PharmaSUG 2014 – Paper AD14

Add Dropdowns and Redirects to Your SAS® Output

Barbara Ross, Advance America, Spartanburg, SC

ABSTRACT

Outputting attractive and interactive reports via SAS onto the web doesn't have to be complicated. For most internal purposes, a webpage directory that links to static reports that update regularly is enough. This paper introduces simple ways to add dropdown menus and redirects within your SAS output. All webpages are created using Base SAS and website directories are managed using an open source FTP client for Windows (WinSCP in this case). Techniques used include: JavaScript to select item on click, PHP to collect form entries, and HTML for hyperlinks. The SQL procedure and DATA step are used for data transformations.

INTRODUCTION

Three techniques for redirecting your users will be introduced in this paper: 1) Hyperlinks 2) Dropdowns with On Click Redirect and 3) Form Entry. They are ordered in increasing complexity, however, none of the techniques introduced require extensive coding and are easy to implement. The first technique will focus on HTML code within your SAS program, while technique two will incorporate JavaScript. Lastly, the third technique will utilize PHP and HTML to collect data from forms. All the programming used in this paper is done within the SAS 9.3 software.

This paper assumes you already have a website set up and are able to upload to that site via SAS. If you need examples of how to set up your webpage and uploading, you can reference http://bharlan.weebly.com/upload-reports-to-website.html for more details.

BASIC DIFFERENCES BETWEEN HTML, JAVASCRIPT, and PHP

To help sort the different languages used throughout this paper, I have compiled some basic definitions with examples for your reference. It is important to note how the statements "open and close", this is how you will be able to distinguish between the languages within the examples.

HTML is used to design basic webpages. It can define layouts, colors, tables, and more. In this paper it will be used to create dropdowns, hyperlinks, and alter fonts.

- HTML is recognizable by its opening and closing tags which use angle brackets to contain keywords/tag names that define the content between the tags.
- The closing tag is written a forward slash before the tag name.
- Everything between the open and closing tags is referred to as an element.

```
<font  size=5> Welcome to my Homepage </font>
```

Output 1. HTML Example

JavaScript is a scripting language that provides a level of interactivity to web pages that is not achievable with simple HTML. JavaScript is not to be confused with Java however. Java is a programming language that allows the user to create applications while JavaScript is a scripting language that is run on the browser only. JavaScript resides embedded inside HTML documents. We will use JavaScript to do on click redirects with drop down menus.

- The <script> and </script> HTML tags tell where the JavaScript starts and ends. The lines between them contain the JavaScript code.
- Semicolons separate JavaScript statements. If only one statement, the closing tag will imply a semicolon.

```
<h1 id="myH1"></h1> <script> document.getElementById("myH1").innerHTML="Welcome to my
                                Homepage"; </script>
```

Output 2. JavaScript embedded in HTML Example

PHP provides a way for you to put instructions in your HTML files to create dynamic content. PHP can be embedded directly into an HTML document, however when it is, you need to use the .php extension for that file so the web server knows to send the file for PHP processing.

- <?php and ?> indicates where the PHP starts and ends.
- Semicolons separate PHP statement. If only one statement, the closing tag will imply a semicolon.

```
<TITLE><?php echo ”Welcome to my Homepage”; ?></TITLE>
```

Output 3. PHP embedded in HTML Example

DEFINITIONS

- **Hyperlink**: A link from a webpage or document to another location or file, typically activated by clicking on the highlighted word or image on the screen
- **HTML Tag**: Names/keywords surrounded by angle brackets like <html>; Usually comes in pairs like <b> and </b>; An HTML element starts with an opening tag and ends with a closing tag
- **HTML Element**: Refers to everything from the start to end tag; Can be nested within other elements; Usage: *The <p> element defines the paragraph in a HTML document*
- **HTML Attribute**: Provide additional information about an element; always a part of the opening tag; comes in name/value pairs like *name=”value”*
- **Drilldown**: Move from summary information to detailed data focusing in on something
- **On Click Redirect**: A redirect that occurs when the user selects an item from list; Alternative to form entry as it does not require the user to press a submit button; (*on click redirect can refer to simpler redirects, but for the purpose of this paper we will use it in this sense)*

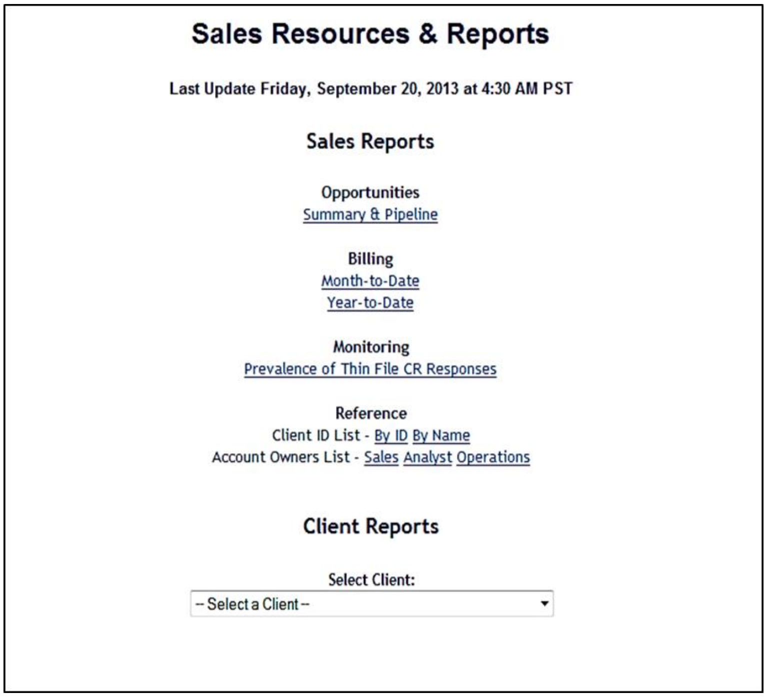
TECHNIQUE 1: HYPERLINKS

Hyperlinking your text is done by placing the text within the HTML hyperlink tags, <a> and </a>. The most important attribute of the <a> element is the href attribute which dictates the link’s destination. To fancy it up a bit, you can also add a title attribute to display text while the user’s mouse hovers over the link.

SYNTAX for Hyperlinking Text:

```
<a href=destination_address title=mouseover_text>Hyperlinked Text</a>
```

You can create directory pages entirely using hyperlinks. I find this is useful when the user may not realize all report options available to them. Figure 1 illustrates a directory page created with hyperlinks using the ODS TEXT statement.



