



FRESCHESOLUTIONS

PHP 101

Mike Pavlak, IT Strategist

NHMUG
New Hampshire
Midrange User Group

Corporate Overview



40+ years in business

- ▶ Management and employee owned
- ▶ 300+ employees
- ▶ 18 yr average employee tenure
- ▶ Named one of Canada's top small and medium employers



22,000+ customers

- ▶ 8 offices around the world
- ▶ North America, Europe, Latin America and Asia Pacific
- ▶ Over 200 partners/resellers



Application Management and Modernization Experts

- ▶ 15+ years application optimization, integration, modernization expertise
- ▶ 40 years of IBM i product development



Innovative Products and Services

- ▶ Web, mobile and GUI modernization
- ▶ Analysis & productivity
- ▶ Application & database modernization

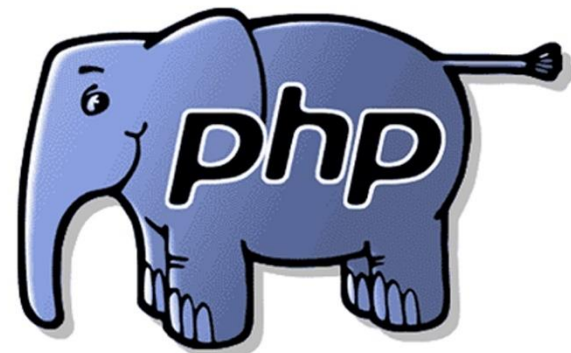
FRESCHESOLUTIONS

BCD looksoftware

QUADRANT
SOFTWARE

Target Audience

- Interested in leveraging web technology and IBM i
- Learn more about how PHP integrates with IBM i
- Understand the development path from RPG to PHP
- Managing team of IBM i developers & looking to do more
- New to PHP in the IBM i arena



Agenda

- Brief intro to PHP
 - ▶ Variables, functions, etc.
- Explore database file listing (load all subfile)
- Explore IBM i Program Call

- Wrap up / Q/A

PHP101: Intro to PHP

PHP Script Structure

```
<?php  
    echo 'hello world';  
?>
```

```
<?php echo 'hello world'; ?>
```

- **<?php** begins block of PHP code
 - ▶ Either in HTML file or alone
 - ▶ Optionally terminate using **?>**
 - ▶ Semicolon terminates simple statements
 - ▶ May embed multiple blocks of PHP code in single file
 - ▶ Can string multiple commands in single line

Rules for Variables

- Case sensitive
- Must begin with dollar sign “\$”
 - ▶ \$thisIsMyVariable
 - ▶ \$_AnotherVariable
 - ▶ \$ this is not a variable
- Implicit casting
- Can be re-typed (Dynamic typing)
- Constant – Value established in script that does not change
 - ▶ Define(‘SPEAKER’, “Mike Pavlak”);

```
<?php
```

```
$field1 = 5;  
$field2 = 10.3;  
$field3 = $field1 + $field2;  
$field1 = “Hello World!”;
```

```
?>
```

The 8 or 9 Data Types

■ Scalar

▶ Integer

- -2,147,483,648 thru 2,147,483,647
- Supports decimal, octal and hex representation

▶ Floating-Point

- 1.7E-308 thru 1.7E+308
- 15 digits of decimal precision

▶ Strings

- Big. Really big. Too big to discuss!

▶ Boolean

- False is 0, 0.0, false keyword, empty string, object w/no values, null. All others are true

- Object, array, null and resource (PHP 7 supports callable)

Variables...(cont.)

- Scope
 - ▶ Global - Available everywhere but inside function (sort of)
 - ▶ Local - Available only in a function, destroyed at end
 - ▶ Static - Available only in a function, but remains
- Arrays (three types)
 - ▶ Enumerated
 - ▶ Associative
 - ▶ Multi-dimensional

Arrays: Value assignment

```
1 <pre><?php
2
3 //  enumerated array...
4 $animals = ['Dog', 'Cat', 'Hamster'];
5
6 //  associative array...
7 $barnYard = [    'Cow'=>'calf',
8                  'Horse'=>'foal',
9                  'Chicken'=>'chick'];
10
11 //  multi-dimensional array...
12 $farm = [    'barn1'=>$animals,
13              'barn2'=>$barnYard];
14
15 print_r($animals);
16
17 print_r($barnYard);
18
19 print_r($farm);
20
21 ?></pre>
```

```
Array
(
    [0] => Dog
    [1] => Cat
    [2] => Hamster
)
Array
(
    [Cow] => calf
    [Horse] => foal
    [Chicken] => chick
)
Array
(
    [barn1] => Array
        (
            [0] => Dog
            [1] => Cat
            [2] => Hamster
        )
    [barn2] => Array
        (
            [Cow] => calf
            [Horse] => foal
            [Chicken] => chick
        )
)
```

Strings

- Most of PHP is character strings
- Single quotes
 - Variables not expanded
 - Heavy use of concatenation “.”
 - `$string_1 = 'This is the value of variable x: ' . $x`
- Double quotes
 - Variables interpolation
 - `$string_1 = “This is the value of variable x: $x”`

Operators

- Most common operators apply (+, -, *, /, etc.)
- Concatenation
 - ▶ \$greeting = 'My name is '
 - ▶ DEFINE (TEACHER, 'Mike Pavlak')
 - ▶ \$salute = \$greeting . TEACHER (My name is Mike Pavlak)
- Increment / decrement
 - ▶ \$a++, ++\$a (\$a = \$a + 1)
 - ▶ \$a--, --\$a (\$a = \$a - 1)
- Logical operators
 - ▶ && and
 - ▶ || or
 - ▶ ! not

Comments

- A brief comment about comments
- `//` (C++ style)
 - indicates single line comment
 - May use at end of line of live code
 - Easily comments a line of code
- `/* ... */` (C Style)
 - Looks like CL? Pretty close
 - Comment block,
 - Can span multiple lines
- Shell style `#`

How many equal signs?

- = Single equal sign is assignment
 - ▶ `$X=3`
 - ▶ `$Y=$X`
 - ▶ Now `$Y = 3`
- == Double equal sign is for conditions
 - ▶ If `($x==$y) { do something} else {do something else}`
 - ▶ If you see single equal in condition, assignment will occur
- === Triple equal sign is exact equal conditions
 - ▶ `$X= 3 (Integer) $Y=3.0 (Float)`
 - ▶ If `($x==$y)` will resolve to **true**
 - ▶ If `($x=== $y)` will resolve to **false**

Flow control

- If (condition is always in parenthesis)

- ▶ Work

- Else

- ▶ Work

```
11 //One line
12 if ($x==5)
13     $x++;
14 else $x--;
```

```
16 //Multi line
17 if ($x==5) {
18     $x++;
19     $y++;
20 }
21 else {
22     $x--;
23     $y--;
24 }
```

Flow control (cont.)

