CIS-355: 04-Idempotency

What is idempotency?

Idempotency (or idempotence) means that an action can be performed multiple times without changing the result. In mathematics, an example of idempotency might be absolute value function, or maximum function. From Wikipedia (http://en.wikipedia.org/wiki/Idempotence)

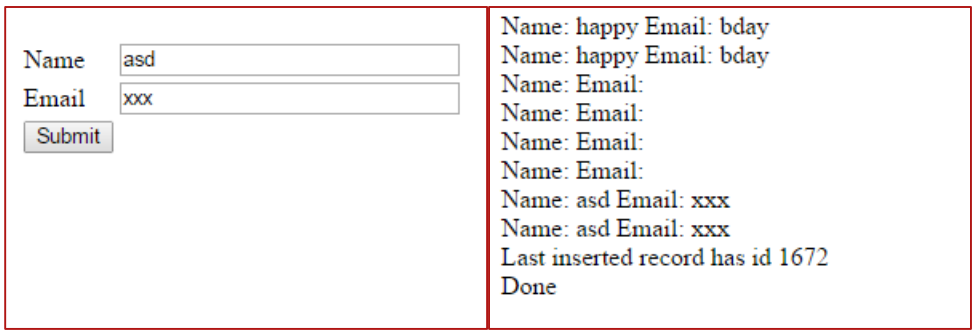
> A unary operation (or function) is idempotent if, whenever it is applied twice to any value, it gives the same result as if it were applied once; i.e., $f(f(x)) \equiv f(x)$. For example, the absolute value: abs(abs(x)) ≡ abs(x).
>
> A binary operation is idempotent if, whenever it is applied to two equal values, it gives that value as the result. For example, the operation giving the maximum value of two values is idempotent: max (x, x) ≡ x.

In computer science we mean operations performed by a computer. In web development that means in this case requests made of the server. From http://www.restapitutorial.com/lessons/idempotency.html :

> From a RESTful service standpoint, for an operation (or service call) to be idempotent, clients can make that same call repeatedly while producing the same result. In other words, making multiple identical requests has the same effect as making a single request. Note that while idempotent operations produce the same result on the server (no side effects), the response itself may not be the same (e.g. a resource's state may change between requests).
>
> The PUT and DELETE methods are defined to be idempotent. However, there is a caveat on DELETE. The problem with DELETE, which if successful would normally return a 200 (OK) or 204 (No Content), will often return a 404 (Not Found) on subsequent calls, unless the service is configured to "mark" resources for deletion without actually deleting them. However, when the service actually deletes the resource, the next call will not find the resource to delete it and return a 404. However, the state on the server is the same after each DELETE call, but the response is different.

Problem

When inserting records using POST method, duplicates occur if browser is refreshed. See: http://csis.svsu.edu/~gpcorser/, specifically, http://csis.svsu.edu/~gpcorser/demo02.php . Note that refreshing the browser causes multiple records to be inserted.

