



Web programming

The HTML language

The HTML language

- ▷ Basic concepts
- ▷ User interfaces in HTML
 - Forms
 - Tables
- ▷ Passing parameters stored in forms

Basic concepts

▷ HTML: HyperText Markup Language

▷ Standard “de facto”

- W3C: World Wide Web Consortium
- <http://www.w3.org/>

▷ Evolving

- Born in 1991
- HTML, HTML 2, HTML 4, XHTML 1, HTML 5

▷ Objective: providing a structured description of an hypertextual document that is independent from the programs

▷ Purely textual: it is based on the first 127 characters of ASCII code

Basic concepts

- ▷ HTML allows to annotate a text to mark parts that compose it
 - Annotations are implemented by "tags"
- ▷ Visualize HTML documents: browser
 - Browsers interpret tags to show the text in a proper way
 - Browsers ignore tags that are not known
- ▷ Write HTML documents: any text editor
 - Notepad, Notepad++
 - Visual studio code
 - Sublime text
 - Microsoft WORD is not a code editor

Tags

- ▷ They are expressions contained between the two symbols of minor (<) and major (>)
- ▷ Usually text portions are delimited by a pair of tags (e.g.: <h1>Title</h1>)
- ▷ The general rule is that the final tag is the same of the initial one, preceded by the symbol "/"
- ▷ "Empty" tags, that are not applied to text portions, are of type

Attributes

- ▷ Using attributes it is possible to better characterize a tag
- ▷ Example: to insert an image in the center, to the left, to the right, to specify the text color, to specify dimensions of a column of a table, ...
- ▷ Attributes are made of a variable and a particular value is assigned to it
 - e.g. `width="100"`

Structure of an HTML document

```
01 <html>
02
03 <head>
04   <title>Text example</title>
05   ...
06 </head>
07
08 <body bgcolor="#FFFFFF">
09   ...
10   ...
11 </body>
12
13 </html>
```



Header



Body

- ▷ Only the content inside the `<body>` section is displayed in a browser.

Struttura di un documento HTML

- ▷ Header: it contains information on the page
 - Title <title>
 - Metadata <meta>
 - Others...
- ▷ Body: it includes the visible part of the HTML page
 - Text (paragraphs, lists, ...)
 - Structures (tables, forms)
 - Others...
- ▷ Note - on modern web pages:
 - The style of visible HTML elements in the page is given through **Style Sheets (css)**

HTML expressive power

▷ What does it allow to do?

- To create static web pages

▷ What does NOT it permit to do?

- To create dynamic web pages

▷ Dynamic web pages

- They are created "on the fly" to answer to the user input

I

| Courses schedule | Time schedule Search | | | | | | | | | | | | | | |
|--|---|---------------|--|-------|-----------|-----------|--------|--------------|-------------------|-------|-----------------------|------|--|---------------------------------------|--|
| <ul style="list-style-type: none">▶ Time Schedule search<ul style="list-style-type: none">▶ Consultazione generale▶ Advanced search▶ Free classrooms by slot▶ Free classrooms map | <table border="1"><thead><tr><th colspan="2">Search Filter</th></tr></thead><tbody><tr><td>Year:</td><td>2016/2017</td></tr><tr><td>Location:</td><td>TORINO</td></tr><tr><td>Degree type:</td><td>Master of Science</td></tr><tr><td>Area:</td><td>School of Engineering</td></tr><tr><td>Cdl:</td><td>COMPUTER AND COMMUNICATION NETWORKS ENGINEERING TELECOMI</td></tr><tr><td colspan="2"><input type="button" value="Search"/></td></tr></tbody></table> | Search Filter | | Year: | 2016/2017 | Location: | TORINO | Degree type: | Master of Science | Area: | School of Engineering | Cdl: | COMPUTER AND COMMUNICATION NETWORKS ENGINEERING TELECOMI | <input type="button" value="Search"/> | |
| Search Filter | | | | | | | | | | | | | | | |
| Year: | 2016/2017 | | | | | | | | | | | | | | |
| Location: | TORINO | | | | | | | | | | | | | | |
| Degree type: | Master of Science | | | | | | | | | | | | | | |
| Area: | School of Engineering | | | | | | | | | | | | | | |
| Cdl: | COMPUTER AND COMMUNICATION NETWORKS ENGINEERING TELECOMI | | | | | | | | | | | | | | |
| <input type="button" value="Search"/> | | | | | | | | | | | | | | | |

