

# EECS1012

## Net-centric Introduction to Computing

Lecture 6

HTML Forms

(User Interface Elements)

---

### **Acknowledgements**

Contents are adapted from web lectures for “Web Programming Step by Step”, by M. Stepp, J. Miller, and V. Kirst.

Slides have been ported to PPT by Dr. Xenia Mountroudou.

These slides have been edited for EECS1012, York University.

The contents of these slides may be modified and redistributed, please give appropriate credit.

(Creative Commons) Michael S. Brown, 2018.

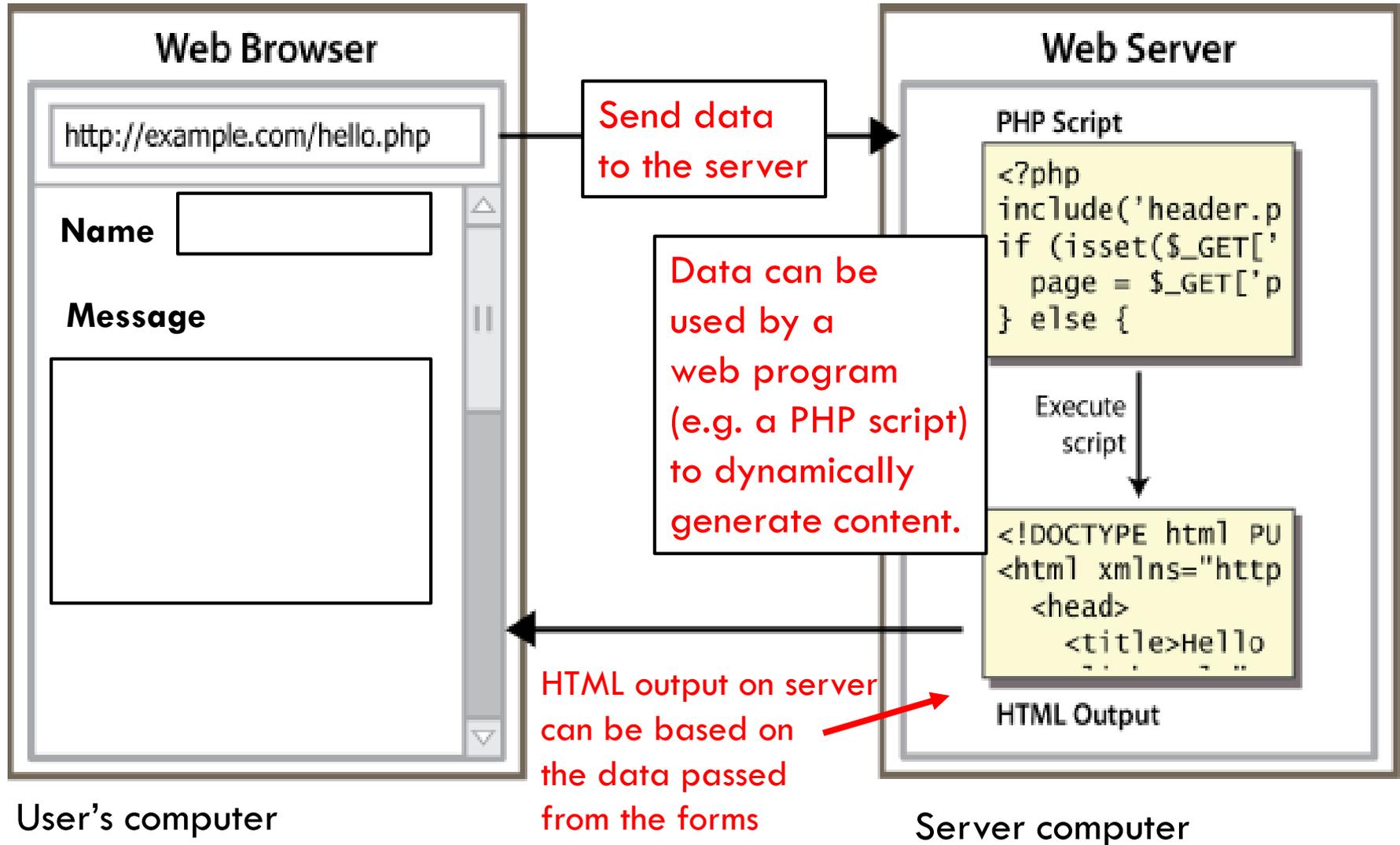
# Two way communication

2

- In this lecture we will look at ways to send data from the web browser to the web server
- This will allow the web server to prepare an HTML page based on the data we have sent.
