

Lecture 2 OverView

- Day I/Lab I Questions
- Review Student Videos
- Todays Topic (**Forms**)
- Lecture



Today's Topics



- Some code hints
- Forms
- Client side validation
- Security
- Firebug hack
- Start your editors



Couple things to look for



	Non-Persistent	Persistent (none)
Local Variables	<ul style="list-style-type: none">• Variables• Attributes• Caller• Arguments• This	
Global Variables	<ul style="list-style-type: none">• Request• CGI• Form• URL	<ul style="list-style-type: none">• Server• Application• Session• Client• Cookie



Couple things to look for

Cf Variables

```
<cfset myvar=ArrayNew(1)>  
<cfset yourvar=1234>  
<cfset hervar="Joe">  
<cfset hisvar=true>
```

Cf Arrays, Lists, Objects

```
<cfset arrayAppend(myvar,'test')>  
<cfset myvar[#i#]=#i#>  
<cfset myob={}> just like JS  
<cfoutput>#ArrayLen(myvar)#</cfoutput>  
  
<cfset numlist="1,2,3,4,5">  
<cfset beatles="paul john ringo george">
```

PHP Variables

```
$myvar=Array();  
$yourvar=1234;  
$hervar="Joe";  
$hisvar=true;
```

PHP Arrays

```
$myAr = array(); or array("a",true);  
$myAssocArray = array("name"=>"orcun");  
count($myAr);  
array_push($myAr,"new val");
```



Couple things to look for

Some of the array methods for CF

ArrayNew() Creates a new array.

ArrayLen() Determines the number of elements in an array.

IsArray() Returns true if value is an array; false otherwise.

ArrayIsEmpty() Returns true if array has zero elements.

ArraySort() Sorts array elements according to a sort type and sort order.

ArrayAppend() Appends an element to an array.

Cf Loops

`<cfloop array="#myvar#" index="i">` or `<cfloop from=0 to=#arraylen(myvar)#`

`<cfloop index="beatle" list="#beatles#" delimiters=" ">`

`<cfloop condition="myVar eq false">`

`<cfloop query="getMovies">`

PHP Loops

`for($i=0; $i<10;$i++){...`

`while ($i <= 10) {...`

`foreach(){...`



Couple things to look for

If you want to write your coldfusion just like JS or AS, you can use cfsript tags

```
<cfsript>
```

```
for(i=0;i<5;i++){
```

```
    writeOutput('my name is ' & i);
```

```
}
```

```
</cfsript>
```



Couple things to look for

CF conditionals

```
<cfif firstName eq "Bono">
```

```
<cfif firstName neq "Bono">
```

```
<cfelseif firstName eq "Ozzy">
```

```
<cfif albumSales lt 100>
```

```
<cfif albumSales gt 1000000>
```

```
<cfif age gte 80>
```

PHP conditionals

```
if($name == "myname"){
```

```
if($name != true){
```





Couple things to look for

CF forms

```
<cfform name="RunningForm" method="post" action="AddEntry-cfform.cfm" format="html">  
<cfinput type="text" name="time" size="20" value="" required="yes" validate="regex" pattern="\d{1,3}:\d{0,2}$" message="You must enter a valid time in the format mm:ss.">
```

with Regex for client site validation built-in CF

```
<cfinput type="text" name="time" size="20" value="" required="yes" validate="regex" pattern="\d{1,3}:\d{0,2}$" message="You must enter a valid time in the format mm:ss.">
```

PHP Forms

Well php uses HTML form structure

```
<form name="myform" action="" method="post">...
```