■ Switch(condition) {

▶ Case "l" :

- Do something;
- Break;

▶ Case "k"

- Do something else;
- Break;

▶ Default

- Do another thing;
- Break;

▶ }

```
35 //Switch, is value 1-3
36
37 switch ($x) {
38     case 1 :
39         echo "Number One";
40         break;
41     case 2 :
42         echo "Number Two";
43         break;
44     case 3 :
45         echo "Number Three";
46         break;
47     default:
48         echo "Some other number!";
49 }
```


How do you make a switch like a select?

- Move condition
- Comparison with value
- Still need break!

```
57 switch (TRUE) {  
58     case $x==1 :  
59         echo "Number One";  
60         break;  
61     case $x==2 :  
62         echo "Number Two";  
63         break;  
64     case $x==3 :  
65         echo "Number Three";  
66         break;  
67     default:  
68         echo "Some other number!";  
69 }
```

Iterative structures

```
While (condition) {  
    Do something  
}
```

```
72 while ($x < $y){  
73     echo $x;  
74     $x++;  
75 }
```

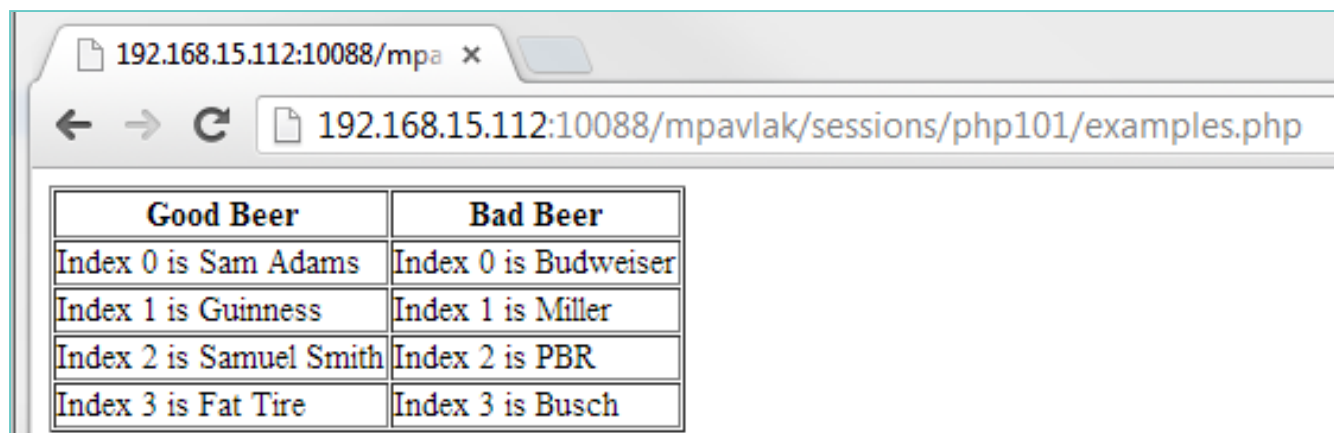
```
Do {  
    Something  
} while ($x<$y)
```

```
78 do {  
79     echo $x;  
80     $x++;  
81 } while ($x<$y)
```

Foreach

■ Jon's favorite function!

```
87 $goodBeer = array('Sam Adams', 'Guinness', 'Samuel Smith', 'Fat Tire');
88 $badBeer = array('Budweiser', 'Miller', 'PBR', 'Busch');
89 echo '<table border="2"><th>Good Beer</th><th>Bad Beer</th>';
90 foreach ($goodBeer as $beerkey=>$beerval) {
91     echo "<tr><td>Index $beerkey is $beerval</td>";
92     echo "<td>Index $beerkey is {$badBeer[$beerkey]}</td></tr>";
93 }
94 echo '</table>';
```

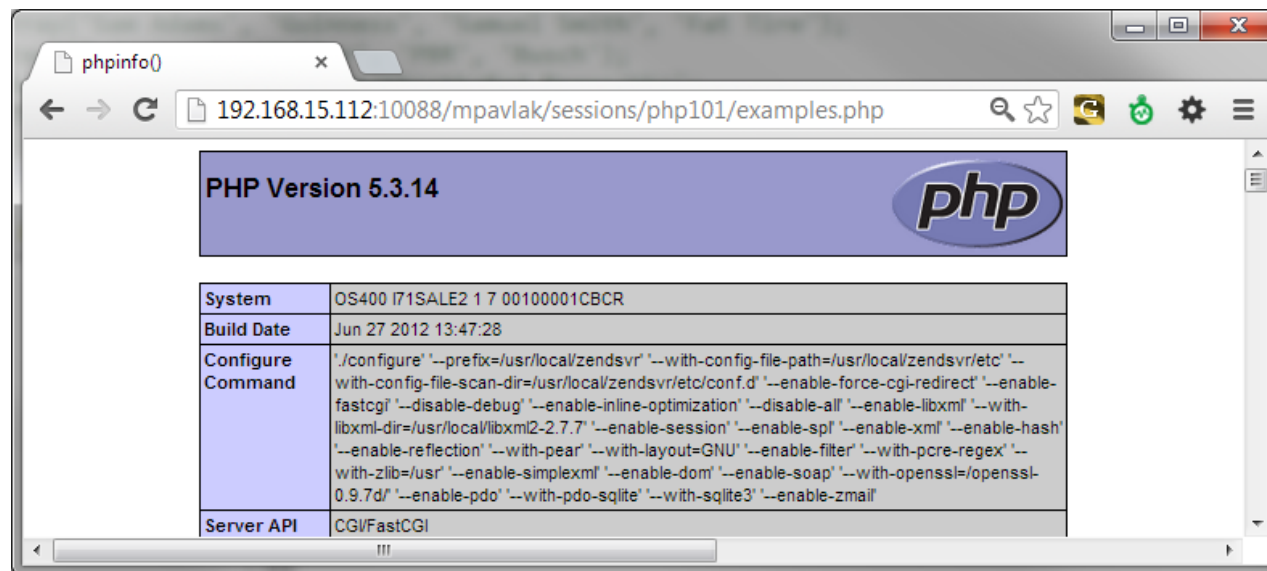


Good Beer	Bad Beer
Index 0 is Sam Adams	Index 0 is Budweiser
Index 1 is Guinness	Index 1 is Miller
Index 2 is Samuel Smith	Index 2 is PBR
Index 3 is Fat Tire	Index 3 is Busch