Sales Resources & Reports

Last Update Friday, September 20, 2013 at 4:30 AM PST

Sales Reports

Opportunities
Summary & Pipeline

Billing
Month-to-Date
Year-to-Date

Monitoring
Prevalence of Thin File CR Responses

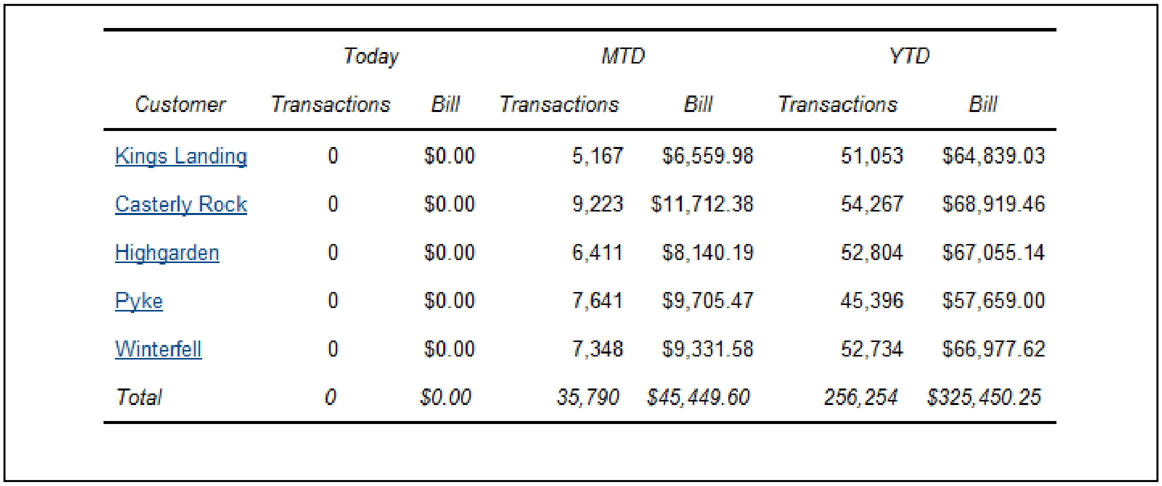
Reference
Client ID List - By ID By Name
Account Owners List - Sales Analyst Operations

Client Reports

Select Client:
-- Select a Client --

Figure 1. Directory Page using Hyperlinks and a Dropdown Menu

The more useful and often overlooked use of hyperlinks, however, is to use them within your reports and tables for drilling down. Take for example a report that tracks customer volume. The user will first reference the overall or summary report to see if anything is alarming. If they see one customer has dropped volume, they can then click on the customer's name to bring them to a detailed report of that customer's transactions. This prevents the user from having to go back to the directory page to get to the report needed.



	Today		*MTD*		*YTD*	
Customer	*Transactions*	*Bill*	*Transactions*	*Bill*	*Transactions*	*Bill*
Kings Landing	0	$0.00	5,167	$6,559.98	51,053	$64,839.03
Casterly Rock	0	$0.00	9,223	$11,712.38	54,267	$68,919.46
Highgarden	0	$0.00	6,411	$8,140.19	52,804	$67,055.14
Pyke	0	$0.00	7,641	$9,705.47	45,396	$57,659.00
Winterfell	0	$0.00	7,348	$9,331.58	52,734	$66,977.62
Total	*0*	*$0.00*	*35,790*	*$45,449.60*	*256,254*	*$325,450.25*

Figure 2. Using Hyperlinks as a Drilldown for Reporting

The trick to efficiently hyperlinking in your code is to name your reports with values within your data. A dataset of customer volumes will include a field for Customer_Name and Cust_ID. When you create that customer's report, use these values in the name of the HTML file. In the text, below the name of the file includes a macro variable of the customer ID.

```
ODS HTML FILE = "reportfor_&CustID..html";
```

This works well when looping through customers to make individual reports. That is outside the scope of this paper, but you can reference http://bharlan.weebly.com/loop-to-make-report-for-each-customer.html for more information.

Now instead of just having the summary table list the Customer_Name, create a new variable called CustLink that will be the Customer_Name only hyperlinked to that customer's specific report.

```
CustLink = CAT("<a href=http://website.com/reportfor_",Cust_ID,
".html>",TRIM(Customer_Name),"</a>");
```

TECHNIQUE 2: DROPDOWNS WITH ON CLICK REDIRECT

While it's simple enough just to have a hyperlinked list as your directory, it is much more attractive to use dropdown menus - especially if you have a long list.

To make a drop down menu:
1) Add HTML tags to list items using the CAT function (similar to earlier)
2) List the items into a macro variable using PROC SQL
3) Add the HTML dropdown code into your SAS program via an ODS TEXT statement
4) Add the JavaScript function code (to allow the on click functionality) to your ODS TEXT statement

STEP 1 - Add HTML Tags to List Items

Steps 1 and 2 are convenience steps that can be skipped if you prefer to manually enter the values of the dropdown in your ODS TEXT statement. If it's a long or dynamic list though, you will want to use this method to save time.

SYNTAX for Dropdown List Item:

```
<option value=report_web_address>List Item</option>
```

The <option> element is used to define options or items to be included in the dropdown. The value attribute is specifying the value of the list item- in this case the web address of that report.

The code below simplifies the dataset to have one record for each customer using the SORT Procedure. The DATA step then encloses the customer name in <option> tags using the CAT function.

```
*Keep one entry per customer;
PROC SORT DATA=save.dataset OUT=save.dataset2 NODUPKEY; BY cust_id;
RUN;
*Add option tags to each list item;
DATA save.dataset2; SET save.dataset2;
drop = CAT ("<option
value=http://website.com/reportfor_",cust_id,".html>",
TRIM(customer_name)," Detail</option>");
RUN;
```

STEP 2 – List the Items into a Macro Variable

```
*List drop down entries into macro variable;
PROC SQL;
SELECT drop INTO :droplist SEPARATED BY ' ' FROM save.dataset2;
QUIT;
```

STEP 3 – Add HTML Dropdown Code

The <select> tag defines a dropdown list in HTML. Within it, we will include the option statements created in Step 1 that are referenced by the &droplist macro variable created in Step 2. This example also includes some additional option statements as well as a valueless option statement which is used as a list separator. This code will be place within the ODS TEXT statement with the JavaScript code in Step 4.

```
<div align='center'>
 <body>
  <p> <b> Show Report: </b><br>
   <select name='navigation'>
     <option value=http://website.com/index.html>Current Summary</option>
     <option>-------</option>
     <option value=http://website.com/index.html>Current Summary</option>
     <option value=http://website.com/summary2012.html>Summary 2012</option>
     <option value http://website.com/summary2011.html>Summary 2011</option>
    <option style='background-color:666666;color:white'> Customers</option>
    &droplist
   </p>
  </body>
 </div>
```

