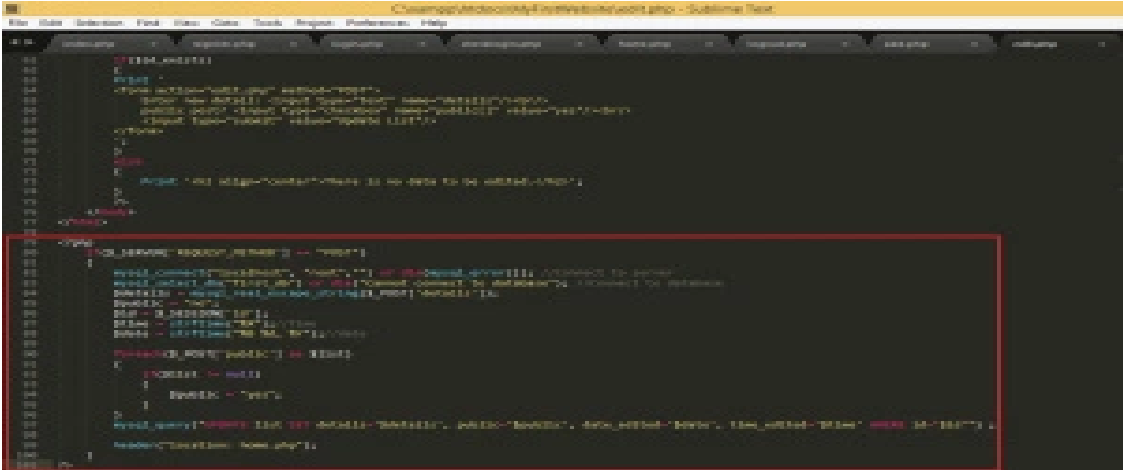
Now try putting a value that is greater than the id number, in our case, let's try 5 and it should result like this:

localhost/MyFirstWebsite/edit.php?id=5



Now that we secured our URL parameters, lets now place the edit syntax. Let's go back to *edit.php* and add some following code to update the information to the database:

edit.php



Hide Copy Code

```
<?php
if($_SERVER['REQUEST_METHOD'] == "POST")
{
    mysql_connect("localhost", "root", "") or die (mysql_error()); //Connect to server
    mysql_select_db("first_db") or die ("Cannot connect to database"); //Connect to database
    $details = mysql_real_escape_string($_POST['details']);
    $public = "no";
    $id = $_SESSION['id'];
    $time = strftime("%X"); //time
    $date = strftime("%B %D, %Y"); //date

    foreach($_POST['public'] as $list)
    {
        if($list != null)
        {
            $public = "yes";
        }
    }

    mysql_query("UPDATE list SET details='$details', public='$public', date_edited='$date',
time_edited='$time' WHERE id='$id'");
    header("location:home.php");
}
?>
```

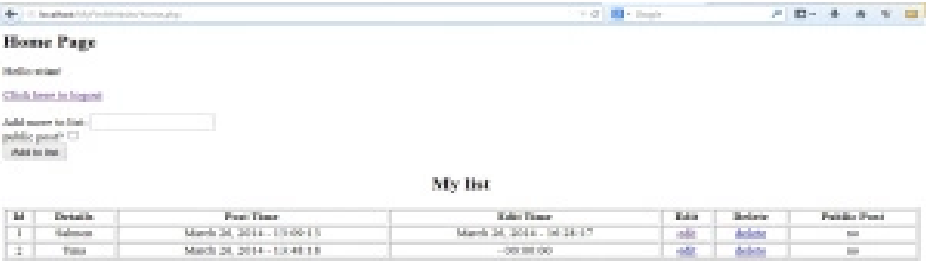
Click [here](#) for the complete edit.php code (Github)

Now try refreshing and go back to the edit page. Let's try a different data. In my case, I'll be using "Salmon" and change it to non-public:

edit.php?id=1



Go ahead and click Update list and you should be redirected to *home.php* and see the updated list.