Most of PHP is Functional

- Like sub procedures...it you know them you got it!
- Loosely like subroutines, but not quite
 - ▶ Pass parameters
 - ▶ Organize work
 - ▶ Get results

```
100 <?php
101
102 phpinfo();
103
104 ?>
```



Functions come from three places

■ Built in

- ▶ Part of core PHP
- ▶ Available everywhere
- ▶ `phpinfo()`; `strtoupper()`; `array()`;

■ Extensions

- ▶ Usually included with PHP
- ▶ Requires local Compiler, tec.
- ▶ `DB2_connect`; `odbc_fetch_row()`;

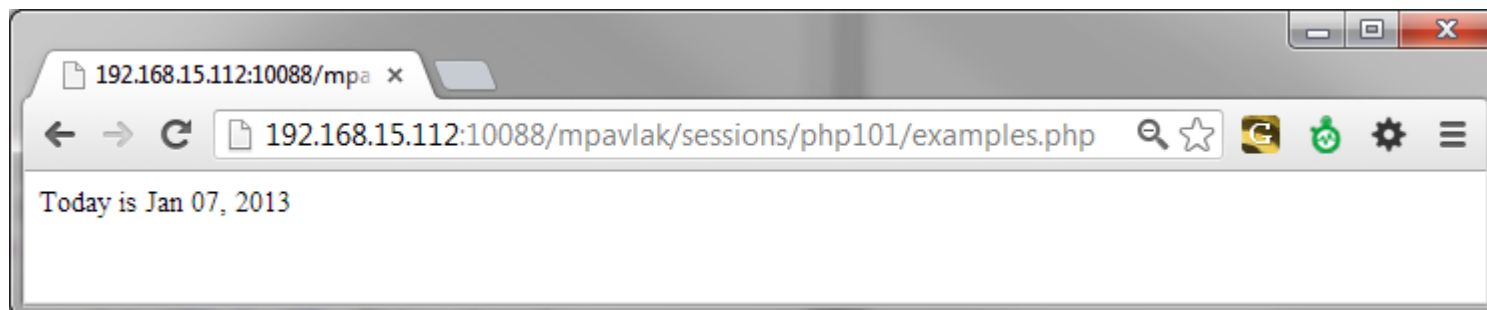
■ User created

- ▶ Like subprocedures

Function is like a Subprocedure

- Code it first, then use (not a hard rule)

```
99 function getMikeyDate($date='today') {  
100     if ($date == 'today')  
101         $timestamp = time();  
102     else $timestamp=$date;  
103  
104     $date = date("M d, Y", $timestamp);  
105     return $date;  
106 }  
107  
108 $today = GetMikeyDate();  
109 echo "Today is $today";
```

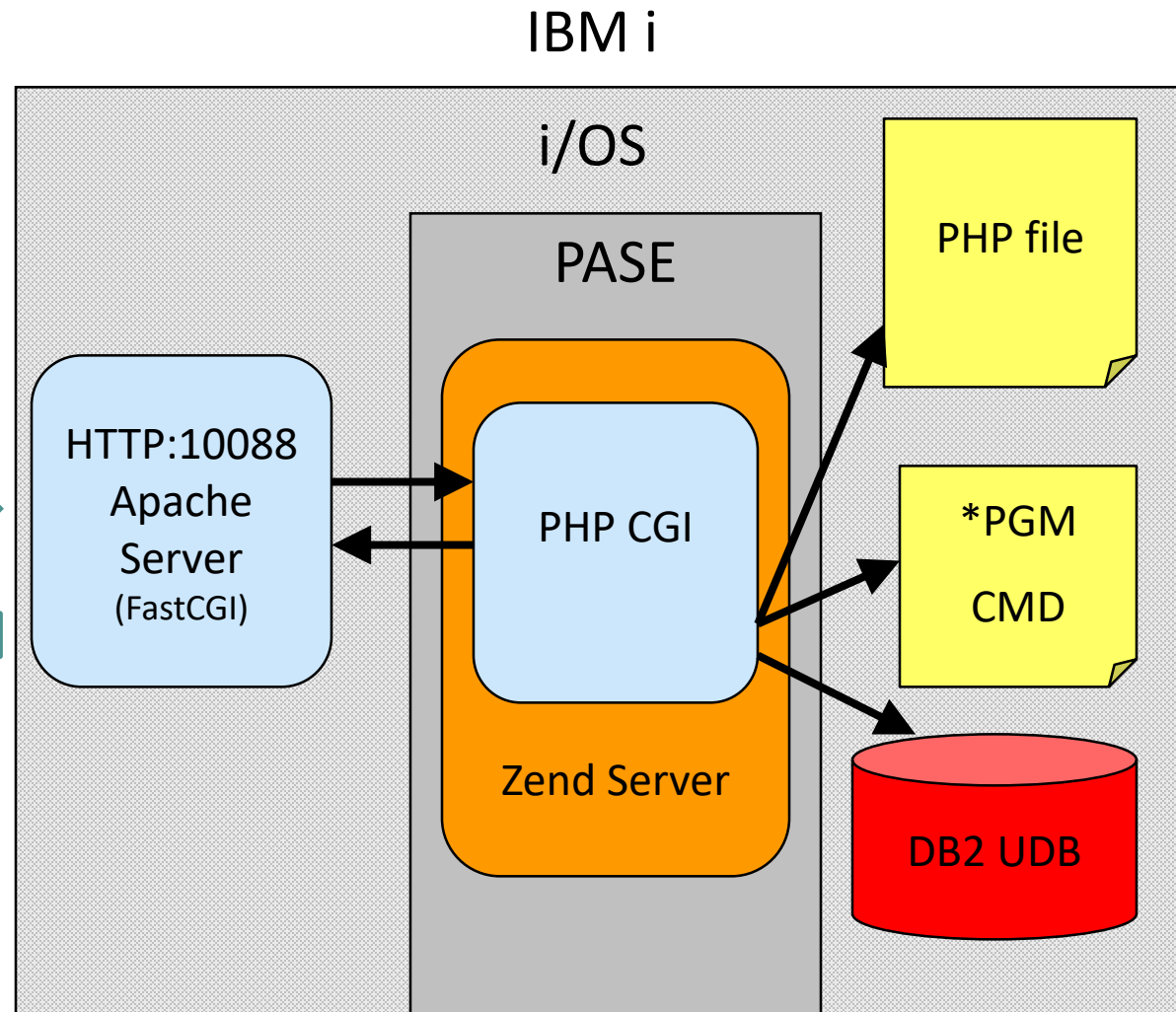
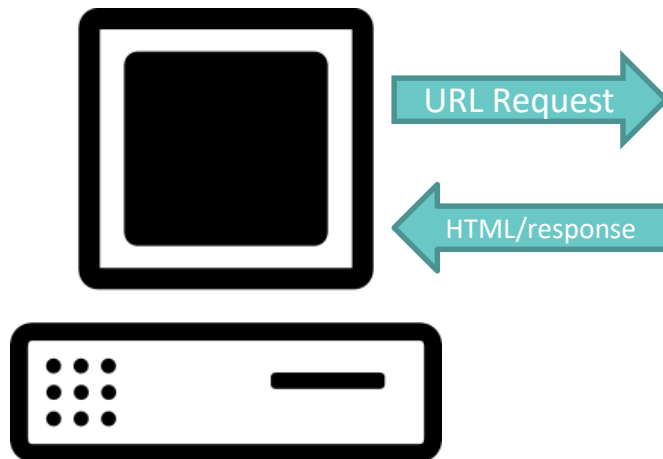