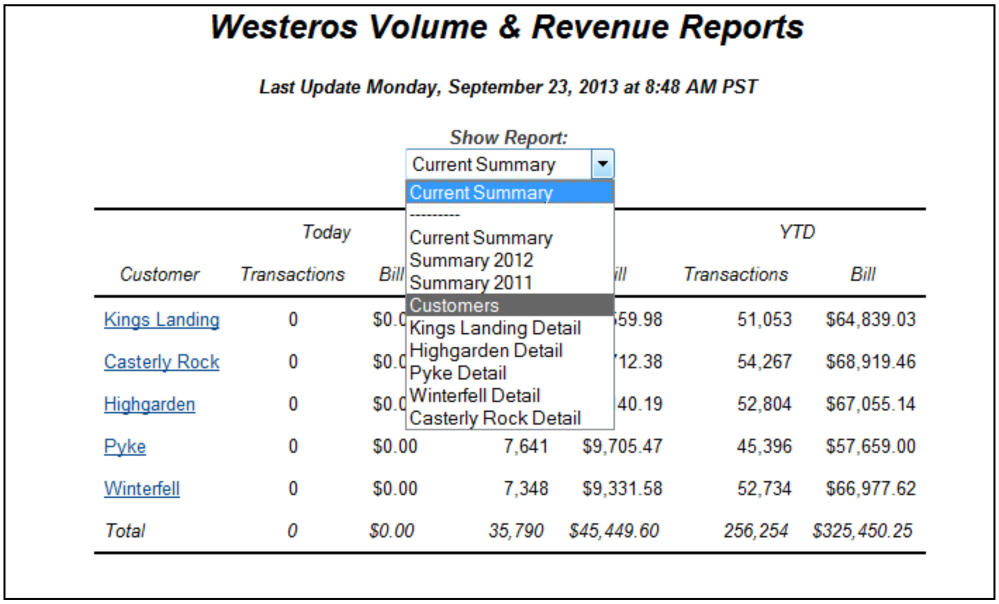
STEP 4 – Add JavaScript Function

The benefit of using a JavaScript function is you can redirect a dropdown without having to place it in a form, or have the user press a button. The user only needs to change the value. I prefer to place this code before my dropdown menu, but it should work placed anywhere. No changes need to be made to the JavaScript function below.

```
ODS TEXT=
"
<script 'text/javascript'>

window.onload = function () {
var nav = document.getElementsByTagName('select')[0];
nav.onchange = function () {
var idx = this.selectedIndex;
window.location.href = this.options[idx].value;
}
}
</script>
…Step 3 code here…..
“;
```

TIPS for Drop-Down Menus

- If the dropdown menu is on a menu page, you can start the list with a blank value such as *<option>--Select--</option>* as to not confuse the user
- If the dropdown menu will reside in the page header, have the first option display the page the user is currently on (Technique 3 uses this in the example)
- SAS output may display incorrectly after creating a dropdown menu with ODS TEXT. If you would like the drop down to remain above you output, place it with the TITLE statement.



Westeros Volume & Revenue Reports

Last Update Monday, September 23, 2013 at 8:48 AM PST

Show Report:

	Today				*YTD*	
Customer	*Transactions*	*Bill*		*ill*	*Transactions*	*Bill*
Kings Landing	0	$0.0		59.98	51,053	$64,839.03
Casterly Rock	0	$0.0		12.38	54,267	$68,919.46
Highgarden	0	$0.0		40.19	52,804	$67,055.14
Pyke	0	$0.00	7,641	$9,705.47	45,396	$57,659.00
Winterfell	0	$0.00	7,348	$9,331.58	52,734	$66,977.62
Total	*0*	*$0.00*	*35,790*	*$45,449.60*	*256,254*	*$325,450.25*

Figure 3. Dropdown Menu within Header Using JavaScript to Redirect On Click

TECHNIQUE 3: FORM ENTRY

Form Entry should be used when there are too many reports for just one dropdown. It is also useful when the user needs to select multiple criteria to get to the correct report.

To create a form:

1) Prepare your dropdown lists values; Can be done manually or with the concatenation method used in Technique 2 Steps 1 & 2
2) Add the <select> element around each dropdown list
3) Add the <form> element around your collection of dropdowns; Before closing the element add the *<input 'submit' />* statement for the submit button
4) Create a PHP file that processes the information from the form- in this case redirecting the user to the correct report

STEPS 1,2,3 – Creating the form

Basic structure of a form in HTML is the <form> element enclosing all the dropdown lists and ending with an <input /> tag. The code below gives the beginnings of the form shown in Figure 4.

```
<form action='form.php' method='get'> *The name of your PHP file created
in step 4;
   Report: <select name='rep'> *'rep' is the variable name;
   <option value=overall >Overall</option> *The report you are on first so
```

```
there's no confusion!;
   <option>---------</option>
   <option value='overall'>Overall</option> *The value of the variable,
ex: rep=overall;
   <option value='services'>Services</option>
   <option value='top10'>Top 10</option>
   </select>&nbsp

   Display: <select name='disp'>
   <option value=table >Table</option>
   <option>---------</option>
   <option value='graph'>Graph</option>
   <option value='table'>Table</option>
   </select>&nbsp
```

****_keep doing that for all fields_****

```
<input type='submit' value='Submit' >
</form>
```

Notice the value attribute within the <option> tag no longer includes the webpage but rather the value of the variable used in naming your HTML output. In Technique 2, we created our dropdown values using the following concatenation:

```
Drop = CAT ("<option value=http://website.com/reportfor_",
cust_id,".html>",TRIM(customer_name)," Detail</option>");
```

A similar option statement for form entry would have the following syntax:

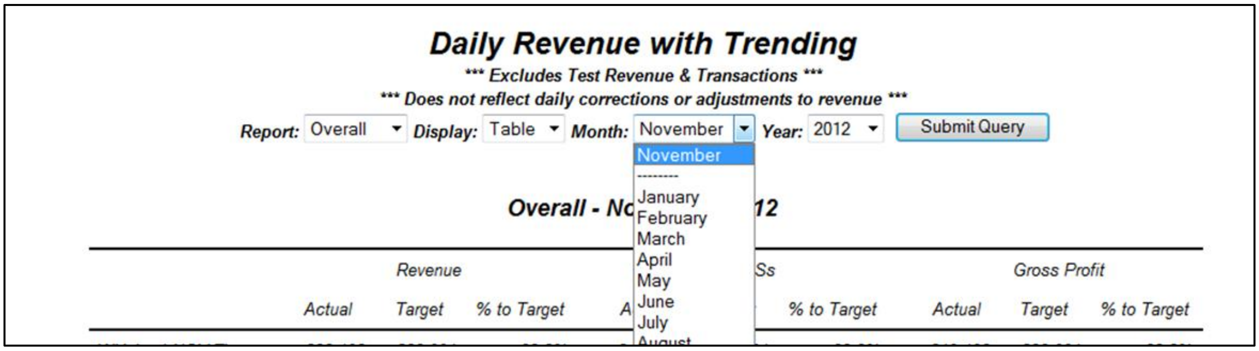
```
<select name='cust_id'>
   <option value=value of cust_id >customer_name</option>
```

STEP 4 – Creating the PHP file

The PHP file used to process the form can be done in any program as long it is saved with the .php extension. I create mine in Notepad. You will upload this file to the same directory as your reports. The short PHP code shown below first gathers the values from the form and assigns them to a variable. Next it redirects the user to a page using the variable's values.

```
<?php
$rep = $_GET["rep"];        // value of the name attribute within the <select> element
$disp = $_GET["disp"];      // variable that contains the value of disp
$yr = $_GET["yr"];
$mo = $_GET["mo"];
?>
<meta http-equiv="refresh" content="0; url=<?php echo $_GET["rep"]; ?>_<?php echo $_GET["disp"];
?>_<?php echo $_GET["yr"]; ?>_<?php echo $_GET["mo"]; ?>.html">
```

Echo is to PHP as the PUT statement is to SAS. Here the <meta> HTML element is redirecting the user. The PHP code within is supplying the name of page it needs to go to, specifically the values of *rep_disp_yr_mo*.html.



Daily Revenue with Trending

*** Excludes Test Revenue & Transactions ***

*** Does not reflect daily corrections or adjustments to revenue ***

Report: Overall Display: Table Month: November Year: 2012

Figure 4. Form within Header using PHP to Redirect on Multiple Criteria

CONCLUSION

Outputting your SAS reports to an internal website facilitates the transfer of information within your company. However without easy navigation, it can quickly become "too complicated" for your end user to use these tools, defeating the purpose of the website. By utilizing web languages such as HTML, JavaScript, and PHP within your SAS programs, you can give your reports the ease users are accustomed too while browsing the web.

Several examples were used within this paper in order to illustrate different uses of these techniques. However to illustrate how all the techniques can be used together, I have placed a full SAS program within the Appendix and on the web at http://bharlan.weebly.com/my-papers.html.

REFERENCES

http://www.w3schools.com/
http://www.java.com/en/download/faq/java_javascript.xml
http://oreilly.com/catalog/phppr/chapter/php_pkt.html
http://bharlan.weebly.com/

RECOMMENDED READING

http://www.w3schools.com/

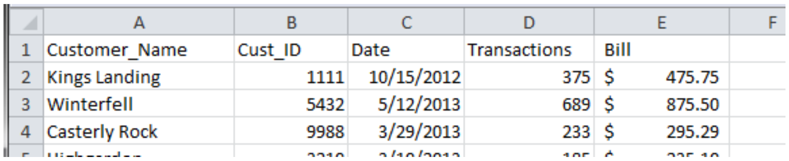
CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Name: Barbara Ross
Enterprise: Advance America
City, State ZIP: Spartanburg, SC 29306
E-mail: bmharlan@gmail.com
Web: bharlan.weebly.com

APPENDIX

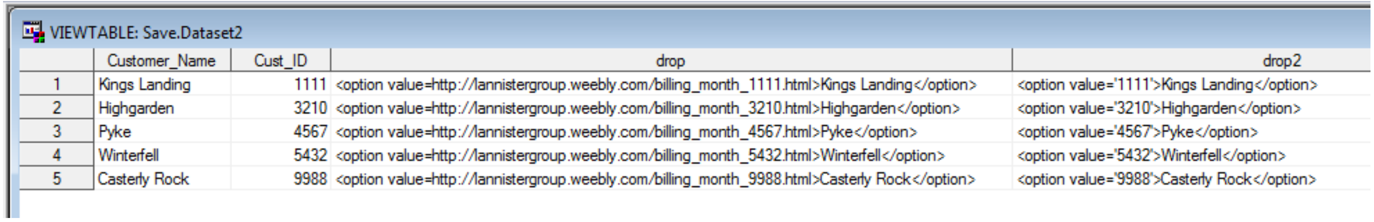
SAVE.DATASET



	A	B	C	D	E	F
1	Customer_Name	Cust_ID	Date	Transactions	Bill	
2	Kings Landing	1111	10/15/2012	375	$ 475.75	
3	Winterfell	5432	5/12/2013	689	$ 875.50	
4	Casterly Rock	9988	3/29/2013	233	$ 295.29	

```
/*************************************************
PROGRAM: WUSS Paper

ABOUT: Example website for Adding Dropdowns and Redirects to your SAS Output

CREATED BY BARBARA HARLAN ROSS
*************************************************/

LIBNAME save "C:\Users\barbara.harlan\Google Drive\SUGs\Redirects paper";

%LET site=http://lannistergroup.weebly.com;
%LET path=C:\Users\barbara.harlan\Google Drive\SUGs\Redirects paper;

*Header Info;
%LET logo=http://img694.imageshack.us/img694/3883/93h.png;
%LET current_date=%sysfunc(today(),weekdate.);
%LET current_time=%sysfunc(time(),timeampm7.);


/********** DROP DOWN***************/
*Keep one entry per customer;
PROC SORT DATA=save.dataset OUT=save.dataset2 NODUPKEY; BY cust_id; RUN;
*Add option tags to each list item;
DATA save.dataset2; SET save.dataset2;
*dropdown for onclick redirect;
drop = CAT ("<option value=&site/billing_month_",cust_id,".html>",TRIM(customer_name),"</option>");
*dropdown for form entry;
drop2 = CAT ("<option value='",cust_id,"'>",TRIM(customer_name),"</option>");
KEEP customer_name cust_id drop drop2;
RUN;
*List drop down entries into macro variable;
PROC SQL NOPRINT;
SELECT drop INTO :droplist SEPARATED BY ' ' FROM save.dataset2;
SELECT drop2 INTO :droplist2 SEPARATED BY ' ' FROM save.dataset2;
QUIT;
```