Dynamic web pages

Search Filter

| | |
|--------------|---------------------------------|
| Year: | 2016/2017 ▼ |
| Location: | TORINO |
| Degree type: | Master of Science |
| Area: | School of Engineering |
| Cdl: | COMPUTER AND COMMUNICATION NETW |

User interface
(static page, HTML)

Query result
(dynamic page)

| Daily view | | Monday 27/03/2017 | Tuesday 28/03/2017 | Wednesday 29/03/2017 |
|------------------|--|--|--|---|
| 8 ⁰⁰ | | | | |
| 9 ⁰⁰ | | Linear Algebra and Geometry GATTO LETTERIO AA - BARB - 1 | Physics I AGNELLO MICHELANGELO AAA - BARB - 1 | Linear Algebra and Geometry GATTO LETTERIO AA - BARB - 1 |
| 10 ⁰⁰ | 4 | Lezione/Esercitazione | | Physics I AGNELLO MICHELANGELO AAA - BARB - 1 |
| 11 ⁰⁰ | | | | Linear Algebra and Geometry GATTO LETTERIO AA - BARB - 1 |
| 12 ⁰⁰ | 4 | Physics I AGNELLO MICHELANGELO AAA - BARB - 1 | | |
| 13 ⁰⁰ | | | | |
| 14 ⁰⁰ | | | | |
| 15 ⁰⁰ | | | | |
| 16 ⁰⁰ | Mathematical analysis II - ... NICOLA FABIO AA - ZZ - 0 29B | Mathematical analysis I tut... TABACCO ANITA MARIA AA - ZZ - 0 15 | Mathematical analysis I tut... TABACCO ANITA MARIA AA - ZZ - 0 55 | Mathematical analysis I tut... TABACCO ANITA MARIA AA - ZZ - 0 8C Lezione/Esercitazione SQUADRA C |
| 17 ⁰⁰ | | | | |
| 18 ⁰⁰ | | | | |

▷ HTML allows to...

- Insert and format text (included bulleted list, numbered list, ...)
- Check colours, font, backgrounds, ... (Use CSS instead)
- Insert images, audio, video
- Insert hypertextual links
- Insert tables
- Insert forms
- ...

Example

Lesson 9: images

Wouldn't it be nice to be able to add a Tim Berners-Lee's picture, the HTML inventor, in the center of your page?

It sounds as a difficult task...

Maybe, instead it is extremely easy to do. All you need is an element:

Example 1

```

```

On your browser it will appear in this way



Example

```
<html>
  <head>
    <title>Example</title>
    <meta http-equiv="content-type" content="text/html; charset=iso-8859-1" />
    <meta name="generator" content="HAPedit 3.1">
  </head>

  <body bgcolor="Lavander">
    <font face="Arial">
      <h1>Lesson 9: images</h1> Tag: Title
      <p>Wouldn't it be nice to be able to add a Tim Bernes-Lee's picture, the
         HTML inventor, in the center of your page? </p> Tag: Paragraph
      <h2>It sounds as a difficult task...</h2>
      <p>Maybe, instead it is extremely easy to do.
         All you need is an element:</p>
      <p><strong>Example 1:</strong></p>
      <pre>
        &lt;img src="tim.jpg" alt="Tim" /&gt;
      </pre>
      <p>On your browser it will appear in this way</p>
       Tag: Image
    </font>
  </body>
</html>
```

ader

ody

D B M G

Example

Otherwise you can download images from other web sites. But be careful to not violate copyrights when you download them, please. It is good to know how you can download images, so have a look here to learn it:

1. Right click on any image that you can find on Internet.
2. Choose “Save image as ...” from the menu that appears.
3. Choose a location on your computer where you want to put the image and press “Save”.

Partially adapted from: [HTML.net](#)

Example

<**p**>Otherwise you can download images from other web sites. But be careful to not violate copyrights when you download them, please. It is good to know how you can download images, so have a look here to learn it:</**p**>

```
<ol>
  <li>Right click on any image that you can find on Internet.
</li>
  <li>Choose "Save image as ..." from the menu that appears. </li>
  <li>Choose a location on your computer where you want to put the image and press "Save".</li>
</ol>

<p>Partially adapted from:
<a href="http://www.html.net">HTML.net</a></p>

</font>
</body>

</html>
```

