

# PHP Form Handling

“ The PHP `$_GET` and `$_POST` variables are used to retrieve information from HTML forms, like user input.

“ **`$_GET` Variable**

- predefined `$_GET` variable is used to collect values in a form with `method="get"`

“ **`$_POST` Variable**

- predefined `$_POST` variable is used to collect values from a form sent with `method="post"`.

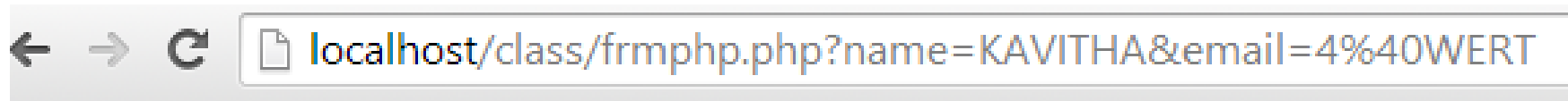
“ **`isset` function**

- Determine if a variable is set and is not NULL.

# PHP Form Handling

## ” Difference between GET & POST

<code>\$_GET</code>	<code>\$_POST</code>
GET requests can be cached	POST requests are never cached
GET requests remain in the browser history	POST requests do not remain in the browser history
GET requests can be bookmarked	POST requests cannot be bookmarked
GET requests should never be used when dealing with sensitive data	
GET requests have length restrictions	POST requests have no restrictions on data length



## ” Form Processing methods

# Form Example1

```
<html><head><title>Calculator</title>
</head><body>
<center>
<h1>Claculator</h1>
<form method="POST" action="" >
Enter 1st value : <input type="text" name="val1"><br><br>
Enter 2nd value : <input type="text" name="val2"><br><br><br><br>
<input type="radio" name="cal" value="1">Add<br>
<input type="radio" name="cal" value="2">Subtract<br>
<input type="radio" name="cal" value="3">Multiply<br>
<input type="radio" name="cal" value="4">Divide<br><br>
<input type="Submit" name="Submit">
</form>
<?php
if(isset($_POST['Submit']))
    {if($_POST['cal']==1)
        echo "<H1>The Answer is " .($_POST['val1'] + $_POST['val2']);
        elseif($_POST['cal']==2)
        echo "<H1>The Answer is " .($_POST['val1'] - $_POST['val2']);
        elseif($_POST['cal']==3)
        echo "<H1>The Answer is " .($_POST['val1'] * $_POST['val2']);
        elseif($_POST['cal']==4)
        echo "<H1>The Answer is " .($_POST['val1'] / $_POST['val2']);
    }
?>
</center></body></html>
```

# Form Example2

## *Form.html*

```
<html>
<body>
<form action="welcome1.php"
method="post">
Name: <input type="text"
name="fname">
Age: <input type="text"
name="age">
<input type="submit">
</form>

</body>
</html>
```

## *welcome.php*

```
<html>
  <body>

Welcome <?php echo
$_POST["fname"]; ?>!<br>
You are <?php echo
$_POST["age"]; ?> years old.

  </body>
</html>
```

# EXERCISES

- ” Create a form to accept Name, Gender, Age(DOB), Address, Registration Status, Id Proof, etc..
- ” Determine voter eligibility and display one of three messages. The messages and criteria are as following.

Message	Criteria
You are too young to vote	Person is younger then 18 years old
You can vote	Person at least 18 years old and is register to vote
You must register before you can vote	Person is at least 18 years old but is not registered to vote