DATASET2

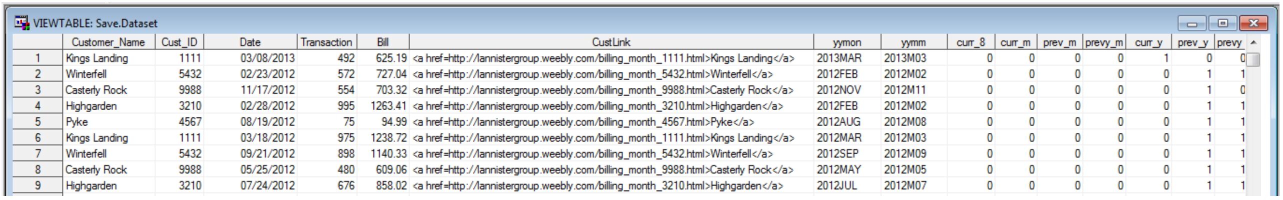


	Customer_Name	Cust_ID	drop	drop2
1	Kings Landing	1111	<option value=http://lannistergroup.weebly.com/billing_month_1111.html>Kings Landing</option>	<option value='1111'>Kings Landing</option>
2	Highgarden	3210	<option value=http://lannistergroup.weebly.com/billing_month_3210.html>Highgarden</option>	<option value='3210'>Highgarden</option>
3	Pyke	4567	<option value=http://lannistergroup.weebly.com/billing_month_4567.html>Pyke</option>	<option value='4567'>Pyke</option>
4	Winterfell	5432	<option value=http://lannistergroup.weebly.com/billing_month_5432.html>Winterfell</option>	<option value='5432'>Winterfell</option>
5	Casterly Rock	9988	<option value=http://lannistergroup.weebly.com/billing_month_9988.html>Casterly Rock</option>	<option value='9988'>Casterly Rock</option>

```
/*************PREP DATASET FOR SUMMARY REPORTS*************/
DATA save.dataset; SET save.dataset;
*hyperlink;
CustLink = CAT("<a href=&site/billing_month_",cust_id,".html>",TRIM(Customer_Name),"</a>");
*Month;
yymon = PUT(date,yymon.);
yymm = PUT(date,yymm.);
*Indicators;
IF date >= today()-8 THEN curr_8=1; ELSE curr_8=0; *Last 8 days;
IF date >= intnx('month',today()-1,0,'beginning') THEN curr_m=1; ELSE curr_m=0; *Current month;
IF intnx('month',today()-1,-1,'beginning') <= date <= intnx('month',today()-1,-1,'end') THEN
prev_m=1; ELSE prev_m=0; *Previous month TOTAL;
```

```
IF intnx('month',today()-1,-12,'beginning') <= date <= intnx('year',today()-1,-1,'same') THEN
prevy_m=1; ELSE prevy_m=0; *prev year's MTD;
IF year(date)=year(today()-1) THEN curr_y=1; ELSE curr_y=0; *This year;
IF year(date)=year(intnx('year',today()-1,-1)) THEN prev_y=1; ELSE prev_y=0; *Last year TOTAL;
IF intnx('year',today()-1,-1,'beginning') <= date <= intnx('year',today()-1,-1,'same') THEN
prevy_y=1; ELSE prevy_y=0; *prev year's YTD;
RUN;
```