- This provides a two way communication between a web browser and a web server.

# Sending Data to a Web Server

3



# Quick example

4

- Recall that a URL can have text after the file name:

- query string: a set of parameters passed to a web program

<http://www.google.com/search?q=poutine>

- parameter q is set to "poutine"

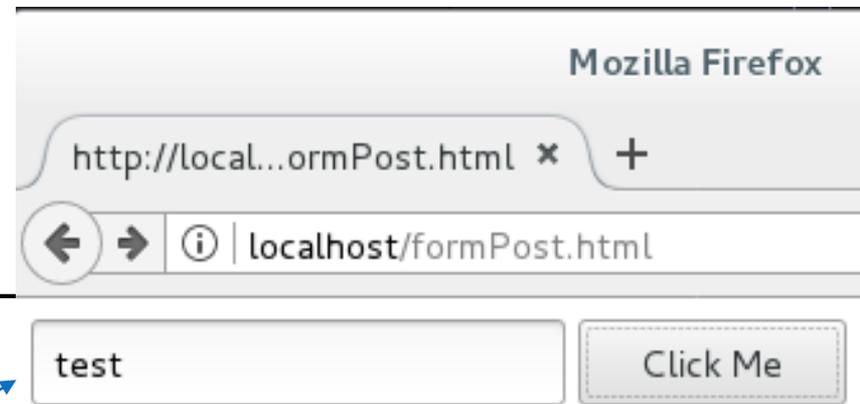
(from our first lecture)

# HTML Forms example

5

- HTML Forms provide user interface (UI) controls that gather data that is sent to the web server.

```
<body>
  <form action="https://www.eecs.yorku.ca/~mbrown/EECS1012/testForm.php"
    method="get">
    <input name="q">
    <input type="submit" value="Click Me">
  </form>
</body>
```