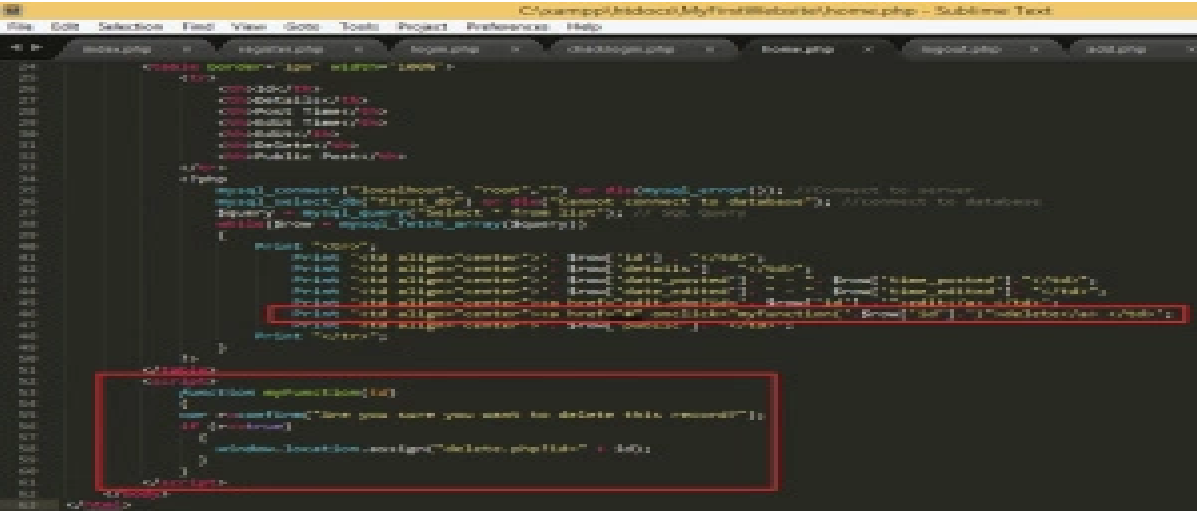
Now, we have a time and date of edit displayed on the *Edit Time* column. Our privacy has been set now to non-public and the value has changed into *Salmon*.

Congratulations! We have now done the edit function!

11.) Deleting data

Looking back to the CRUD, we have now done creating(adding), reading(displaying), and updating(editing) records. Now for the last part, deleting records. For this part it's just relatively the same as what we have done on edit but what differs is just the SQL statement. Instead of using UPDATE, we will be using the DELETE syntax. In deleting records, we have to prompt people making sure that they'd really want to delete the record so we will be adding a little javascript in *home.php*. To do that, let's modify our code and add some int *home.php*

home.php



Hide Shrink Copy Code

```
<table border="1px" width="100%">

    Id
    Details
    Post Time
    Edit Time
    Edit
    Delete
    Public Post

';

    Print '. $row['id'] . " ";
    Print '. $row['details'] . " ";
    Print '. $row['date_posted'] . " - " . $row['time_posted'] . " ";
    Print '. $row['date_edited'] . " - " . $row['time_edited'] . " ";
    Print '<a href="edit.php?id='.$row['id'].'">edit';
    Print '<a href="#" onclick="myfunction('.$row['id'].')">delete</a>';
    Print '. $row['public'] . ' ';
    Print ' ';
}

?>

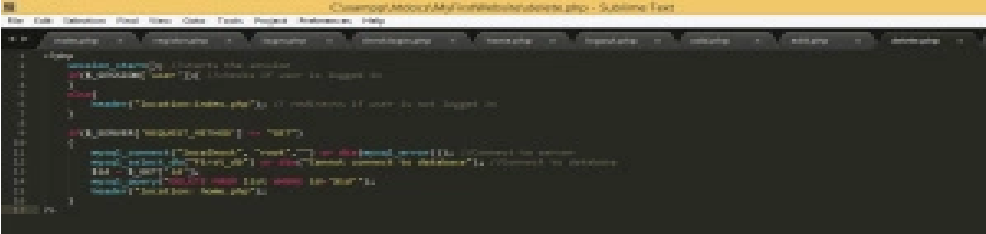
<script>
function myFunction(id)
```

```
{
    var r = confirm("Are you sure you want to delete this record?");
    if(r == true)
    {
        window.location.assign("delete.php?id=" + id);
    }
}
</script>
```

Click [here](#) for the complete home.php code

As you have noticed, we edited the link for the delete. We changed *href* into *"#"* and added and *onclick* function for Javascript for the method of *myFunction* and inside it's parameter is the id of the row. Below the table written is the Javascript syntax wherein it prompts the user if he/she want's to delete the record. If the user confirms, the page then directs to *delete.php* together embedded with the value of the *id*. Now let's create *delete.php* and here's the following syntax:

delete.php



Hide Copy Code

```
<?php
session_start(); //starts the session
if($_SESSION['user']){ //checks if user is logged in
}
else {
    header("location:index.php"); //redirects if user is not logged in.
}

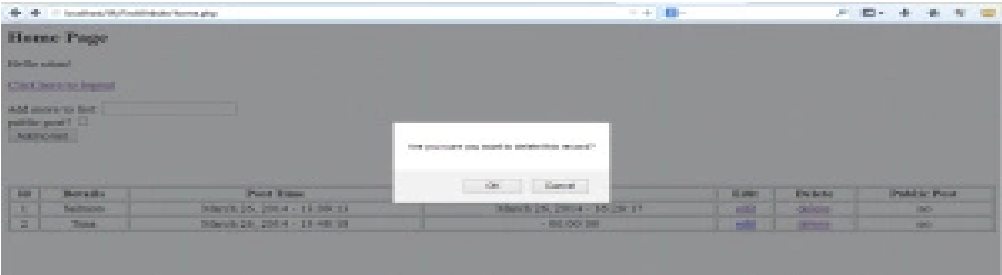
if($_SERVER['REQUEST_METHOD'] == "GET")
{
    mysql_connect("localhost", "root", "") or die(mysql_error()); //connect to server
    mysql_select_db("first_db") or die("cannot connect to database"); //Connect to database
    $id = $_GET['id'];
    mysql_query("DELETE FROM list WHERE id='$id'");
    header("location:home.php");
}

?>
```

Click [here](#) for the complete delete.php code

The code is just simple and the syntax are also the ones that we used before but noticed we have changed our request method into *GET*. We are now using the *GET* request since we have a URL parameter. Now try refreshing *home.php* and let's try deleting the first record. This should be the result:

Prompting:



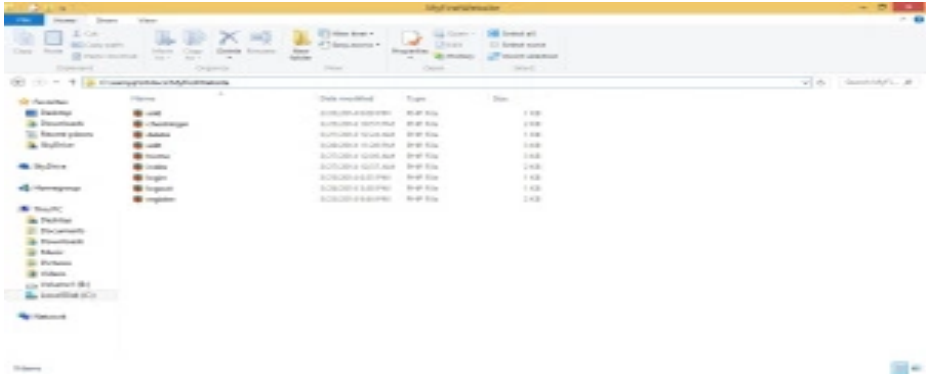
End-result:



Congratulations! Now we have now officially completed our CRUD statements!

12.) Displaying public data

Now for the last part, displaying public data. We will be displaying data that's been set to *yes* in our *index.php*, in which is a page for non-authenticated users. It's very simple. We just have to edit our *index.php* and add a some php code and table. Here's our updated *index.php*:



As I mentioned earlier, I'll be uploading the finished product. Click [here](#)

I also uploaded the sql file for the database in case you want to use the exact data that I have. To import the sql file, you can just simply click on *export* and select the sql file but make sure to create the database first. If you like, I can create a separate blog for that. Just let me know in the comments section.

Anticipated questions

Here are the questions that I think you will be asking. I tried to anticipate these to avoid confusions:

- Whenever I register a new user, the same list appear. I was expecting for each different user, there would be a separate list for them. Did I do something wrong?

That's actually a good question. As you can see, I'm only teaching BASIC CRUD functionalities, authentication, and security not a membership system. Creating a separate list per user requires a separate table for each of them. So the logic behind it is for each time after you insert data, you will insert it in another table and have it relationships to one another. You can use the CREATE syntax for creating a separate table for the user which goes something like this:

Create table \$username (column1 int auto_increment, column2 varchar(50), etc...)

You can click [here](#) for the Create table syntax.

The reason why I didn't do that part is because your gonna have to do a lot of querying and it would make my code look complicating which can consumes a lot of time developing it so it's definitely out of my scope. It would be better if you just try it out yourself and figure out the way. After all, everything you need is already in this blog.

- Would it be easier to use master pages? Because i've noticed you had to re-write the server and database connection all over again for each page

Yes, it is easier to use master pages. The reason why I didn't use it is for our readers to easily understand the logic behind it. Maser pages can make it a bit confusing for starters.

- Why not use a framework? I bet that would look neat and easy.

Then again, I mentioned in the title *Creating your First PHP Program **FROM SCRATCH***. It's nice to start of from nothing that from a "template" so that our reader would know the code by heart.

- I know that this is might be an old way of developing PHP. I've heard that there are new features released.

I only presented based on my mastery. I've been working with PHP for a long time now and I might not heard about the new features. If you think the new one makes it look good then try it out then.

- Why not create a separate file for the CSS and Javascript?

Simple, i'm only demonstrating you about the functionalities. Were more on the back-end code rather than the front-end. I really didn't focused on the design. If you ask how to design your code with the PHP, well it's as easy as styling a regular HTML file.