SAVE.DATASET

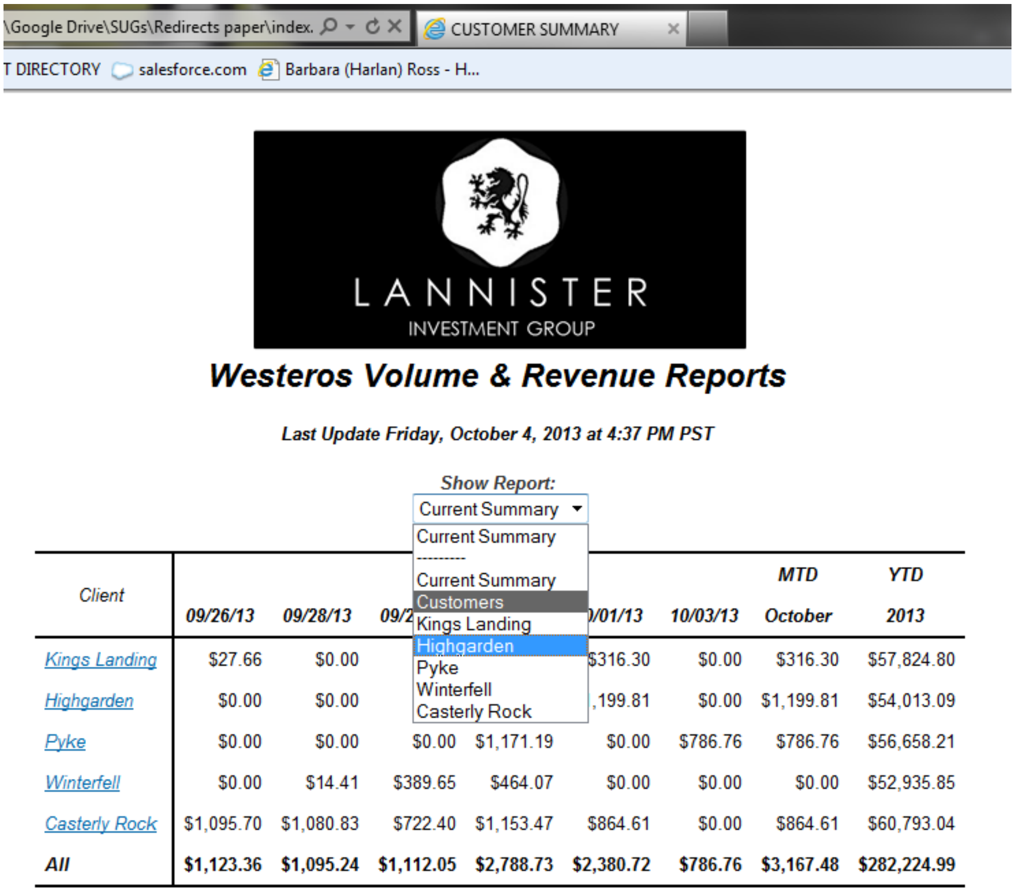


VIEWTABLE: Save.Dataset

	Customer_Name	Cust_ID	Date	Transaction	Bill	CustLink	yymon	yymm	curr_8	curr_m	prev_m	prevy_m	curr_y	prev_y	prevy
1	Kings Landing	1111	03/08/2013	492	625.19	<a href=http://lannistergroup.weebly.com/billing_month_1111.html>Kings Landing</a>	2013MAR	2013M03	0	0	0	0	1	0	0
2	Winterfell	5432	02/23/2012	572	727.04	<a href=http://lannistergroup.weebly.com/billing_month_5432.html>Winterfell</a>	2012FEB	2012M02	0	0	0	0	0	1	1
3	Casterly Rock	9988	11/17/2012	554	703.32	<a href=http://lannistergroup.weebly.com/billing_month_9988.html>Casterly Rock</a>	2012NOV	2012M11	0	0	0	0	0	1	0
4	Highgarden	3210	02/28/2012	995	1263.41	<a href=http://lannistergroup.weebly.com/billing_month_3210.html>Highgarden</a>	2012FEB	2012M02	0	0	0	0	0	1	1
5	Pyke	4567	08/19/2012	75	94.99	<a href=http://lannistergroup.weebly.com/billing_month_4567.html>Pyke</a>	2012AUG	2012M08	0	0	0	0	0	1	1
6	Kings Landing	1111	03/18/2012	975	1238.72	<a href=http://lannistergroup.weebly.com/billing_month_1111.html>Kings Landing</a>	2012MAR	2012M03	0	0	0	0	0	1	1
7	Winterfell	5432	09/21/2012	898	1140.33	<a href=http://lannistergroup.weebly.com/billing_month_5432.html>Winterfell</a>	2012SEP	2012M09	0	0	0	0	0	1	1
8	Casterly Rock	9988	05/25/2012	480	609.06	<a href=http://lannistergroup.weebly.com/billing_month_9988.html>Casterly Rock</a>	2012MAY	2012M05	0	0	0	0	0	1	1
9	Highgarden	3210	07/24/2012	676	858.02	<a href=http://lannistergroup.weebly.com/billing_month_3210.html>Highgarden</a>	2012JUL	2012M07	0	0	0	0	0	1	1

```
*Choose to summarize outside of the reporting procedures so the report code isn't as long;
*monthly table summary;
PROC SQL;
CREATE TABLE mc1 AS SELECT cust_id, custlink, 'm' as ds, ' MTD        ' as cat, PUT(today()-
1,worddate9.) as date, sum(Bill) as Bill, sum(transactions) as Txs FROM save.dataset WHERE
curr_m=1 GROUP BY cust_id, custlink;
CREATE TABLE mc2 AS SELECT cust_id, custlink, 's' as ds, ' Last 8 Days' as cat,
PUT(date,mmddyy8.) as date, sum(Bill) as Bill, sum(transactions) as Txs FROM save.dataset WHERE
curr_8=1 GROUP BY cust_id, custlink, date;
CREATE TABLE mc3 AS SELECT cust_id, custlink, 'm' as ds, 'Prev Month  ' as cat,
PUT(intnx('month',today()-1,-1),worddate9.) as date, sum(Bill) as Bill, sum(transactions) as Txs
FROM save.dataset WHERE prev_m=1 GROUP BY cust_id, custlink;
CREATE TABLE mc4 AS SELECT cust_id, custlink, 'm' as ds, ' Prev Yr MTD' as cat,
PUT(intnx('month',today()-1,0),worddate9.) as date, sum(Bill) as Bill, sum(transactions) as Txs
FROM save.dataset WHERE prevy_m=1 GROUP BY cust_id, custlink;
CREATE TABLE mc5 AS SELECT cust_id, custlink, 'm' as ds, '  This Month' as cat,
PUT(date,mmddyy8.) as date, sum(Bill) as Bill, sum(transactions) as Txs FROM save.dataset WHERE
curr_m=1 GROUP BY cust_id, custlink, date;
QUIT;
*Yearly table summary;
PROC SQL;
CREATE TABLE yc1 AS SELECT cust_id, custlink, 'y' as ds, ' YTD        ' as cat, PUT(today(),year4.)
as date, sum(Bill) as Bill, sum(transactions) as Txs FROM save.dataset WHERE curr_y=1 GROUP BY
cust_id, custlink;
CREATE TABLE yc2 AS SELECT cust_id, custlink, 'y' as ds, ' Monthly  ' as cat, yymm as date,
sum(Bill) as Bill, sum(transactions) as Txs FROM save.dataset WHERE curr_y=1 GROUP BY cust_id,
custlink, yymm;
CREATE TABLE yc3 AS SELECT cust_id, custlink, 'y' as ds, 'Prev Year ' as cat,
PUT(intnx('year',today()-1,-1),year4.) as date, sum(Bill) as Bill, sum(transactions) as Txs FROM
save.dataset WHERE prev_y=1 GROUP BY cust_id, custlink;
CREATE TABLE yc4 AS SELECT cust_id, custlink, 'y' as ds, 'Prev YTD  ' as cat,
PUT(intnx('year',today()-1,-1),year4.) as date, sum(Bill) as Bill, sum(transactions) as Txs FROM
save.dataset WHERE prevy_y=1 GROUP BY cust_id, custlink;
QUIT;
DATA save.datasetsummary; SET mc1 mc2 mc3 mc4 mc5 yc2 yc1 yc3 yc4; RUN;


/*********** INDEX PAGE*************/
ODS HTML PATH="&path" FILE="index.html" (TITLE="CUSTOMER SUMMARY") STYLE=journal;
TITLE1 "<img src=&logo alt='Lannister Logo'>";
TITLE2 "<font size=5 color=black>Westeros Volume & Revenue Reports</font>";
TITLE3 " ";
TITLE4 "<font size=2 color=black>Last Update &current_date at &current_time PST</font>";
*Dropdown menu;
TITLE6
"
<script 'text/javascript'>

window.onload = function () {
var nav = document.getElementsByTagName('select')[0];
nav.onchange = function () {
var idx = this.selectedIndex;
window.location.href = this.options[idx].value;
}
}
</script>

<div align='center'>
 <body>
    <p>
    <b><font size=2 color=333333> Show Report: </font></b><br>
    <select name='navigation'>
    <option value=&site/index.html>Current Summary</option>
       <option>---------</option>
       <option value=&site/index.html>Current Summary</option>
       <option style='background-color:666666;color:white'>Customers</option>
    &droplist
    </p>
 </body>
</div>
<br\>
";
*Report;
PROC TABULATE DATA=save.datasetsummary MISSING;
WHERE cat in (' Last 8 Days',' MTD         ',' YTD       ');
CLASS custlink cat date;
CLASSLEV cat date / STYLE=[font_weight=bold];
VAR Bill / STYLE=[font_weight=bold];
TABLE custlink=' ' all*{STYLE=[font_weight=bold]}, cat=' '*date=' '*(Bill=' '*sum='
'*FORMAT=dollar20.2) / MISSTEXT="$0.00";
KEYWORD all / STYLE=[font_weight=bold];
RUN;
ODS HTML CLOSE;
```