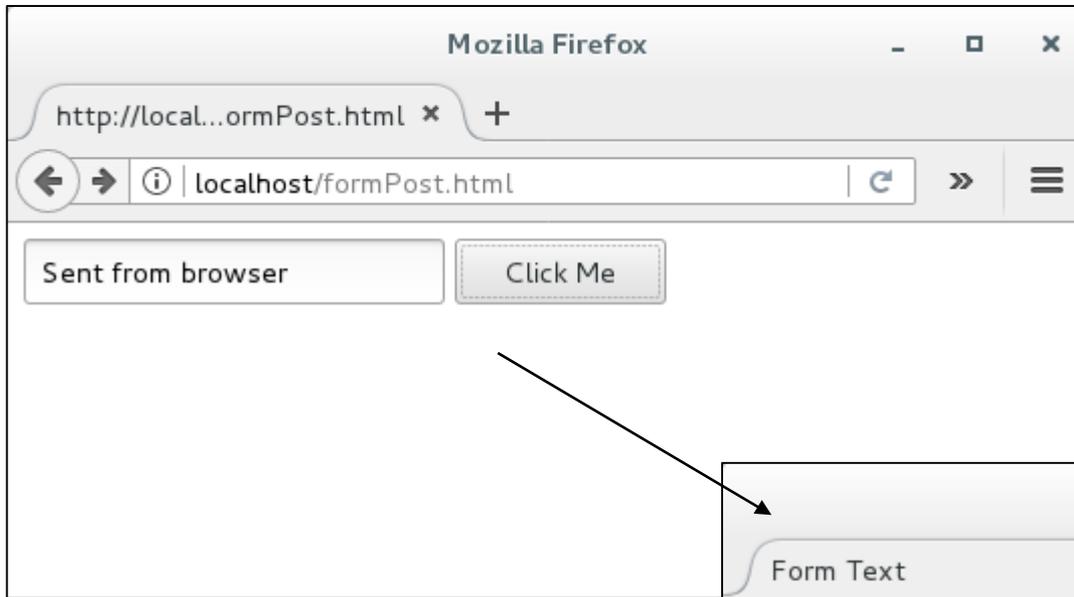


`<input name="q">` Place to input text.  
Variable name is "q"

Submit (button) text of button is "Click Me"  
`<input type="submit" value="Click Me">`

# HTML Forms cont'

6



formPost.html

formTest.php



We have sent data from our HTML page to the web server program! Data is sent in "pairs". The name of the data, and the value.

The name of the data is "q", the value is: "Sent from browser". Notice the user has typed in "Sent from browser" into the form UI.

# In this lecture . . HTML Forms

7

- We will look at the various type of HTML forms
  - ▣ HTML forms provide user interface (UI) controls for users to input data
- We will not cover how the backend program works. Unfortunately we don't have enough time in this class.
- At this stage, we will just focus on how to send data. Web programs will be provided for use.

8

# Form Basics

# HTML forms

9

- **form:** a group of UI controls that accepts information from the user and sends the information to a web server
- the information is sent to the server as a "query string"



The image shows a web form with three input fields and one button. The first field is labeled 'Name:' and is a single-line text input. The second field is labeled 'E-mail:' and is a single-line text input. The third field is labeled 'Message:' and is a multi-line text area. Below the message field is a button labeled 'Send your message'.

# HTML form: `<form>`

10

```
<form action="destination URL">  
  form controls  
</form>
```

HTML

- form element tells HTML that this page will submit information to the URL specified.
- when form has been filled out and **submitted**, its data will be sent to the action attribute's **destination URL**
- **NOTE: Only form elements nested inside the `<form>` `</form>` tags will be sent to the destination URL**

# A Test Destination URL

11

<https://www.eecs.yorku.ca/~mbrown/EECS1012/testForm.php>

The URL above is a simple web program that accepts a form post and prints out all values sent to the program. You may use this for testing.

← → ↻ Secure | <https://www.eecs.yorku.ca/~mbrown/EECS1012/testForm.php?first=Abdel&last=Zhang>

## Data sent from HTML Form

Name	Value
first	Abdel
last	Zhang

# Form – a simple example

12

```
<form action="http://www.google.com/search">
  <p>
    Let's search Google:
    <input name="q">
    <input type="submit" value="Submit Query">
  </p>
</form>
```

HTML

Let's search Google:

- This example will send a query to [www.google.com](http://www.google.com) with name "q" and the value written in the text field. The data is sent with the submit button is pressed.

13

# Form controls

# <input . . . >

14

- <input . . . > can be used to specify several types of different user interface controls
- The type attribute is used to select the type of UI control

Type	Description
<input type="text"...>	Defines a one-line text input field
<input type="radio"..>	Defines a radio button (for selecting one of many choices)
<input type="submit"..>	Defines a submit button (for submitting the form)
<input type="checkbox"..>	Defines a checkbox

# <input type="text"> (1 / 3)

15

```
<form action="http://localhost/formTest.php">  
<input type="text" name="q" value="email">  
<input type="submit">  
</form>
```

# <input type="text">

(2/3)

16

```
<form action="http://localhost/formTest.php">
```

```
<input type="text" name="q" value="email">
```



This means  
the input  
is text field.