HTML page layout structure

▷ Solution 1: nested tables

- Old fashioned way



▷ Solution 2: use <div> tag

- Each `<div>` tag has an id or class
- The style for each id/class is given through CSS



HTML 4 VS HTML 5

- ▷ HTML5 offers new semantic elements to define different parts of a web page

HTML 4

```
<div id="header">
```

```
<div id="nav">
```

```
<div class="article">
```

```
<div class="section">
```

```
<div id="sidebar">
```

```
<div id="footer">
```

HTML 5

```
<header>
```

```
<nav>
```

```
<article>
```

```
<section>
```

```
<aside>
```

```
<footer>
```

Our mission

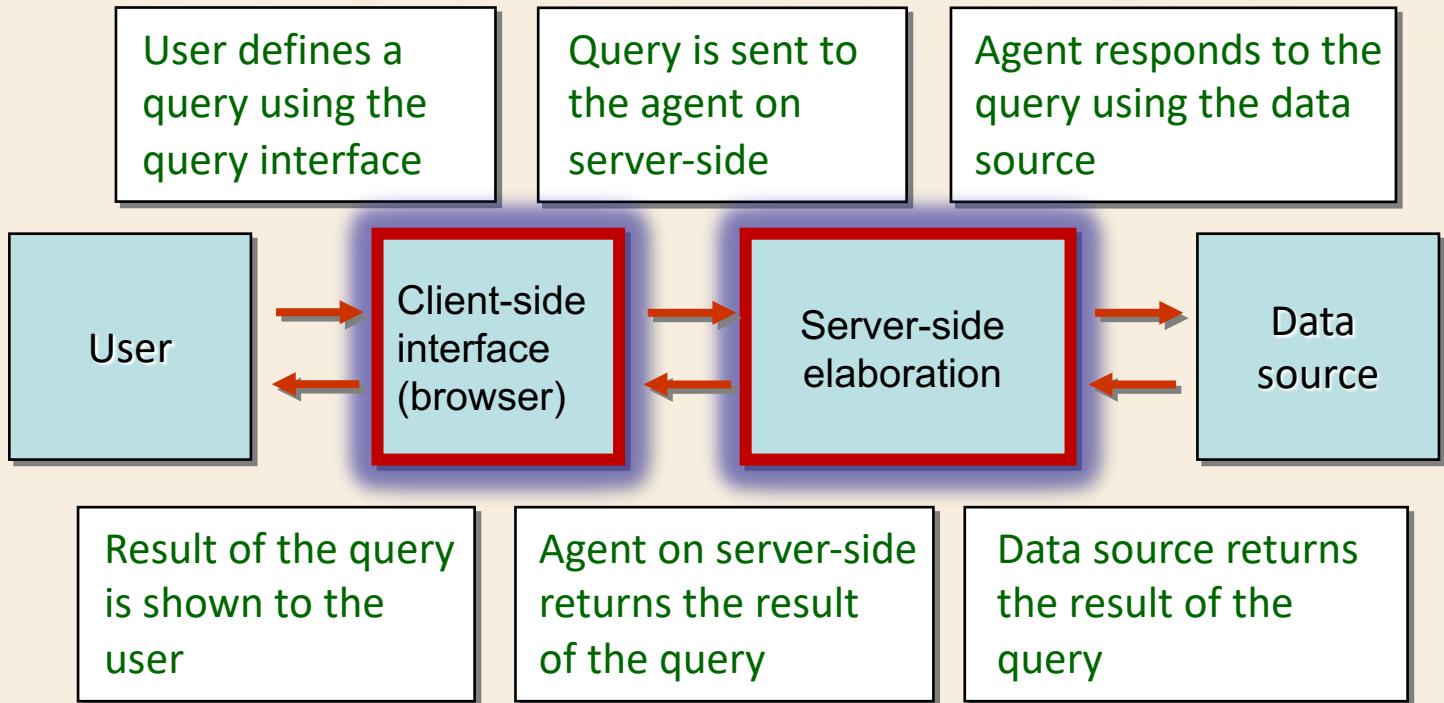
▷ Teaching HTML? Not exactly

- Many resources available
- Online courses, e.g.
<https://www.w3schools.com/html>
- Tags and attributes list, e.g.
<http://www.htmldog.com/reference/htmltags/>