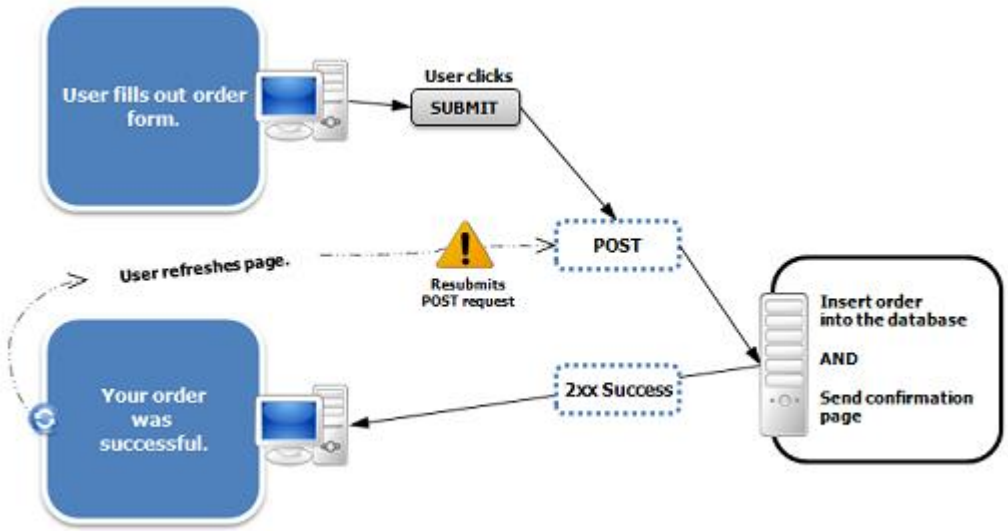


Analysis

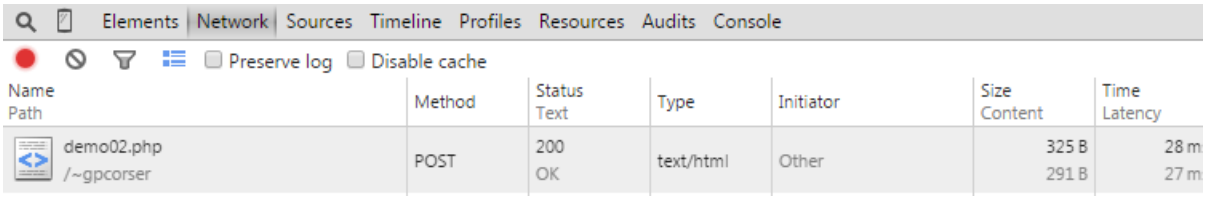
Recall demo02.php code.

```
<?php

// Step 0: ----- Set variables from HTML form
$name = $_POST["name"];
$email = $_POST["email"];

// Step 1: ----- Connect to database -----
$hostname="localhost";
$username="student";
$password="learn";
$dbname="gpcorser";
$usertable="table01";
$con = mysql_connect($hostname,$username, $password)
  or die ("<html><script language='JavaScript'>alert('Cannot connect.'),history.go(-
1)</script></html>");
mysql_select_db($dbname);

// Step 2: ----- Check if any records in table -----
$query = "SELECT * FROM $usertable";
$result = mysql_query($query);

// Step 3: ----- If records, print name field and add another random record
if($result) { // if $result is empty there is no output and no message
  while($row = mysql_fetch_array($result)){
    $val1 = $row[1];
       $val2 = $row[2];
    echo "Name: ".$val1." Email: ".$val2."<br>"; // generates html code
  }

  $query = "INSERT INTO $dbname.$usertable (`id`, `name`, `email`) VALUES (NULL, '$name',
'$email')";
  $result2 = mysql_query($query);
  printf("Last inserted record has id %d\n", mysql_insert_id());
  echo "<br>Done<br>";
}
?>
```

The diagram below outlines the problem.

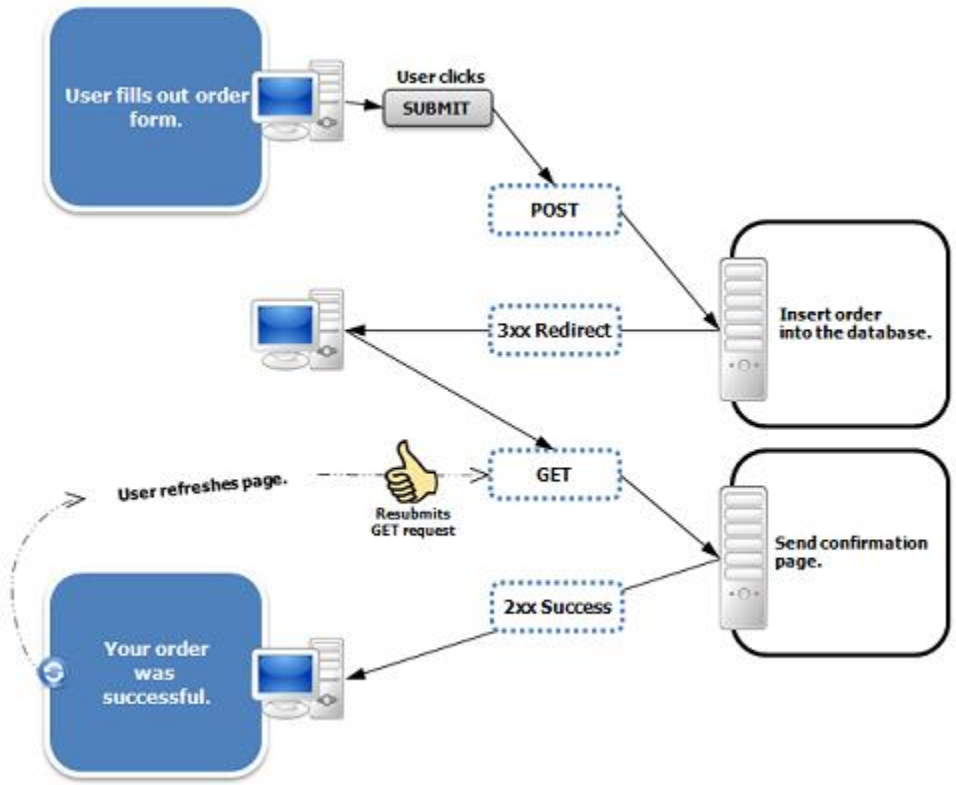


Inpect element in chrome shows what's going on at the network level.