Name to  
associate with  
this data when  
it is posted.



default value to put  
in the text field.

email

Submit

---

<http://localhost/formTest.php?q=email>

**q="email"** the word email was entered by the user

When "submit" is  
clicked, the data  
with value q will be  
passed to a web program.

# <input type="submit"> (3/3)

17

```
<form action="http://localhost/formTest.php">  
<input type="text" name="q" value="email"> <br>  
<input type="submit" value="Click me">
```

Creates a button  
that when clicked  
submits the data

text to place in the button  
when displayed. Default is "Submit".



The image shows a rendered HTML form. At the top is a text input field containing the text "email". Below the input field is a button with the text "Click me".

\*why does this example have the text field on the top and button on the bottom. Notice the <br> after the first <input>. The <input . . > elements are in-line elements.

# maxlength attribute

18

```
<input type="text" name="code" maxlength="5">
```

maxlength attributes will only allow a fixed amount of characters to be typed. In this example, this input text will only allow 5 characters.

# <input type="radio">

19

```
<form action=" http://localhost/formTest.php">  
  <input type="radio" name="age" value="child" checked> Child <br>  
  <input type="radio" name="age" value="adult"> Adult <br>  
  <input type="radio" name="age" value="senior"> Senior <br>  
  <input type="submit">  
</form>
```

Radio button

variable name

value of the  
button

HTML text

Child  
 Adult  
 Senior

Note that the text shown is HTML text .. it is not part of the <input> tag.

<http://localhost/formTest.php>

**Data sent associated with the name will be the value.**

When "submit" is clicked, the data with value age will be passed to the PHP program.

# <input type="checkbox">

20

```
<form action=" http://localhost/formTest.php">  
  <input type="checkbox" name="vehicle" value="Bike"> bike  
  <input type="checkbox" name="vehicle" value="Car" checked> car  
  <input type="submit">  
</form>
```

variable name

values

default check is on

bike

car

Submit

<http://localhost/formTest.php>

Sends data pair "name" = "car" to webserver.

# Text boxes: `<textarea>`

21

```
<textarea rows="4" cols="20">  
Type your comments here.  
</textarea>
```

HTML

```
Type your comments  
here.
```

- `<textarea>` `</texarea>` creates a text box area of size "rows" and "cols". Note that text appearing between the open/end tags is the default data to be placed in the text area.

# Text labels: <label>

22

```
<form action="...">
<label><input type="checkbox" name="bold"> Bold </label>
<label><input type="checkbox" name="italic"> Italic </label>
</form>
```

HTML

**Bold**  *Italic*

- One problem with checkboxes and radio buttons is that the associated text is just basic HTML
- *The <label> tag allows you to associate text with a radio button or checkbox*
- In this example, because of the <label> tag, you click on the words (Bold or Italic) to activate the radio button

# Text labels: <label> (ver2)

23

```
<form action="...">
<label for="bold"> Bold </label>
<input id="bold" type="checkbox" name="bold">
<label for="italic"> Italic </label>
<input id="italic" type="checkbox" name="italic">
</form>
```

HTML

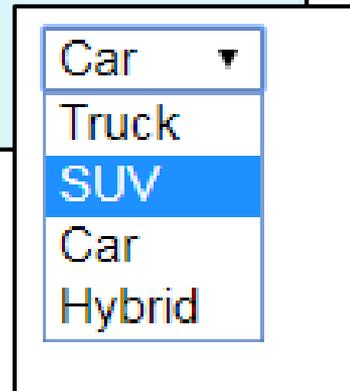
**Bold**  *Italic*

- Another way to do this is to create an "id" for each radio button or checkbox, then associate the label using for="id"
- This is shown above.
- Personally, I prefer the version on the previous slide, but both are valid.