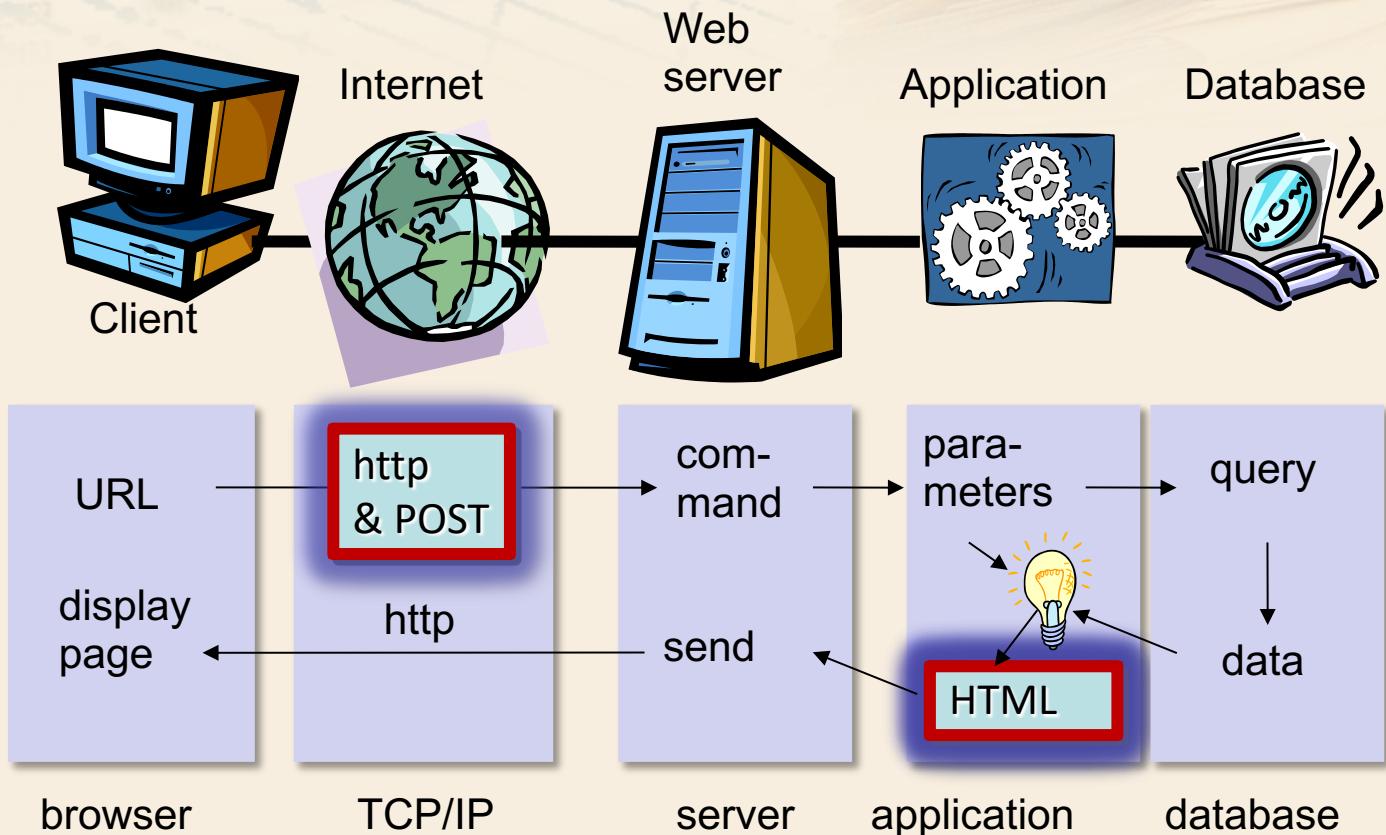
▷ Mission: to teach you to interact with a database via web

- Create the user interface to pass data to the queries
- Visualize an HTML document that contains queries results

HTML in web application workflow



Transaction on a database



User interfaces

- ▷ Allow to send data to programs that process them
 - User's choices
- ▷ In HTML they use
 - Forms
 - Tables

The screenshot shows a web application for searching course schedules. On the left, there is a sidebar with a blue header "Courses schedule" containing links: "Time Schedule search", "Consultazione generale", "Advanced search", "Free classrooms by slot", and "Free classrooms map". The main area has a header "Time schedule Search". Below the header is a "Search Filter" section with five dropdown menus:

- Year: 2016/2017
- Location: TORINO
- Degree type: Master of Science
- Area: School of Engineering
- Cdl: COMPUTER AND COMMUNICATION NETWORKS ENGINEERING TELECOMI

A green "Search" button is located at the bottom right of the filter section.