Couple things to look for

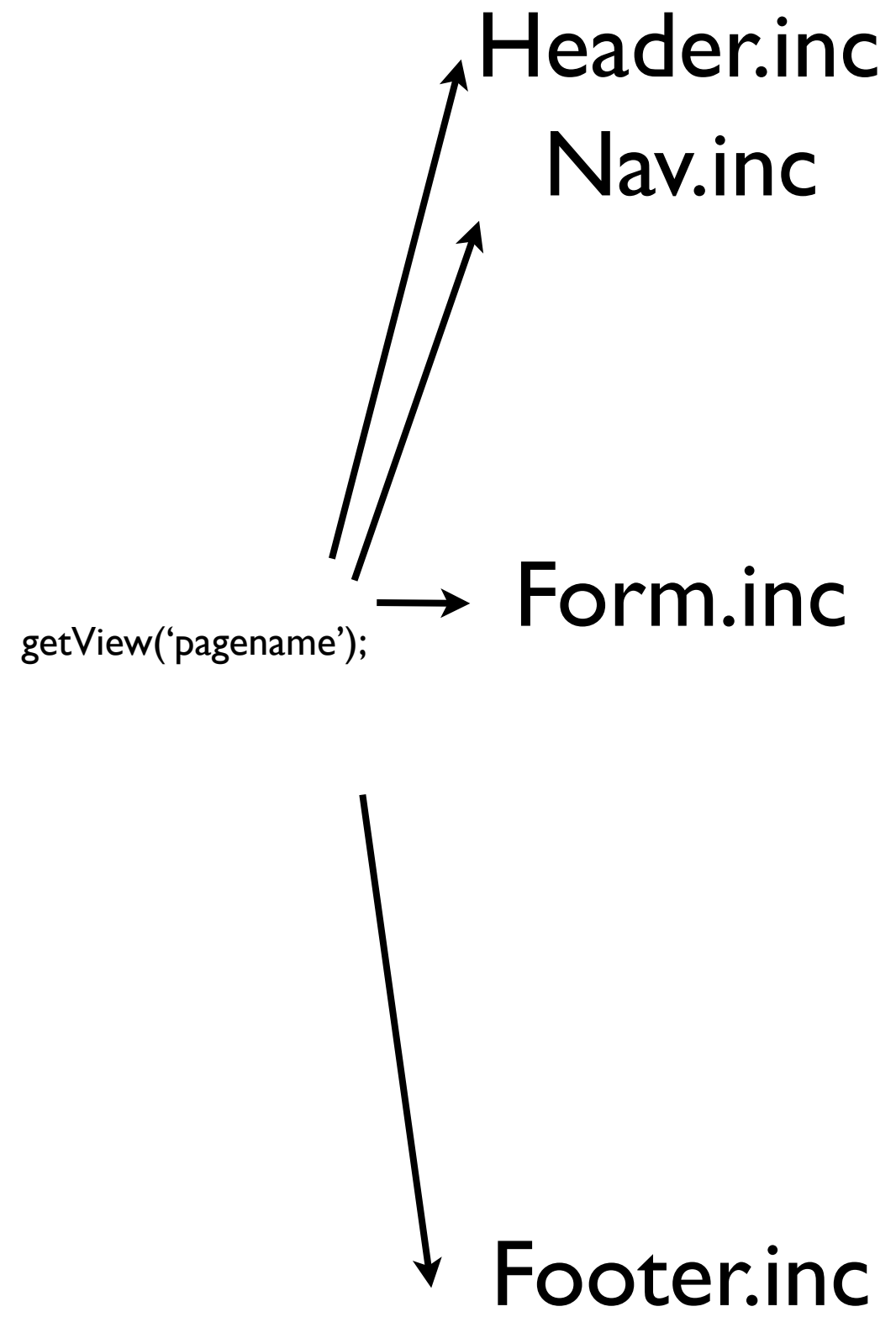
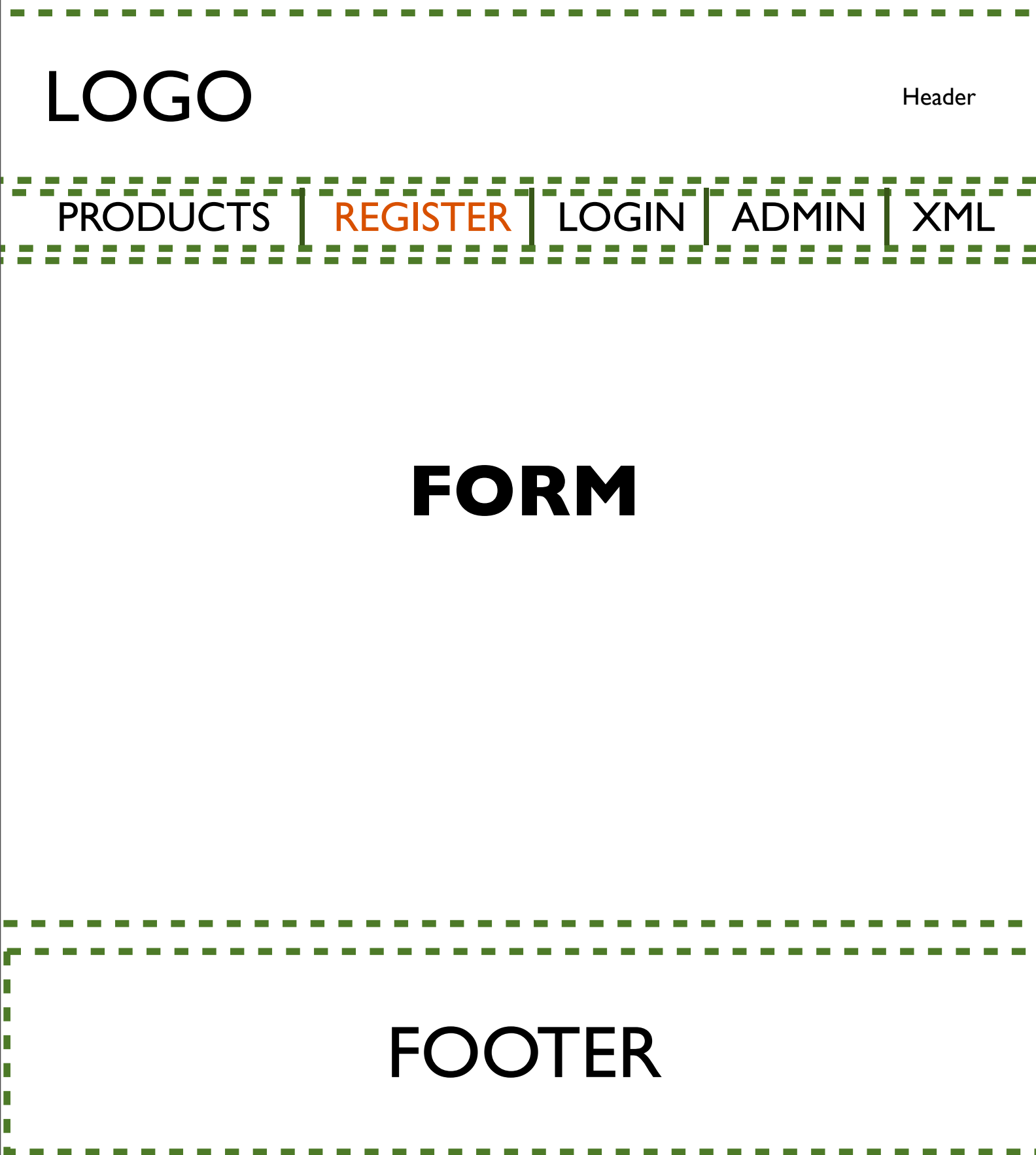
How would you validate on the server side?

CF forms

```
<cfparam name="Form.date" default="">
<cfif NOT isDate(FORM.date)>
<cfoutput>#isDefined("FORM.time")#</cfoutput>
<cfif NOT isNumeric(intDistance)>
  <cfif REFindNocase("\d[1-4]$", Form.name)>
```

PHP forms

```
$_GET or $_POST["name"];
empty(
isset(
preg_match('#[^a-z]+$#i', $_GET["name"])
```





Today's Lab (2)

Please refer to <http://wddbs.com/SSL lab #2>

In this lab you will perform the same field-by-field form validation you have done in previous courses on the client side with JavaScript, but this time on the server side. Your forms will include a number of different data types which will require different ways of validating their data.