# Drop down lists: `<select>` & `<option>`

24

```
<select name="vehicletype">
  <option>Truck</option>
  <option>SUV</option>
  <option selected="selected">Car</option>
  <option>Hybrid</option>
</select>
```

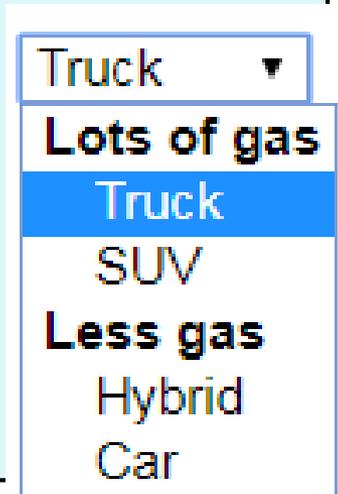


- `<select name="..">` creates a pulldown list
- option element represents each choice from a pull down list
- optional selected attribute sets which one is initially chosen

# Option groups: <optgroup>

25

```
<select name="vehicletype">
  <optgroup label="Lots of gas">
    <option>Truck</option>
    <option>SUV</option>
  </optgroup>
  <optgroup label="Less gas">
    <option>Hybrid</option>
    <option>Car</option>
  </optgroup>
</select>
```

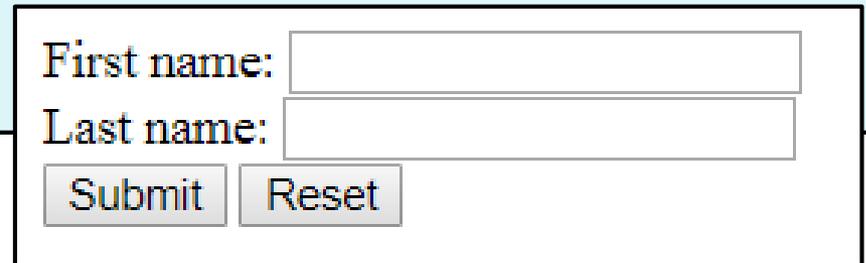


- `optgroup` allows us to group our items within the dropdown list.

# Reset button

26

```
<form action=... method="get">
<div>
First name: <input type="text" name="firstname"> <br>
Last name: <input type="text" name="lastname"> <br>
<input type="submit">
<input type="reset">
</form>
```



First name:

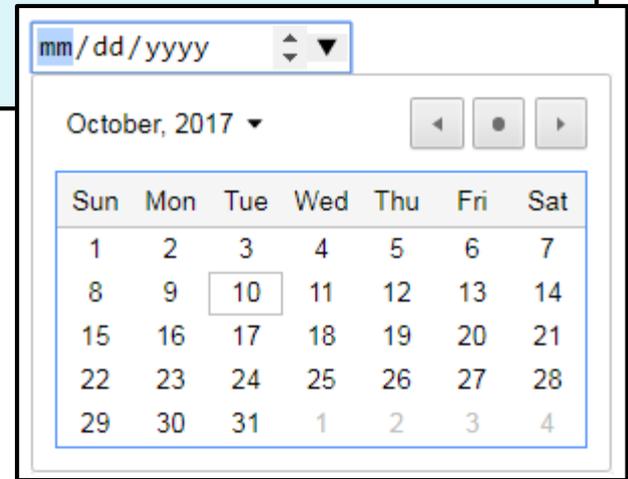
Last name:

- ❑ Reset type create a button that clears all the fields/inputs
- ❑ This is useful to allow the user a quick way to reset all the values in a form.

# HTML5 UI controls

27

```
<form action="...">
<div> <input type="date" name="date"> </div>
<input type="submit">
</form>
```



The image shows a date picker UI control. At the top, there is a text input field with the placeholder "mm/dd/yyyy" and a dropdown arrow. Below this is a calendar for "October, 2017". The calendar has a header row with days of the week: Sun, Mon, Tue, Wed, Thu, Fri, Sat. The dates are arranged in a grid. The date "10" is highlighted with a white background and a blue border. Navigation buttons (back, forward, and a central dot) are visible to the right of the calendar header.