PHP101: Database Listing

Zend Server, under the covers

ILE Apache:10088

- Default configuration FastCGI



Employee list with Payroll

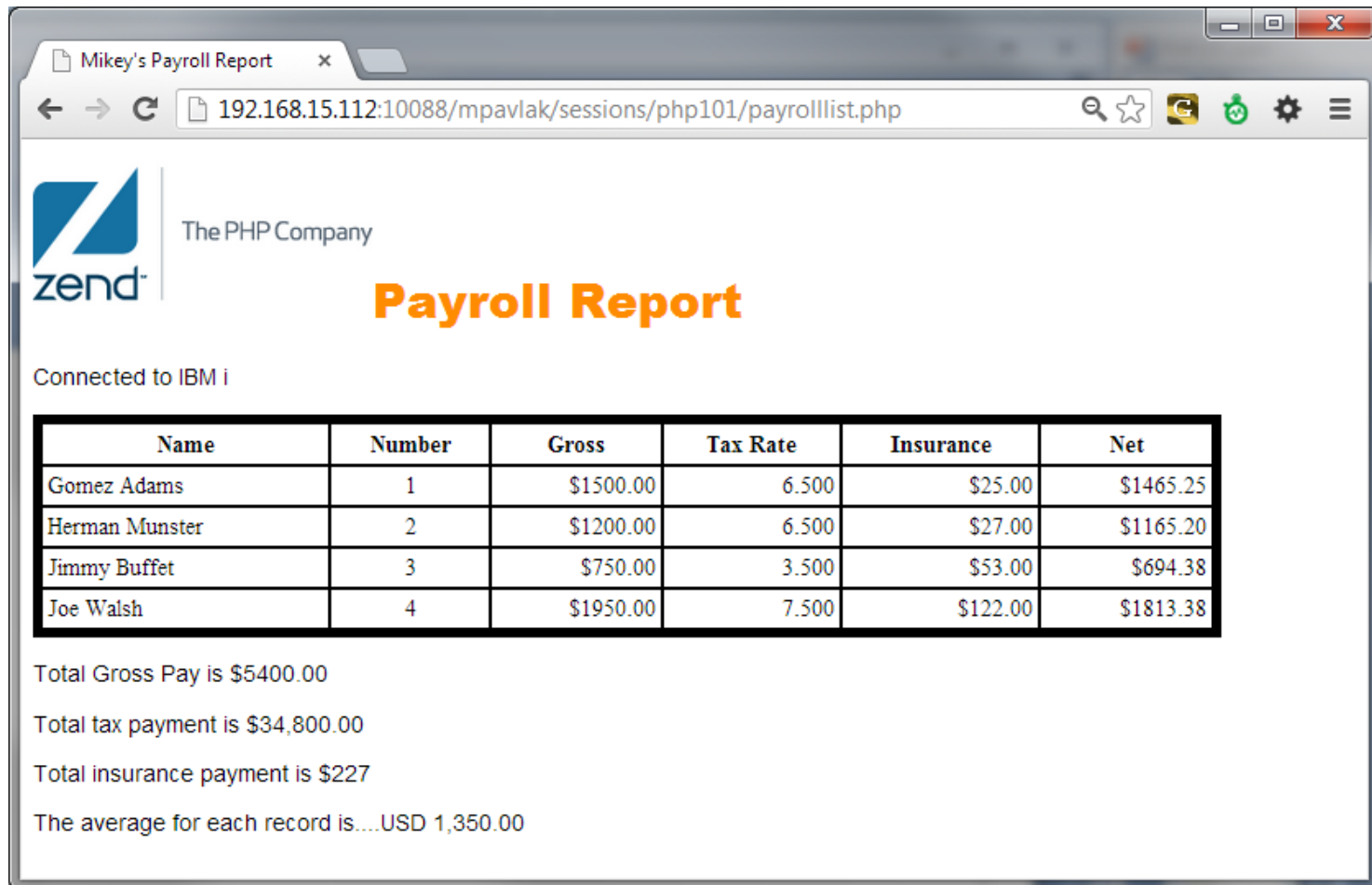
- Describe the file
- Explore database functions
- Perform some calculations
- Put the output to the web server

Describe the data

- Raw data in STRSQL display...
- Using long field names as both long and short are available to PHP via SQL.


```
EMPLOYEE_NAME      EMPNUM  EMPLOYEE_WEEK_PAY  EMPLOYEE_TAX_RATE  EMPINS
Gomez Adams        1        1,500.00          6.500             25.00
Herman Munster     2        1,200.00          6.500             27.00
Jimmy Buffet       3         750.00           3.500             53.00
Joe Walsh          4        1,950.00          7.500            122.00
***** End of data *****
```

Payroll Master Output



Mikey's Payroll Report

192.168.15.112:10088/mpavlak/sessions/php101/payrolllist.php

 The PHP Company

Payroll Report

Connected to IBM i

Name	Number	Gross	Tax Rate	Insurance	Net
Gomez Adams	1	\$1500.00	6.500	\$25.00	\$1465.25
Herman Munster	2	\$1200.00	6.500	\$27.00	\$1165.20
Jimmy Buffet	3	\$750.00	3.500	\$53.00	\$694.38
Joe Walsh	4	\$1950.00	7.500	\$122.00	\$1813.38