SCREENSHOT OF INDEX PAGE



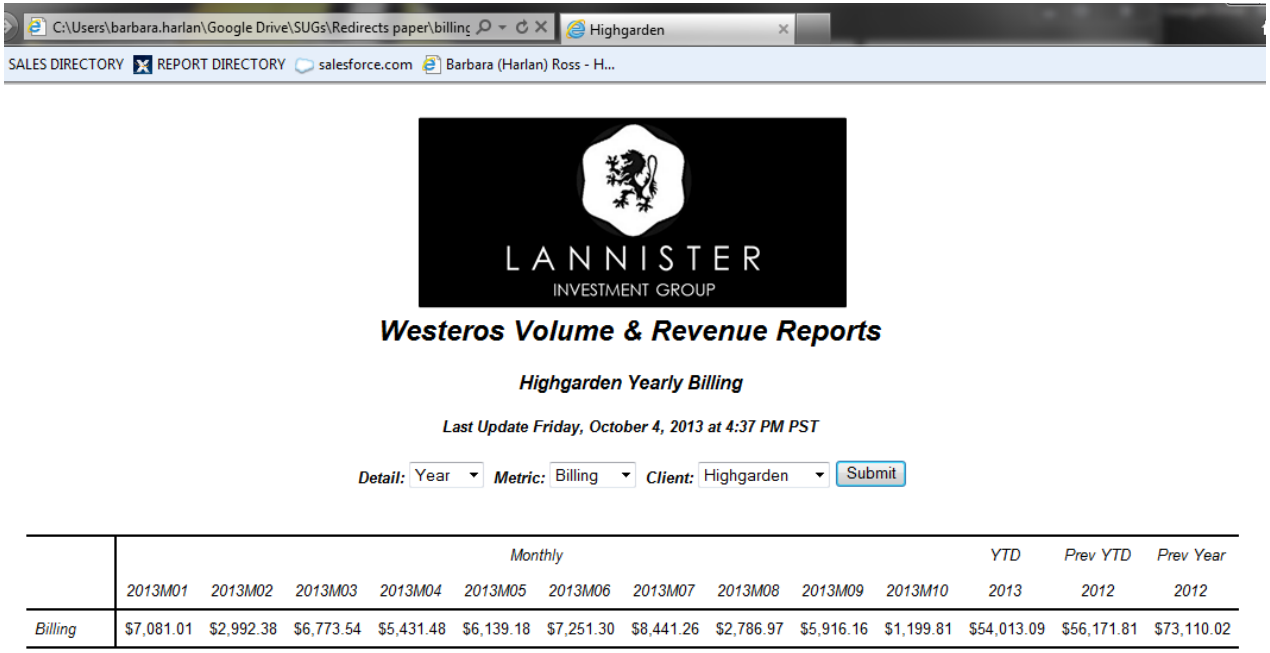
Westeros Volume & Revenue Reports

Last Update Friday, October 4, 2013 at 4:37 PM PST

Client							MTD	YTD
	09/26/13	09/28/13	09/2		0/01/13	10/03/13	October	2013
Kings Landing	$27.66	$0.00			$316.30	$0.00	$316.30	$57,824.80
Highgarden	$0.00	$0.00			,199.81	$0.00	$1,199.81	$54,013.09
Pyke	$0.00	$0.00	$0.00	$1,171.19	$0.00	$786.76	$786.76	$56,658.21
Winterfell	$0.00	$14.41	$389.65	$464.07	$0.00	$0.00	$0.00	$52,935.85
Casterly Rock	$1,095.70	$1,080.83	$722.40	$1,153.47	$864.61	$0.00	$864.61	$60,793.04
All	**$1,123.36**	**$1,095.24**	**$1,112.05**	**$2,788.73**	**$2,380.72**	**$786.76**	**$3,167.48**	**$282,224.99**

```
/*********** INDIVIDUAL REPORTS*************/
%MACRO loop(i=);
*get customer name & id;
%LET id=%SCAN( &listid, &i );
DATA _null_; SET save.dataset2;
        WHERE cust_id=&id;
        CALL SYMPUT("name",customer_name);
        RUN;

*YEARLY BILLING;
ODS HTML PATH="&path" FILE="billing_year_&id..html" (TITLE="&name") STYLE=journal;
TITLE1 "<img src=&logo alt='Lannister Logo'>";
TITLE2 "<font size=5 color=black>Westeros Volume & Revenue Reports</font>";
TITLE4 "<font size=3 color=black>&name Yearly Billing</font>";
TITLE6 "<font size=2 color=black>Last Update &current_date at &current_time PST</font>";
TITLE8
"
<form action='form.php' method='get'>
   Detail: <select name='detail'>
        <option value=year >Year</option>
        <option>--------</option>
        <option value='month'>Month</option>
        <option value='year'>Year</option>
   </select>&nbsp

   Metric: <select name='metric'>
        <option value=billing >Billing</option>
        <option>--------</option>
        <option value='billing'>Billing</option>
        <option value='txs'>Volume</option>
   </select>&nbsp

   Client: <select name='custid'>
        <option value=&id >&name</option>
        <option>--------</option>
        &droplist2
   </select>&nbsp

<input type='submit' value='Submit'>
</form>
";
PROC TABULATE DATA=save.datasetsummary MISSING;
WHERE cust_id=&id and ds='y';
CLASS cat date;
VAR Bill txs;
TABLE (Bill="Billing"*sum=' '*FORMAT=dollar20.2),cat=' '*date=' ' / MISSTEXT="$0.00";
RUN;
ODS HTML CLOSE;

*YEARLY TRANSACTIONS;
ODS HTML PATH="&path" FILE="txs_year_&id..html" (TITLE="&name") STYLE=journal;
TITLE1 "<img src=&logo alt='Lannister Logo'>";
TITLE2 "<font size=5 color=black>Westeros Volume & Revenue Reports</font>";
TITLE4 "<font size=3 color=black>&name Yearly Volume</font>";
TITLE6 "<font size=2 color=black>Last Update &current_date at &current_time PST</font>";
TITLE8
"
<form action='form.php' method='get'>
   Detail: <select name='detail'>
        <option value=year >Year</option>
        <option>--------</option>
        <option value='month'>Month</option>
        <option value='year'>Year</option>
   </select>&nbsp

   Metric: <select name='metric'>
        <option value=txs >Volume</option>
        <option>--------</option>
        <option value='billing'>Billing</option>
        <option value='txs'>Volume</option>
   </select>&nbsp
   Client: <select name='custid'>
        <option value=&id >&name</option>
        <option>--------</option>
        &droplist2
   </select>&nbsp

<input type='submit' value='Submit'>
</form>
";
PROC TABULATE DATA=save.datasetsummary MISSING;
WHERE cust_id=&id and ds='y';
CLASS cat date;
VAR bill txs;
TABLE (txs="Transactions"*sum=' '*FORMAT=comma10.),cat=' '*date=' ' / MISSTEXT="$0.00";
RUN;
ODS HTML CLOSE;

*MONTHLY BILLING;
ODS HTML PATH="&path" FILE="billing_month_&id..html" (TITLE="&name") STYLE=journal;
TITLE1 "<img src=&logo alt='Lannister Logo'>";
TITLE2 "<font size=5 color=black>Westeros Volume & Revenue Reports</font>";