Sun	Mon	Tue	Wed	Thu	Fri	Sat
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4

- There are a number of HTML5 specific UI types
- I'm showing date above
- The problem, however, is some browsers do not support these newer UI controls . . . (e.g., browsers in our lab)

# Grouping input: `<fieldset>`, `<legend>`

28

```
<fieldset>
  <legend>Credit cards:</legend>
  <input type="radio" name="cc" value="visa"
checked="checked" /> Visa
  <input type="radio" name="cc" value="mastercard" />
MasterCard
  <input type="radio" name="cc" value="amex" />
American Express
</fieldset>
```



Credit cards:  Visa  MasterCard  American Express

- `fieldset` groups related input fields, adds a border; `legend` supplies a caption

# Hidden input parameters

29

```
<input type="text" name="username"> Name <br>  
<input type="text" name="sid"> SID <br>  
<input type="hidden" name="school" value="YU">  
<input type="hidden" name="year" value="2048">
```

*HTML*

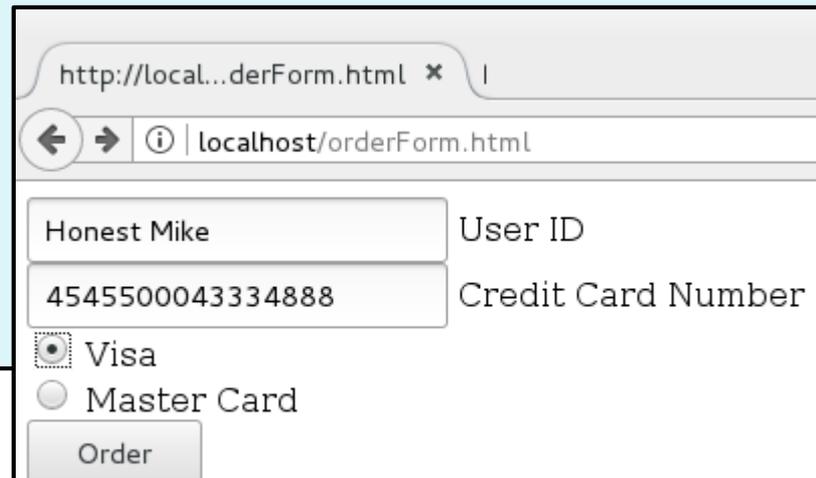
- an invisible parameter that is still passed to the server when the form is submitted
- useful for passing on additional information that isn't modified by the user

# Putting it together. . . HTML forms

30

```
<html>
<body>
<form action="http://localhost/printForm.php">
<div>
  <input type="text" name="userid" maxlength="14"> User ID <br>
  <input type="text" name="ccard_number" maxlength="14">Credit Card Num <br>
  <label> <input type="radio" name="creditcard" value="visa"> Visa </label> <br>
  <label> <input type="radio" name="creditcard" value="master"> Master Card
  </label> <br>
  <input type="submit" value="Order">
</div>
</form>
</body>
</html>
```

URL of the PHP program to  
pass our form's info to



The screenshot shows a web browser window with the URL `http://localhost/orderForm.html`. The form contains the following elements:

- A text input field containing "Honest Mike" with the label "User ID".
- A text input field containing "4545500043334888" with the label "Credit Card Number".
- Two radio button options: "Visa" (selected) and "Master Card".
- An "Order" submit button.

# Common UI control errors

31

- “I changed the form's HTML code ... but when I refresh, the page doesn't update!”
- By default, when you refresh a page, it leaves the previous values in all form controls
  - if you want it to clear out all UI controls' state and values, you must do a full refresh
    - Firefox: Shift-Ctrl-R
    - Mac: Shift-Command-R
- Consider: adding a "reset" button to your form!