Forms

- ▷ Forms allow users to build services' graphic interface
 - To insert data that will be sent to a program that processes them and/or to a database
 - Text boxes, buttons, drop down menu, ...
- ▷ HTML allows to create the interface, but it doesn't allow to process inserted data
 - It is explicitly shown, in the form, the name of the program that will use data
 - Need to use other programming languages, e.g. PHP, Java, Python...

Form example

| | Item | Image | Size | Quantity | Cost |
|-------------------------------------|--------------------|--|----------|--------------------------------|---------|
| <input checked="" type="checkbox"/> | Orange t-shirt |  | small ▾ | <input type="text" value="1"/> | 61.00 € |
| <input type="checkbox"/> | Blue sweater |  | medium ▾ | <input type="text" value="0"/> | 70.20 € |
| <input checked="" type="checkbox"/> | Blue-striped shirt |  | medium ▾ | <input type="text" value="3"/> | 25.00 € |
| <input checked="" type="checkbox"/> | Gym suit |  | large ▾ | <input type="text" value="2"/> | 45.70 € |
| <input type="checkbox"/> | Grey trousers |  | medium ▾ | <input type="text" value="0"/> | 53.50 € |

Payment method:

- Cash
- Prepaid card
- Credit card (fee of 2,50€)

Input elements

- Interactive
- Not interactive

Form creation

```
<form name="userData" action="respondPage.php" method="GET">  
    Input elements  
</form>
```

▷ "Form" tag with some attributes

- Name: form name
- Action: name of the program that will process form's data
- Method: the way in which parameters are passed from the form to the program (it can be "GET" or "POST")

▷ There are more input elements in the form

Form example

| Item | Image | Size | Quantity | Cost |
|--|--|--------------------------------------|--------------------------------|---------|
| <input checked="" type="checkbox"/> Orange t-shirt |  | small ▾ | <input type="text" value="1"/> | 61.00 € |
| <input type="checkbox"/> Blue sweater |  | medium ▾ | <input type="text" value="0"/> | 70.20 € |
| <input checked="" type="checkbox"/> Blue-striped shirt |  | medium ▾ small medium large | <input type="text" value="3"/> | 25.00 € |
| <input checked="" type="checkbox"/> Gym suit |  | large ▾ | <input type="text" value="2"/> | 45.70 € |
| <input type="checkbox"/> Grey trousers |  | medium ▾ | <input type="text" value="0"/> | 53.50 € |

Payment method:

- Cash
- Prepaid card
- Credit card (fee of 2,50€)

Input elements

- Text field
- Checkbox
- Radio button
- “Submit” button
- “Reset” button
- ...
- Text
- Images

Input elements

▷ General structure (with few exceptions)

```
<input type="text" name="address" size="30"  
       value="type here your address" />
```

▷ "Input" tag with some attributes

- type: element type
- name: element name
- value: value that will be passed to the program that processes user request
- Other attributes that are specific for element types (e.g. size for "text" type)

Input elements example

| | | | | | |
|-------------------------------------|--------------------|---|--|--------------------------------|---------|
| <input checked="" type="checkbox"/> | Blue-striped shirt |  | <select name="dim3"><option value="1">small</option><option selected="" value="2">medium</option><option value="3">large</option></select> | <input type="text" value="3"/> | 25.00 € |
| | |  | | | |

```
<input type="checkbox" name="item3" value="1" />
```

```
<input type="image" name="shirt" src="./img/stripped_shirt.jpg" height="80"/>
```

```
<select name="dim3">
  <option value="1">small</option>
  <option value="2" selected>medium</option>
  <option value="3">large</option>
</select>
```

```
<input type="text" name="q3" value="0" size="2" />
```

Input elements example

Payment method:

- Cash
- Prepaid card
- Credit card (fee of 2,50€)

Selected element when the page is loaded

```
<input type="radio" name="pag" value="0" checked="" />Cash<br />
<input type="radio" name="pag" value="1" />Prepaid card<br />
<input type="radio" name="pag" value="2" />Credit card (fee of
2,50€)<br />
```

Important: same name

Send order Cancel

```
<input type="submit" name="send" value="Send order" />
<input type="reset" name="cancel" value="Cancel" />
```