TITLE4 "<font size=3 color=black>&name Monthly Billing</font>";
TITLE6 "<font size=2 color=black>Last Update &current_date at &current_time PST</font>";
TITLE8
"
<form action='form.php' method='get'>
   Detail: <select name='detail'>
        <option value=month >Month</option>
        <option>--------</option>
        <option value='month'>Month</option>
        <option value='year'>Year</option>
   </select>&nbsp

   Metric: <select name='metric'>
        <option value=billing >Billing</option>
        <option>--------</option>
        <option value='billing'>Billing</option>
        <option value='txs'>Volume</option>
   </select>&nbsp

   Client: <select name='custid'>
        <option value=&id >&name</option>
        <option>--------</option>
        &droplist2
   </select>&nbsp

<input type='submit' value='Submit'>
</form>
";
PROC TABULATE DATA=save.datasetsummary MISSING;
WHERE cust_id=&id and ds='m';
CLASS cat date;
VAR Bill txs;
TABLE (Bill="Billing"*sum=' '*FORMAT=dollar20.2),cat=' '*date=' ' / MISSTEXT="$0.00";
RUN;
ODS HTML CLOSE;

*MONTHLY TRANSACTIONS;
ODS HTML PATH="&path" FILE="txs_month_&id..html" (TITLE="&name") STYLE=journal;
TITLE1 "<img src=&logo alt='Lannister Logo'>";
TITLE2 "<font size=5 color=black>Westeros Volume & Revenue Reports</font>";
TITLE4 "<font size=3 color=black>&name Monthly Volume</font>";
TITLE6 "<font size=2 color=black>Last Update &current_date at &current_time PST</font>";
TITLE8
"
<form action='form.php' method='get'>
   Detail: <select name='detail'>
        <option value=month >Month</option>
        <option>--------</option>
        <option value='month'>Month</option>
        <option value='year'>Year</option>
   </select>&nbsp

   Metric: <select name='metric'>
        <option value=txs >Volume</option>
        <option>---------</option>
        <option value='billing'>Billing</option>
        <option value='txs'>Volume</option>
   </select>&nbsp

   Client: <select name='custid'>
        <option value=&id >&name</option>
        <option>---------</option>
        &droplist2
   </select>&nbsp

<input type='submit' value='Submit'>
</form>
";
PROC TABULATE DATA=save.datasetsummary MISSING;
WHERE cust_id=&id and ds='m';
CLASS cat date;
VAR bill txs;
TABLE (txs="Transactions"*sum=' '*FORMAT=comma10.),cat=' '*date=' ' / MISSTEXT="$0.00";
RUN;
ODS HTML CLOSE;
%MEND;

*Enumerate company list and put list into macro variable;
PROC SQL NOPRINT;
SELECT count(*) INTO :num SEPARATED BY ' ' FROM save.dataset2;
SELECT cust_id INTO :listid SEPARATED BY ' ' FROM save.dataset2;
QUIT;

*loop through client list;
%MACRO envelope_n;
    %DO j=1 %TO &num;
        %loop(i=&j);
    %END;
%MEND;
%envelope_n;
```

SCREENSHOT OF billing_year_3210.html


C:\Users\barbara.harlan\Google Drive\SUGs\Redirects paper\billing
Highgarden
SALES DIRECTORY
REPORT DIRECTORY
salesforce.com
Barbara (Harlan) Ross - H...
LANNISTER
INVESTMENT GROUP

Westeros Volume & Revenue Reports

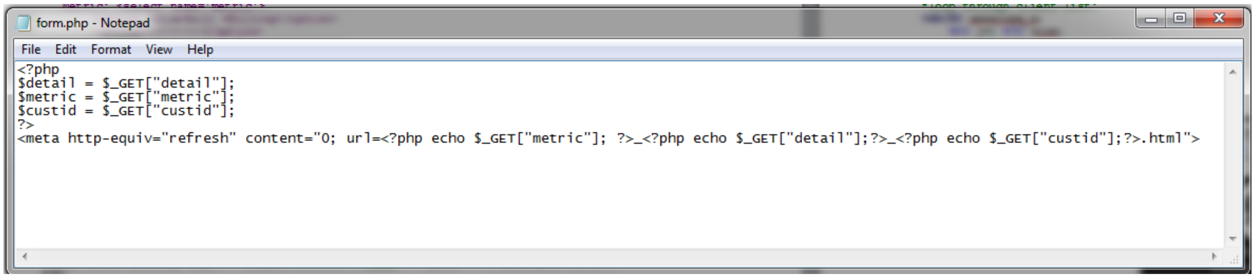
Highgarden Yearly Billing

Last Update Friday, October 4, 2013 at 4:37 PM PST

Detail: Year ▾ ***Metric:*** Billing ▾ ***Client:*** Highgarden ▾ Submit

	Monthly										*YTD*	*Prev YTD*	*Prev Year*
	2013M01	*2013M02*	*2013M03*	*2013M04*	*2013M05*	*2013M06*	*2013M07*	*2013M08*	*2013M09*	*2013M10*	*2013*	*2012*	*2012*
Billing	$7,081.01	$2,992.38	$6,773.54	$5,431.48	$6,139.18	$7,251.30	$8,441.26	$2,786.97	$5,916.16	$1,199.81	$54,013.09	$56,171.81	$73,110.02

SCREENSHOT OF FORM.PHP PROGRAM



```
<?php
$detail = $_GET["detail"];
$metric = $_GET["metric"];
$custid = $_GET["custid"];
?>
<meta http-equiv="refresh" content="0; url=<?php echo $_GET["metric"]; ?>_<?php echo $_GET["detail"];?>_<?php echo $_GET["custid"];?>.html">
```