# Styling form controls in CSS

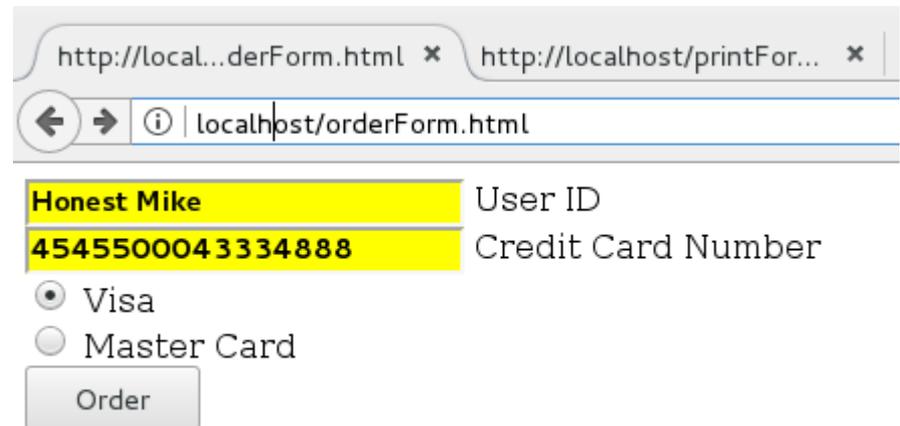
32

```
input[type="text"] {  
    background-color: yellow;  
    font-weight: bold;  
}
```

CSS

- We can use CSS to style our various UI controls. The following is a selector for input with type "text"

CSS above applied to slide 28's code.



The screenshot shows a web browser window with two tabs. The active tab is titled 'localhost/orderForm.html'. The browser address bar shows 'localhost/orderForm.html'. The page content includes a form with the following elements:

- A text input field containing 'Honest Mike', which is highlighted in yellow.
- A text input field containing '4545500043334888', which is highlighted in yellow.
- A label 'User ID' to the right of the first input field.
- A label 'Credit Card Number' to the right of the second input field.
- Two radio button options: 'Visa' (selected) and 'Master Card'.
- An 'Order' button at the bottom.

# Recap

33

- Forms gives you a way to create UI elements to allow the users to input data
- The `<input type="submit">` creates a button that will submit the form
  - ▣ The browser will automatically generate a URL with the form's data embedded in the URL
  - ▣ This is used to access the PHP program

34

# Submitting data – GET vs POST

# HTTP GET vs. POST requests

35

- GET : asks a server for a page or data
  - ▣ if the request has parameters, they are sent in the URL as a query string
  - ▣ **All examples you have seen so far have used GET**
- POST : submits data to a web server and retrieves the server's response
  - ▣ if the request has parameters, they are embedded in the request's HTTP packet, not the URL!

# HTTP GET vs. POST requests

36

- For submitting data, a POST request is more appropriate than a GET
  - ▣ GET requests embed their parameters in their URLs
  - ▣ URLs are limited in length (~ 1024 characters)
  - ▣ URLs cannot contain special characters without encoding
  - ▣ private data in a URL can be seen or modified by users
- POST is generally preferred over GET. However, for EECS1012, we will use GET, since it helps us find errors by looking at the URL

# Form POST example

37

```
<form action="http://localhost/app.php" method="post">
<div>
  Name: <input type="text" name="name"> <br>
  Food: <input type="text" name="meal"> <br>
  <label>Meat? <input type="checkbox"
name="meat"><label> <br>
  <input type="submit">
</div>
</form>
```

HTML

- Very easy to change, modify `<form action>` to include `method="post"`

# A fun "trick" to hack web programs

38

- If you realize a web program is printing out strings exactly as they are passed, you can do some interesting tricks – esp with GET is used.