Solution

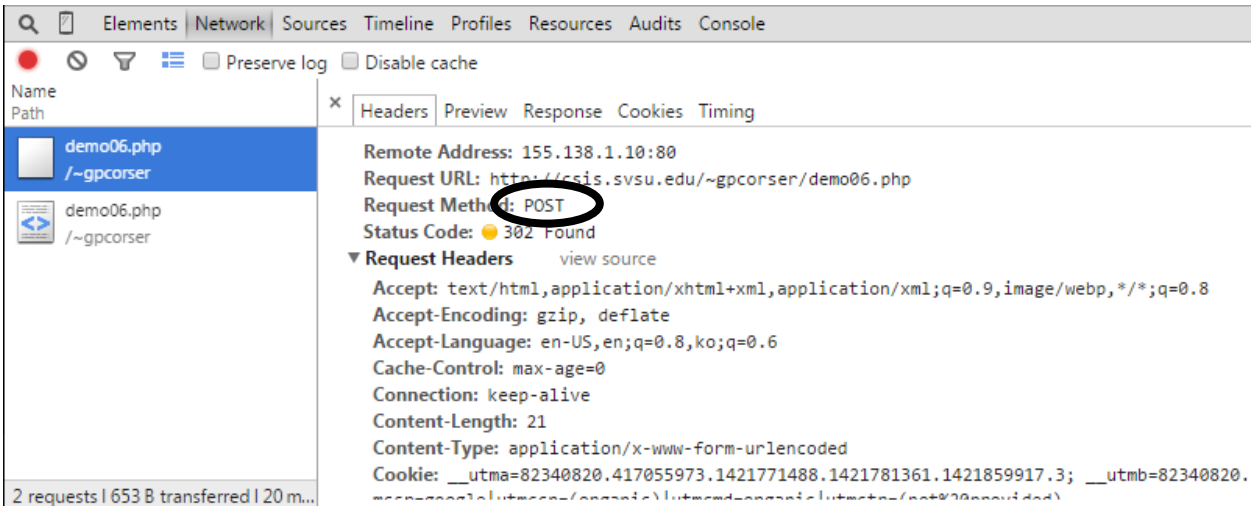
A widely accepted solution is PRG, or Post-Redirect-Get.

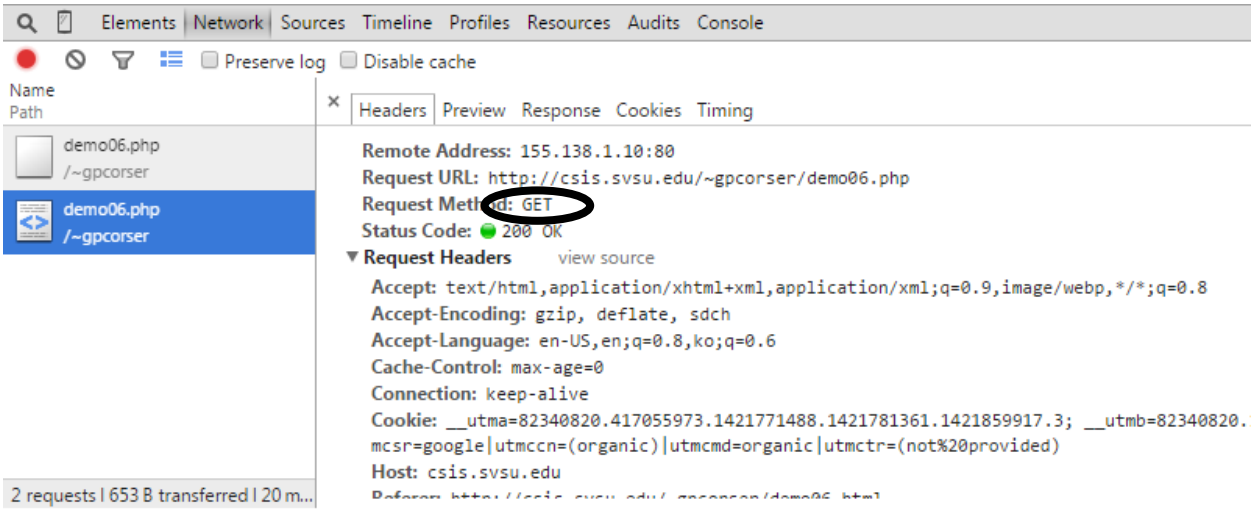


New code: demo06.php... snippet is just section, "Step 3."

```
// Step 3: ----- If records, print name field and add another random record
// Post-get-redirect
if($_POST){//the post part...so if we are submitting something we are doing it in here
technically you can put/patch too
 $query = "INSERT INTO $dbname.$usertable (`id`, `name`, `email`) VALUES (NULL, '$name',
'$email')";
 $result2 = mysql_query($query);
//the magic we are setting the header to become a RESTful get ... or the redirect part
 header("Location: ".$_SERVER['REQUEST_URI']);
 exit();//safety valve
}
```

Inpect element in chrome shows what's going on at the network level.





For more information on the header() function

See: http://php.net/manual/en/function.header.php