Total Gross Pay is \$5400.00

Total tax payment is \$34,800.00

Total insurance payment is \$227

The average for each record is....USD 1,350.00

First page of code

```
1 <html><head><title>Mikey's Payroll Report</title>
2 <style type="text/css" media=screen>
3     p {font-family: arial, serif}
4     h1 {font-family: "arial black", serif;color: #FF8C00}</style></head>
5 <body bgcolor='#FFFFFF' text = '#000000'>
6 <h1><img src='zend_logo.gif'>Payroll Report</h1>
7
8 <?php
9
10 //define some variables for database connection
11 $conn = "V7R1GOLD"; //database name of IBM i
12
13 //connect to i5 DB2
14 $db2Connection = db2_connect($conn, "", ""); //Notice no user id or password
15 if ($db2Connection) echo "<p>Connected to IBM i</p>";
16 else {
17     echo "<p>Connection failed: ".db2_stmt_error()." : ".db2_stmt_errormsg()."</p>";
18     exit;
19 }
20 //prepare sql statement
21 $sql = "SELECT * from zenddata.payroll_master";
22
23 //execute sql statement to retrieve the data
24 $resultSet = db2_exec($db2Connection,$sql)
25 or die("<p>Failed query:".db2_stmt_error()." : ".db2_stmt_errormsg()."</p>");
```

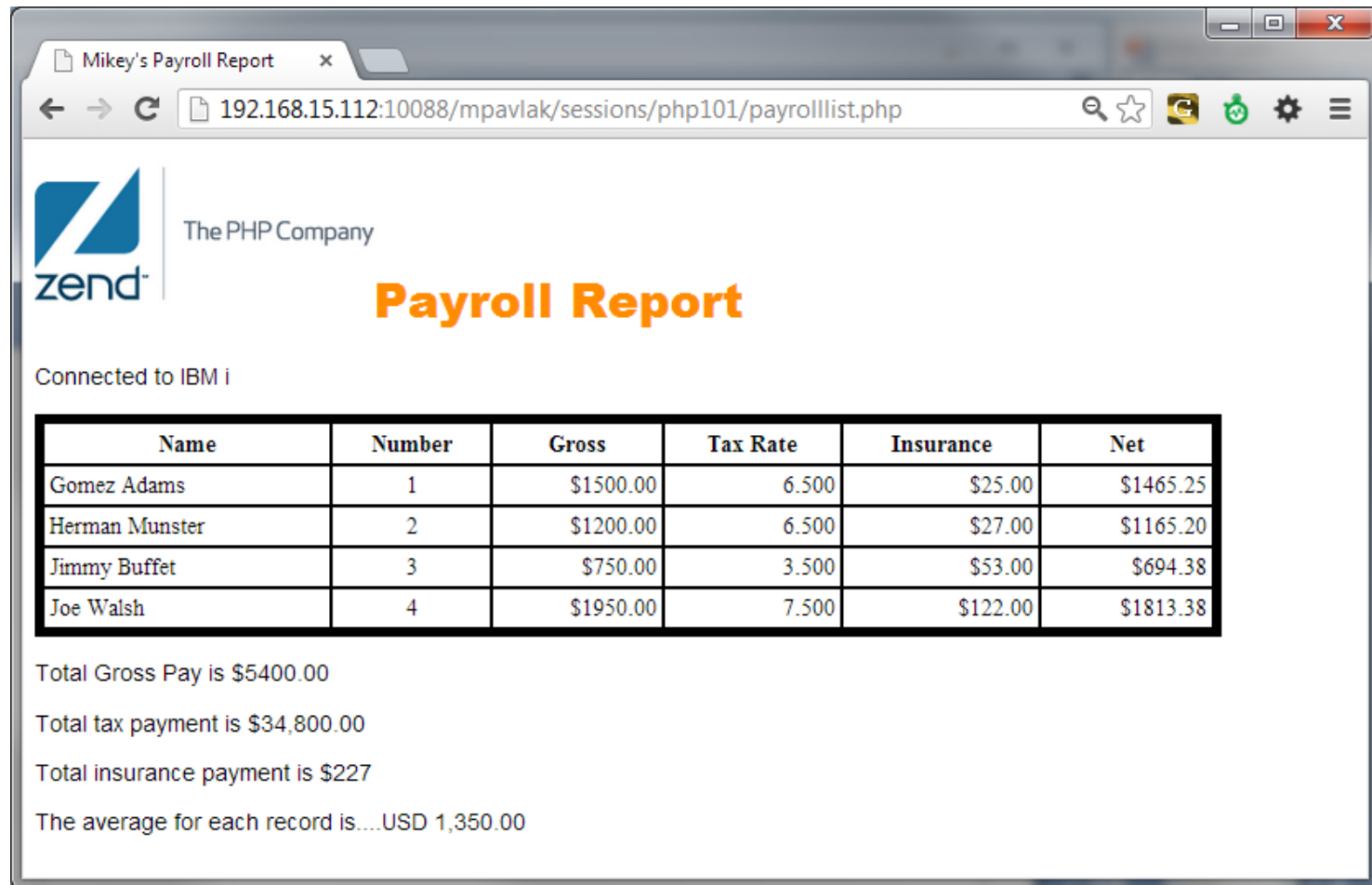
Second page of code

```
28 //begin output table content
29 echo "<table border='6' cellpadding=4 cellspacing=0 bordercolor='#000000' width='90%'>";
30 echo "<th>Name</th> <th>Number</th><th>Gross</th><th>Tax Rate</th><th>Insurance</th><th>Net</th>";
31
32 $Count = 0;
33 //fetch the data from each record and print them out
34 while($row=db2_fetch_array($resultSet)){
35     //retrieve the fields from a row
36     list( $EMP_NAME, $EMP_NUM, $WEEKLY_PAY, $TAX, $INSURANCE)= $row;
37     $Net= $WEEKLY_PAY - ($WEEKLY_PAY * .001 * $TAX) - $INSURANCE;
38     echo("<tr><td>$EMP_NAME</td> <td align='center'>$EMP_NUM</td>
39     <td align='right'>$$WEEKLY_PAY</td>
40     <td align='right'>$TAX</td><td align='right'>$$INSURANCE</td><td align='right'>$");
41
42     printf("%.2f",$Net);          echo("</td></tr>");
43
44     $Total_Gross = $Total_Gross + $WEEKLY_PAY;    //accumulate the total Gross
45     $Total_Net = $Total_Net + $Net;    //accumulate the total Net
46     $Total_Ins = $Total_Ins + $INSURANCE;    //accumulate the insurance
47     $Total_Tax = $Total_Tax + $WEEKLY_PAY * $TAX;    //accumulate the tax
48     ++$Count;    //count the total number of employees
49 }
50 echo "</table>"; //end of table output
```

