# COMP 4021
Internet Computing

Use of Hidden Fields

Gibson Lam and David Rossiter

Hidden Fields

- Hidden fields are useful for implementing 'system' variables shared between the server and the client

- A hidden field is also useful for the client to send information to the server (without using the standard input fields such as text fields and selection boxes)

Example – Color Selector

- An HTML form does not provide a simple input field which handles color selection

- We can build a color selector using several DIVs and then send the color using a hidden field
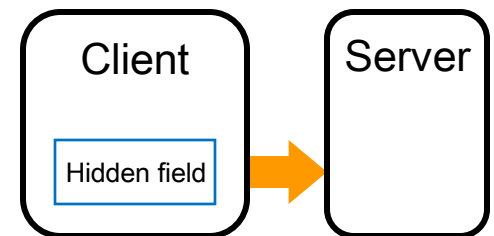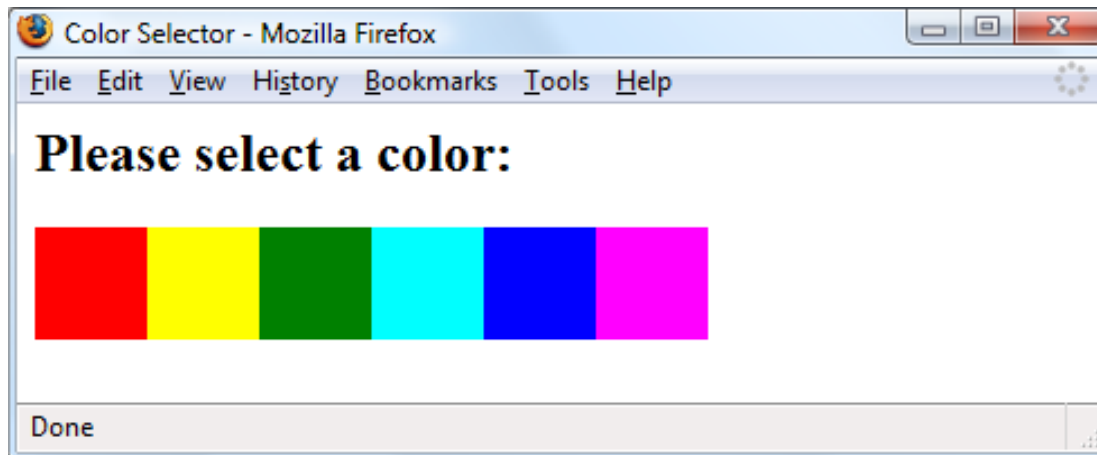
Example 1

```
<html>
  <head>
    <title>Color Selector</title>
    <style>
      div {
          position: absolute;
          width: 50px;
          height: 50px;   }
    </style>
    <script>
    function select(color) {
        document.getElementById("color").value=color;
        document.getElementById("color").form.submit();  }
    </script>
  </head>
```

Set hidden field value and submit the form using JavaScript

Continued in next slide

Example 1

The DIVs for a simple color selector

```
<body> <h2>Please select a color:</h2>
   <div style="position:relative">
   <div style="background-color:red;left:0px"
       onclick="select('red')"></div>
   <div style="background-color:yellow;left:50px"
       onclick="select('yellow')"></div>
   <div style="background-color:green;left:100px"
       onclick="select('green')"></div>
   <div style="background-color:cyan;left:150px"
       onclick="select('cyan')"></div>
   <div style="background-color:blue;left:200px"
       onclick="select('blue')"></div>
   <div style="background-color:magenta;left:250px"
       onclick="select('magenta')"></div>  </div>
   <form action="..." method="GET">
   <input type="hidden" name="color" id="color" value="" /></form>
 </body>  </html>
```

Setting Hidden Field in the Server

- A hidden field can be changed by server side code

- For example, the hidden field can be changed to a different color by some code on the server

```
...
<input type="hidden" name="color" id="color"
    value="<?php print $my_color; ?>" />
...
```

Example 2

```php
<?php
   $my_color = ... // set the color of the hidden field
?>
<html>  <head>
     <title>Color Selector</title>
     <style>. . .</style>
     <script>
     function select(color) {
        var fld = document.getElementById("color");
        if (color != fld.value) {
           if (!confirm("Do you want to use the new color?"))
              return;
           fld.value=color;
           fld.form.submit();
        }  }
     </script>  </head>
```

On the client submit the form using JavaScript only if the value is changed to a new value

Example 2

```
<body>   <h2>Please select a color:</h2>
   <div style="position:relative">
      <div style="background-color:red;left:0px"
         onclick="select('red')"></div>
      <div style="background-color:yellow;left:50px"
         onclick="select('yellow')"></div>
      <div style="background-color:green;left:100px"
         onclick="select('green')"></div>
      <div style="background-color:cyan;left:150px"
         onclick="select('cyan')"></div>
      <div style="background-color:blue;left:200px"
         onclick="select('blue')"></div>
      <div style="background-color:magenta;left:250px"
         onclick="select('magenta')"></div>   </div>
   <form action="..." method="GET">
      <input type="hidden" name="color" id="color"
         value="<?php print $my_color; ?>" /> </form></body></html>
```

The color hidden field with a value set by the server

Take Home Message

- Hidden fields in forms are very useful in passing states of the clients to the server
    - Those states can be passed to the server when the form is submitted
    - No additional action is required other than submitting the form (which the user will do anyway); user indeed does not know about this