**Recall our form generated the following URL:**

`http://localhost/printForm.php?userid=Honest+Mike&ccard_number=4545500043334888&creditcard=visa`

What if we passed the following, instead of "Honest Mike", we passed the string:  
'<a href="http://dogecoin.com/"> Honest Mike </a>'

This can be done by using services to convert strings into formatted URLs  
(<https://meyerweb.com/eric/tools/dencoder/>)

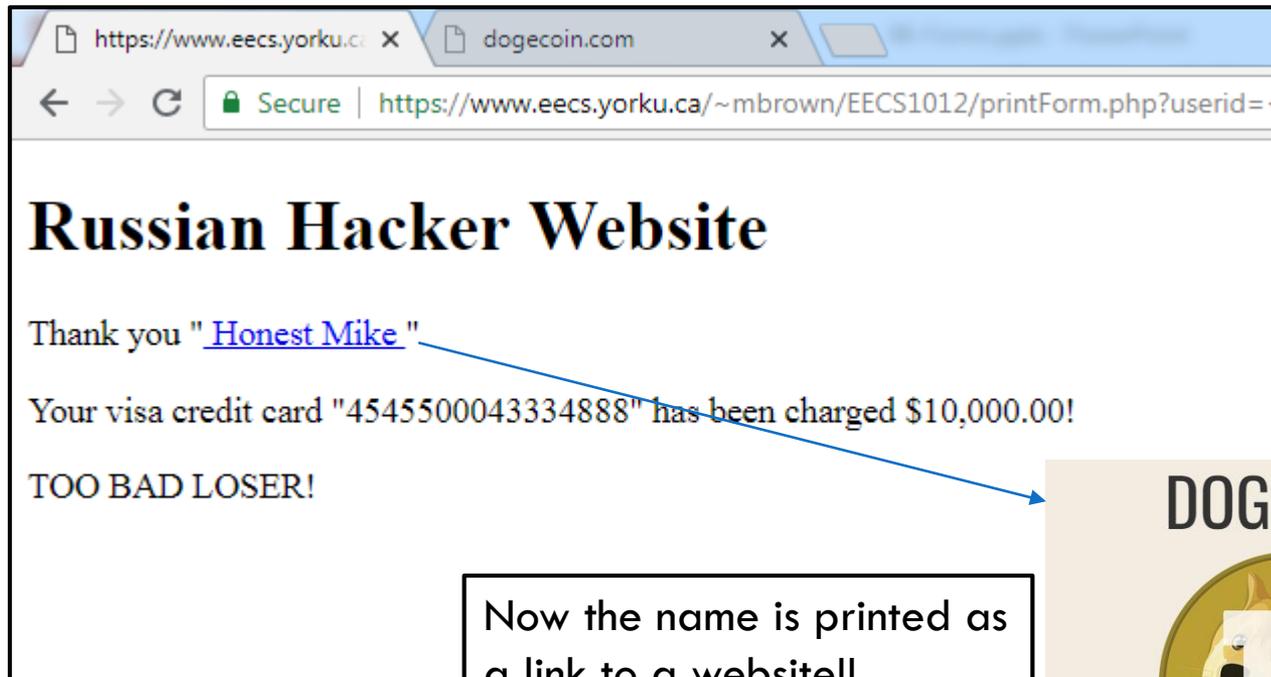
<a href="http://dogecoin.com/"> Honest Mike </a> is encoded as:  
%3Ca%20href%3D%22http%3A%2F%2Fdogecoin.com%2F%22%3E%20Honest%20Mike%20%3C%2Fa%3E%0A

# Applying our trick

39

The value for "userid" is now the text for an html link element. This webserver displays this on the webpage.

`https://www.eecs.yorku.ca/~mbrown/EECS1012/printForm.php?userid=%3Ca%20href%3D%22http%3A%2F%2Fdogecoin.com%2F%22%3E%20Honest%20Mike%20%3C%2Fa%3E%0A&ccard_number=4545500043334888&creditcard=visa`



Now the name is printed as a link to a website!!