Input elements

- Submit button: perform a call to a processing program
 - `input type="submit"`
- Reset button: resets all form data
 - `input type="reset"`
- Text field
 - `input type="text"`
- Checkbox
 - `input type="checkbox"`
- Radio button
 - `input type="radio"`
- Image
 - `input type="image"`

Input elements

- ▷ Password field: it shows stars or dots instead of characters

- `input type="password"`

```
<input type="password" maxlength="8" size="18" name="passwd" />
```

- ▷ Textarea field

- `textarea`

Here you can write your text

```
<textarea name="text" rows="5" cols="40">  
Here you can write your text  
</textarea>
```

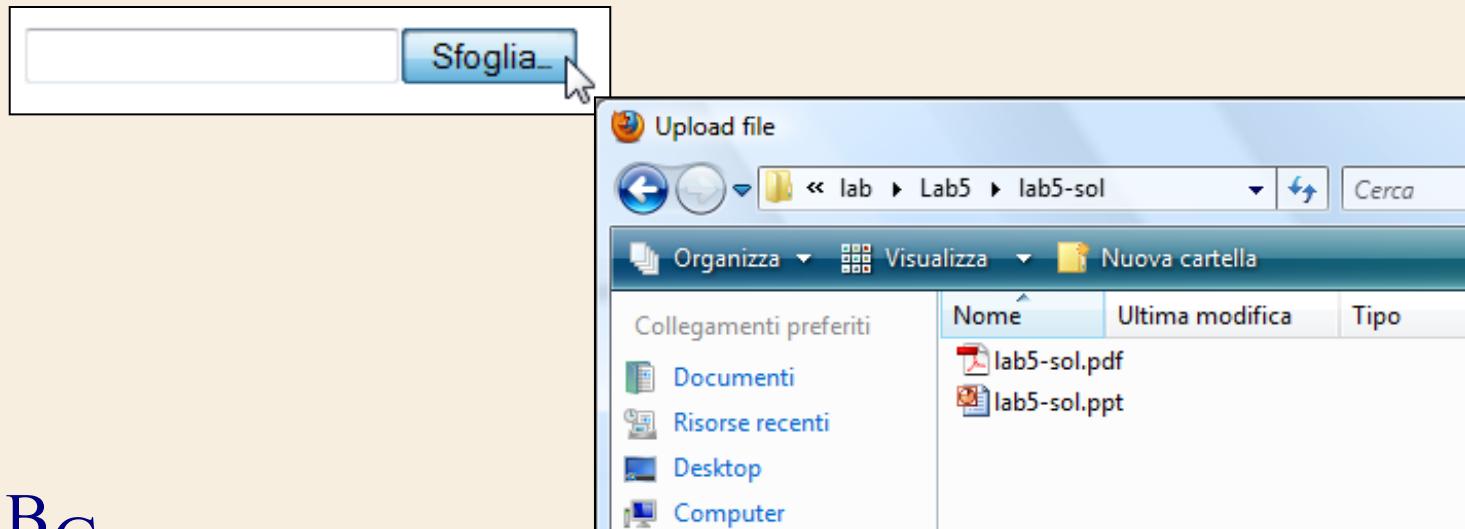
- ▷ Menu of options

- `select`

Input elements

File

```
<input name="userFile" type="file" size="20" />
```



▷ Base structure

- Rows (table row)
- Cells (table data)

| | |
|------------|-------------|
| First cell | Second cell |
| Third cell | Fourth cell |

```
<table border="1">
  <tr>
    <td>First cell</td>
    <td>Second cell</td>
  </tr>
  <tr>
    <td>Third cell</td>
    <td>Fourth cell</td>
  </tr>
</table>
```

Tables

- ▷ It is possible to define columns width

| | |
|------------|-------------|
| First cell | Second cell |
| Third cell | Fourth cell |

```
<table border="1">
  <tr>
    <td width="100">First cell</td>
    <td width="100">Second cell</td>
  </tr>
  <tr>
    <td>Third cell</td>
    <td>Fourth cell</td>
  </tr>
</table>
```

Tables

- ▷ It is possible to define a row header
 - Table header

| CodP | NameP | Color | Size | Storage |
|------|---------|-------|------|---------|
| P1 | Sweater | Red | 40 | Turin |
| P2 | Jeans | Green | 48 | Milan |
| P3 | Shirt | Blue | 48 | Rome |
| P4 | Shirt | Blue | 44 | Turin |
| P5 | Skirt | Blue | 40 | Milan |
| P6 | Shorts | Red | 42 | Turin |