Last page


```
53 //print out totals
54 echo("<p>Total Gross Pay is $");
55     printf('%.2f',$Total_Gross);
56 echo ("</p>");
57
58 echo("<p>Total tax payment is $");
59 echo number_format($Total_Tax, 2, '.', ',');
60 echo ("</p>");
61
62 echo("<p>Total insurance payment is $$Total_Ins </p>");
63
64 $Average= round($Total_Gross / $Count, 2); // Average Pay Rate
65
66 setlocale(LC_MONETARY, 'en_US');
67 echo("<p>The average for each record is....");
68 echo     money_format('%i',$Average);
69 echo ("</p>");
70
71 db2_close($db2Connection);    //close the database connection
72
73 ?>
74 </body></html>
```

Payroll Master Output



Mikey's Payroll Report

192.168.15.112:10088/mpavlak/sessions/php101/payrolllist.php

 The PHP Company

Payroll Report

Connected to IBM i

Name	Number	Gross	Tax Rate	Insurance	Net
Gomez Adams	1	\$1500.00	6.500	\$25.00	\$1465.25
Herman Munster	2	\$1200.00	6.500	\$27.00	\$1165.20
Jimmy Buffet	3	\$750.00	3.500	\$53.00	\$694.38
Joe Walsh	4	\$1950.00	7.500	\$122.00	\$1813.38

Total Gross Pay is \$5400.00

Total tax payment is \$34,800.00

Total insurance payment is \$227

The average for each record is....USD 1,350.00

PHP101: Db2 Features

Which retrieval method?

- Difference is column names in index
- Benchmarks via Z-Ray on micro-partition

```
Array
(
    [0] => 1221
    [1] => Kauai Dive Shoppe
    [2] => LINA
    [3] => Norman
    [4] => 2
    [5] => 4-976 Sugarloaf Hwy
    [6] => Suite 103
    [7] => Kapaa Kauai
    [8] => HI
    [9] => 94766-1234
    [10] => US
    [11] => ☎ 808-555-0269
    [12] => ☎ 808-555-0278
)
```

DB2_fetch_array()
21 ms

```
Array
(
    [CUST_ID] => 1221
    [COMPANY] => Kauai Dive Shoppe
    [FIRSTNAME] => LINA
    [LASTNAME] => Norman
    [CIVIL] => 2
    [ADDRESS] => 4-976 Sugarloaf Hwy
    [ADDR2] => Suite 103
    [CITY] => Kapaa Kauai
    [STATE] => HI
    [ZIP] => 94766-1234
    [COUNTRY] => US
    [PHONE] => ☎ 808-555-0269
    [FAX] => ☎ 808-555-0278
)
```

DB2_fetch_assoc()
24 ms

```
Array
(
    [CUST_ID] => 1221
    [0] => 1221
    [COMPANY] => Kauai Dive Shoppe
    [1] => Kauai Dive Shoppe
    [FIRSTNAME] => LINA
    [2] => LINA
    [LASTNAME] => Norman
    [3] => Norman
    [CIVIL] => 2
    [4] => 2
    [ADDRESS] => 4-976 Sugarloaf Hwy
    [5] => 4-976 Sugarloaf Hwy
    [ADDR2] => Suite 103
    [6] => Suite 103
    [CITY] => Kapaa Kauai
    [7] => Kapaa Kauai
    [STATE] => HI
    [8] => HI
    [ZIP] => 94766-1234
    [9] => 94766-1234
    [COUNTRY] => US
    [10] => US
    [PHONE] => ☎ 808-555-0269
    [11] => ☎ 808-555-0269
    [FAX] => ☎ 808-555-0278
    [12] => ☎ 808-555-0278
)
```

DB2_fetch_both()
39 ms

DB2 & PHP: options for *LIBL

Option parameter of DB2_Connect

- Options parameter is an array.
- Supports three kinds of library list
 - ▶ i5_lib – Single library
 - ▶ i5_naming – Enable library list via user profile
 - ▶ i5_libl – Specify library list in PHP
- Supports many other parameters

```
resource db2_connect ( string $database , string $username , string $password [, array $options ] )
```

Single library approach

- Similar to the world of MySQL/MariaDB.

```
9 $options = array("i5_lib"=>"ZENDSVR6");
10
11 $con1 = db2_connect($db, $user, $pwd, $options);
12 if (!$con1) {
13     echo 'Error connecting to ' . $db . ' - ' . db2_conn_error() .
14         ' - ' . db2_conn_errormsg();
15     exit();
16 }
17
18 $sql = "select * from sp_cust";
19
20 $result = db2_exec($con1, $sql);
21 if (!$result) {
22     echo 'Error running SQL for ' . $sql . ' - ' . db2_stmt_error() .
23         ' - ' . db2_stmt_errormsg();
24     exit();
25 }
26
27 $firstRow = db2_fetch_assoc($result);
28 print_r($firstRow);
```

```
Array
(
    [CUST_ID] => 1221
    [COMPANY] => Kauai Dive Shoppe
    [FIRSTNAME] => LINA
    [LASTNAME] => Norman
    [CIVIL] => 2
    [ADDRESS] => 4-976 Sugarloaf Hwy
    [ADDR2] => Suite 103
    [CITY] => Kapaa Kauai
    [STATE] => HI
    [ZIP] => 94766-1234
    [COUNTRY] => US
    [PHONE] => ☎ 808-555-0269
    [FAX] => ☎ 808-555-0278
)
```

Multiple libraries hardcoded in PHP

■ More like real life

```
9 $options = array( 'i5_naming' => DB2_I5_NAMING_ON,  
10                  'i5_lib1'=>"MPAVLAK ZENDSVR6 AMAZON");  
11  
12 $con1 = db2_connect($db, $user, $pwd, $options);  
13 if (!$con1) {  
14     echo 'Error connecting to ' . $db . ' - ' . db2_conn_errc  
15         ' - ' . db2_conn_errmsg;  
16     exit();  
17 }  
18  
19  
20 $sql = "select * from sp_cust";  
21  
22 $result = db2_exec($con1, $sql);  
23 if (!$result) {  
24     echo 'Error running SQL for ' . $sql . ' - ' . db2_stmt_  
25         ' - ' . db2_stmt_errmsg;  
26     exit();  
27 }  
28  
29 $firstRow = db2_fetch_assoc($result);  
30 echo '<pre>'; print_r($firstRow); echo '</pre>';
```