Well I guess that concludes everything. Thanks guys I hope you learned something from this topic.

Just to reiterate, in case you want to learn making your own ASP.NET MVC 5 web application from scratch, you can check it out right [here](#). Follow me on [Twitter](#) and you can tweet out your feedbacks and comments on either articles.

License

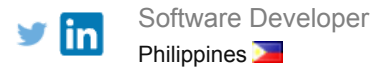
This article, along with any associated source code and files, is licensed under [The Code Project Open License \(CPOL\)](#)

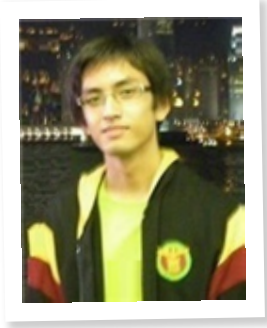
Share



About the Author

Kristian Guevara





A Software Architect, Full-stack developer and a visionary residing in the Philippines. An enthusiast of Microsoft and Open-source technologies particularly from the web, extending to the emerging Big Data and Cloud technology. Out of the keyboard, he love's reading books and watching documentary videos.

Visit him at his website: <https://xtianguevara.com/>

Comments and Discussions

You must [Sign In](#) to use this message board.

Search Comments

Spacing

Relaxed

Layout

Normal

Per page

25

Update

First

Prev

Next

	thanks		4-Aug-18 19:28
	Message Closed		30-Jul-18 1:54
	Message Closed		26-Mar-18 8:44
	Message Closed		26-Mar-18 8:44
	PHP Tutorial for beginners - My Recommendation	mwilliams530	6-Oct-17 7:59
	Bug in PHP Code - Register.php	mwilliams530	4-Oct-17 8:22
	Re: Bug in PHP Code - Register.php	mwilliams530	5-Oct-17 9:24
	Hi Sir		25-Sep-17 23:39
	Re: Hi Sir		13-Jan-18 1:28
	unable to register user on the registration page		2-Sep-17 13:19
	Message Closed		18-Jul-17 8:30
	Message Closed	azazqadir	5-Jun-17 20:44
	Re: Good introduction to PHP programming	Kristian Guevara	21-Jun-17 0:29
	Re: Good introduction to PHP programming	mwilliams530	4-Oct-17 8:55
	Fatal error: Uncaught Error: Call to undefined function mysql_real_escape_string() in C:\xampp\htdocs\Blogphp\register.php:18 Stack trace: #0 {main} thrown in C:\xampp\htdocs\Blogphp\register.php on line 18		18-Apr-17 2:00
	Re: Fatal error: Uncaught Error: Call to undefined function mysql_real_escape_string() in C:\xampp\htdocs\Blogphp\register.php:18 Stack trace: #0 {main} thrown in C:\xampp\htdocs\Blogphp\register.php on line 18	Kristian Guevara	18-Jun-17 5:41
	Mac-OS Tutorial		4-Mar-17 5:21
	Re: Mac-OS Tutorial	Kristian Guevara	25-Mar-17 3:33

<div><div></div><div>?</div></div> <div>Password_hash ? Help</div> <div></div>	<div><div></div><div></div></div>	<div>9-Feb-17 9:26</div>
<div><div></div><div></div></div> <div>Re: Password_hash ? Help</div> <div></div>	<div><div></div><div>Kristian Guevara</div></div>	<div>17-Feb-17 0:06</div>
<div><div></div><div>?</div></div> <div>Checklogin</div> <div></div>	<div><div></div><div>james morris</div></div>	<div>1-Feb-17 7:40</div>
<div><div></div><div></div></div> <div>Re: Checklogin</div> <div></div>	<div><div></div><div>Kristian Guevara</div></div>	<div>17-Feb-17 0:02</div>
<div><div></div><div>?</div></div> <div>Thanks ;)</div> <div></div>	<div><div></div><div>Et-Cetera</div></div>	<div>18-Jan-17 9:28</div>
<div><div></div><div></div></div> <div>Re: Thanks ;)</div> <div></div>	<div><div></div><div>Kristian Guevara</div></div>	<div>22-Jan-17 5:03</div>
<div><div></div><div>?</div></div> <div>Message Closed</div> <div></div>	<div><div></div><div></div></div>	<div>23-Dec-16 1:42</div>

Last Visit: 31-Dec-99 18:00

Last Update: 28-Sep-18 4:31

Refresh

1234567Next»

- General
- News
- Suggestion
- Question
- Bug
- Answer
- Joke
- Praise
- Rant
- Admin

Use Ctrl+Left/Right to switch messages, Ctrl+Up/Down to switch threads, Ctrl+Shift+Left/Right to switch pages.