Array

```
(  
    [CUST_ID] => 1221  
    [COMPANY] => Kauai Dive Shoppe  
    [FIRSTNAME] => LINA  
    [LASTNAME] => Norman  
    [CIVIL] => 2  
    [ADDRESS] => 4-976 Sugarloaf Hwy  
    [ADDR2] => Suite 103  
    [CITY] => Kapaa Kauai  
    [STATE] => HI  
    [ZIP] => 94766-1234  
    [COUNTRY] => US  
    [PHONE] => ☎ 808-555-0269  
    [FAX] => ☎ 808-555-0278  
)
```

Job Description – PHPUSER2

■ WRKJOB, DSPJOB

```
— Display Job Description                                     System:  I71SALES
Job description:  PHPTEST      Library:  QGPL

Initial library list:

Sequence      Library
number
  10  MPAVLAK
  20  ZENDSVR6
  30  AMAZON

Sequence      Library
number
Sequence      Library
number

Bottom

Press Enter to continue.

F3=Exit  F12=Cancel
```

Multiple libraries via User Profile

■ Parallels Green Screen access

```

8 $user = 'PHPUSER2';
9
10 $options = array('i5_naming' => DB2_I5_NAMING_ON);
11
12 $con1 = db2_connect($db, $user, $pwd, $options);
13 if (!$con1) {
14     echo 'Error connecting to ' . $db . ' - ' . db2_conn_error() . Array
15         ' - ' . db2_conn_errormsg(
16     exit();
17 }
18
19 $sql = "select * from sp_cust";
20
21 $result = db2_exec($con1, $sql);
22 if (!$result) {
23     echo 'Error running SQL for ' . $sql . ' - ' . db2_stmt_error
24         ' - ' . db2_stmt_error
25     exit();
26 }
27
28 $firstRow = db2_fetch_assoc($result);
29 echo '<pre>'; print_r($firstRow); echo '</pre>';

```

```

[CUST_ID] => 1221
[COMPANY] => Kauai Dive Shoppe
[FIRSTNAME] => LINA
[LASTNAME] => Norman
[CIVIL] => 2
[ADDRESS] => 4-976 Sugarloaf Hwy
[ADDR2] => Suite 103
[CITY] => Kapaa Kauai
[STATE] => HI
[ZIP] => 94766-1234
[COUNTRY] => US
[PHONE] => ☎ 808-555-0269
[FAX] => ☎ 808-555-0278
)

```

DB2 & PHP: ibm_db2.ini

php.ini

- File in IFS:
- Zend Server Admin GUI port 10081 or 10091

```

Browse : /usr/local/zendsvr6/etc/conf.d/ibm_db2.ini
Record : _____1_____ of _____3 by _____18
Control : _____

.....+.....1.....+.....2.....+.....3.....+.....4.....+.....5.....+
*****Beginning of data*****
extension=ibm_db2.so
ibm_db2.i5_allow_commit=0
ibm_db2.i5_blank_userid=1
*****End of Data*****
  
```

ibm_db2.i5_allow_commit

This option controls the commit mode used for i5 schema collections in the PHP application.

no commit ▼



ibm_db2	Loaded	1.9.7	IBM DB2 DB Access
Directives <p>These functions enable you to access the IBM DB2 Universal Database, IBM Cloudscape, and Apache Derby databases using the DB2 Call Level Interface (DB2 CLI).</p> <p>ibm_db2.binmode This option controls the mode used for converting to and from binary data in the PHP application. DB2_BINARY ▼</p> <p>ibm_db2.i5_allow_commit This option controls the commit mode used for i5 schema collections in the PHP application. no commit ▼</p> <p>ibm_db2.i5_check_pconnect Advanced db2_pconnect monitor try conn new statement ▼</p>			

Block anonymous

- Samples shipped usually work
- Default setting is “ON”
- Default profile is QTMHHTTP
- Change requires restart of Zend Server PHP

Subsystem/Job	Current User
QHTTPSVR	QSYS
ZENDSVR6	QTMHHTTP
ZENDSVR6	QTMHHTTP
ZENDSVR6	QTMHHTTP
ZENDSVR6	QTMHHTTP
ZENDSVR6	QTMHHTTP
ZENDSVR6	QTMHHTTP
ZENDSVR6	QTMHHTTP
ZENDSVR6	QTMHHTTP



ibm_db2	Loaded	1.9.7	IBM DB2 DB Access
---------	--------	-------	-------------------

ibm_db2.i5_blank_userid
Allow blank userid/password

ON

Connection failed. SQL Err:08001
Authorization failure on distributed database connection attempt. SQLCODE=-30082

PHP101: Program Call

Call RPG Pricing Program

- Any IBM i program object will do
- Create toolkit object
- Setup parameters
- Call program
- Display results

```
C      *ENTRY          PLIST
C                      PARM          Unit_Price          9 2
C                      PARM          Qty                 9 2
C                      PARM          Total_Price         11 2

/Free
  Total_Price = Qty * Unit_Price;
  If (Total_Price > 100000);
    Total_Price = Total_Price * .9;
  Else;
    Total_Price = Total_Price * .95;
  EndIf;

  *InLR = *On;
```

HTML Form to get Parm

```
1 <html><head>
2 <title>Call RPG Pricing Routine</title>
3 <style type="text/css" media=screen>
4   h1 {font-family: "arial black", serif;color: #FF8C00}
5   th {text-align:center; background-color:pink;}
6   input[type="text"] {text-align: right; background-color:lime;}
7 </style></head><body>
8
9 <h1>Calling an RPG program with two parameters</h1>
10 <form method="post" action="CustomerPrice.php">
11
12 <table border="1">
13 <tr><th>Price</th><th>Quantity</th></tr>
14 <tr><td><input type="text" name="Price" value="0.00"></td>
15 <td><input type="text" name="Quantity" value="0"></td></tr>
16
17 <tr><td colspan="2" align="right"><input type="reset">
18 <input type="submit" value="Call program"></td>
19 </tr></table></form></body></html>
```

Input form, default zeros then enter data

Call RPG Pricing Routine x

192.168.15.112:10088/mpavlak/sessions/php101/programcall.html

Calling an RPG program with two parameters

Price	Quantity
0.00	0

Reset Call program

Call RPG Pricing Routine x

192.168.15.112:10088/mpavlak/sessions/php101/programcall.html

Calling an RPG program with two parameters

Price	Quantity
25.50	600

Reset Call program

Program call Part 1

```
1 <html> <head><title>Call RPG Pricing Routine</title>
2 <style type="text/css" media=screen>
3     h1 {font-family: "arial black", serif;color: #FF8C00}
4     th {text-align:center; background-color:pink;}
5     input[type="text"] {text-align: right; background-color:lime;}
6 </style></head><body>
7 <h1>Results from RPG Pricing program call with two parameters</h1>
```

Program Call part 2 - Functions

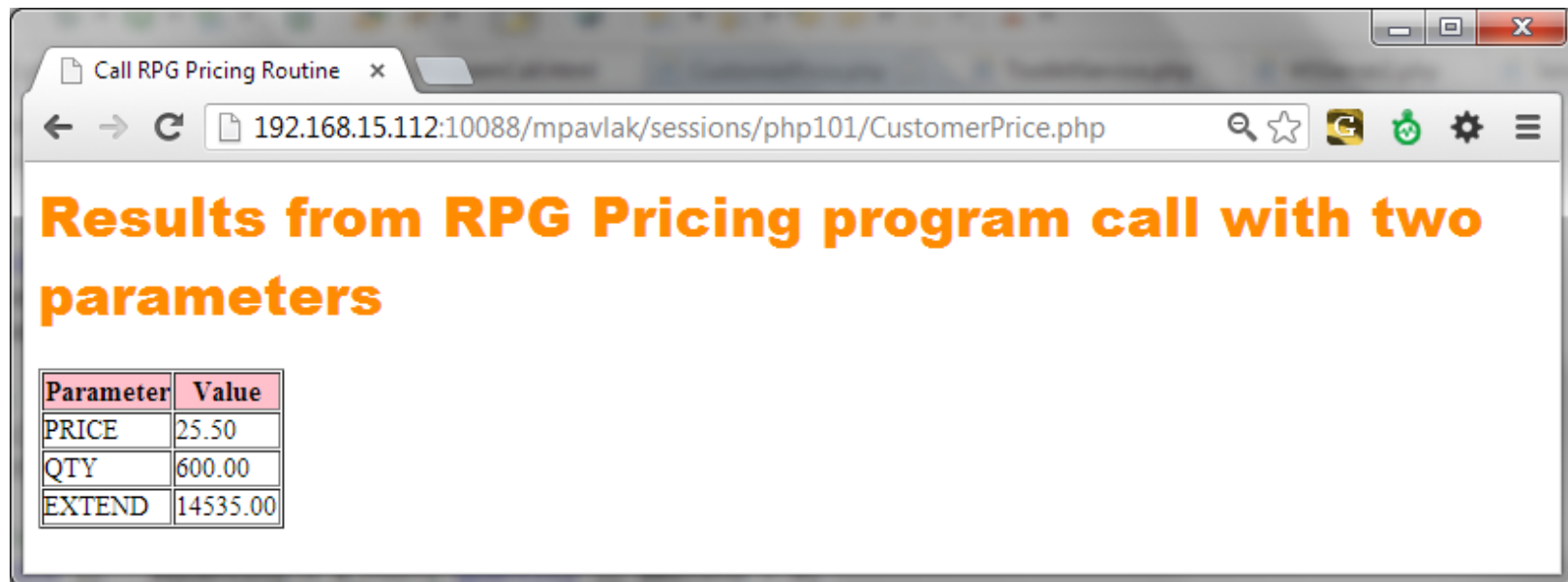
```
11 function getSession() {
12
13     //Check session state, if active, return ID else create session...
14     if (session_status==PHP_SESSION_ACTIVE)
15         $sess = session_id();
16     else {
17         session_start();
18         $sess = session_id();
19     }
20     return $sess;
21 }
22
23 function displayResult($toolkitArray) {
24
25     //format response from program call using standard routine...
26     echo '<table border="1"><tr><th>Parameter</th><th>Value</th></tr>';
27     $io = $toolkitArray['io_param'];
28     foreach ($io as $parm=>$value) {
29         echo "<tr><td>$parm</td><td>$value</td></tr>";
30     }
31     echo '</table>';
32 }
```


Program Call – Part 3

```
11+ function getSession() {}
22+ function displayResult($toolkitArray) {}
32
33 //Setup parms and create the toolkit object...
34 include_once 'ToolkitService.php';
35 $db='V7R1GOLD'; $user=''; $pwd=''; $extension='ibm_db2';
36 $toolObj = ToolkitService::getInstance($db, $user, $pwd, $extension);
37
38 //Establish session info and setup internal key...
39 $sess = getSession();
40 $toolObj->setToolkitServiceParams(array('InternalKey'=>"/tmp/$sess"));
41
42 //Load parameters for program call...
43 $price = $_POST['Price']; $quantity = $_POST['Quantity']; $extend = 0;
44
45 //Setup parameter array...
46 $param[] = $toolObj->AddParameterPackDec('both', 9, 2, 'PRICE', 'PRICE', $price);
47 $param[] = $toolObj->AddParameterPackDec('both', 9, 2, 'QTY', 'QTY', $quantity);
48 $param[] = $toolObj->AddParameterPackDec('both', 11, 2, 'EXTEND', 'EXTEND', $extend);
49
50 //Call program...
51 $result = $toolObj->PgmCall("CUSTPRICE", "ZENDDATA", $param, null, null);
52
53 //Interrogate results...
54 if($result){
55     displayResult($result);
56 }
57 else echo "Execution failed.";
58
59 /* Do not use the disconnect() function for "state full" connection */
60 $toolObj->disconnect();    ?> </body></html>
```

Output of the Program Call

- See input and output values from toolkit
- Yes, values can be changed in RPG and routed back
 - ▶ Essentially, that is what happened to EXTEND...



The screenshot shows a web browser window with the title 'Call RPG Pricing Routine'. The address bar displays the URL '192.168.15.112:10088/mpavlak/sessions/php101/CustomerPrice.php'. The main content area features a heading 'Results from RPG Pricing program call with two parameters' in orange text. Below the heading is a table with two columns: 'Parameter' and 'Value'.

Parameter	Value
PRICE	25.50
QTY	600.00
EXTEND	14535.00

Where to get more

- HTML training
 - ▶ W3schools.com
 - ▶ Local college, continuing education, etc.
- PHP training
 - ▶ Anywhere will work
 - ▶ Zend, iProDeveloper (formerly SystemiNetwork)
- Get a project
 - ▶ Define a Proof-of-concept (POC)
 - ▶ Dig up something out of the backlog

End-to-End Application Modernization Solutions

GUI, Web & Mobile



Analysis & Productivity



Reporting & Document Distribution



Code & Database Modernization



IT Strategy Services



Application Services & Staffing



THANK YOU

Mike Pavlak – IT Strategist, Fresche Solutions

Mike.Pavlak@freschesolutions.com



FRESCHÉ
SOLUTIONS