- ▷ It is possible to define a row header
 - Table header

```
<table width="50%" border="1">
  <tr>
    <th width="15%">CodP</th>
    <th width="25%">NameP</th>
    <th width="20%">Color</th>
    <th width="15%">Size</th>
    <th width="25%">Storage</th>
  </tr>
  <tr>
    <td>P1</td>
    <td>Sweater</td>
    <td>Red</td>
    <td>40</td>
    <td>Turin</td>
  </tr>
  . . . . .
  <tr>
    <td>P6</td>
    <td>Shorts</td>
    <td>Red</td>
    <td>42</td>
    <td>Turin</td>
  </tr>
</table>
```

Tables and forms

- Tables are often used with forms for “aesthetic” purposes

| | Item | Image | Size | Quantity | Cost |
|--------------------------|----------------|---|----------|--------------------------------|---------|
| <input type="checkbox"/> | Orange t-shirt |  | medium ▾ | <input type="text" value="0"/> | 61.00 € |
| <input type="checkbox"/> | Blue sweater |  | medium ▾ | <input type="text" value="0"/> | 70.20 € |

Tables and forms – Example 1

```
<form name="myForm" action="order.php" method="GET">
<table border="1">
<tr>
  <th width="30"></th>
  <th width="100">Item</th>
  <th width="100">Image</th>
  <th width="100">Size</th>
  <th width="100">Quantity</th>
  <th width="100">Cost</th>
</tr>
<tr>
  <td align="center"><input type="checkbox" name="item1" value="1" /></td>
  <td>Orange t-shirt</td>
  <td align="center"><input type="image" name="orange" src=".//img/orange_t-shirt.jpg" height="80" /></td>
  <td>
    <select name="dim1">
      <option value="1">small</option>
      <option value="2" selected>medium</option>
      <option value="3">large</option>
    </select>
  </td>
  <td align="center"><input type="text" name="q1" value="0" size="2" /></td>
  <td align="center">61.00 €</td>
</tr>
. . . . . . . . . . . . . . .
</table>
```

Tables and forms – Example 2

Payment method:

- Cash
- Prepaid card
- Credit card (fee of 2,50€)

Send order

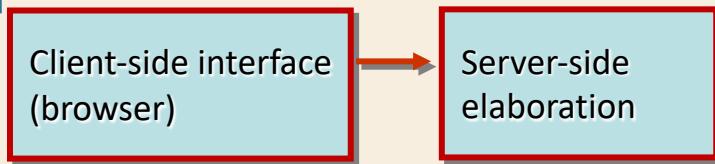
Cancel

Payment method:

```
<table border="0">
  <tr>
    <td width="300">
      <input type="radio" name="pag" value="0" checked />Cash<br />
      <input type="radio" name="pag" value="1" />Prepaid card<br />
      <input type="radio" name="pag" value="2" />Credit card (fee of
        2,50€)<br />
    </td>
    <td>
      <input type="submit" name="send" value="Send order" />
      <input type="reset" name="cancel" value="Cancel" />
    </td>
  </tr>
</table>
```

Parameters passing

- ▷ In addition to ask a page to the web server, form allows to specify some parameters too, that will be used by the script
 - Example: selected products, size, quantity, payment method



- ▷ Two submission methods: GET and POST

```
<form name="userData" action="responsePage.php" method="GET">  
    Input elements  
</form>
```

Parameter passing

▷ GET method

- It allows to append data to the address of the requested page, by following the page name by a question mark and by name/value pairs of data in which we are interested
- Name and value are separated by the equal sign
- Different name/value pairs are separated by '&'

GET method example

Set data

| | |
|-----------|--|
| Congress: | ICSE |
| Year: | 2006 ▾ |
| Articles: | <input type="radio"/> 1 <input checked="" type="radio"/> 2 <input type="radio"/> 3 |

```
<form method="get" action="test.php">
  <table>
    <tr>
      <td> Congress: </td>
      <td><input type="text" name="num" size="20"></td>
    </tr>
    <tr>
      <td> Year: </td>
      <td>
        <select name="year">
          <option value="2005">2005</option>
          <option value="2006">2006</option>
        </select>
      </td>
    </tr>
    <tr>
      <td> Articles: </td>
      <td>
        <input type="radio" name="number" value="1"> 1
        <input type="radio" name="number" value="2" checked> 2
        <input type="radio" name="number" value="3"> 3
      </td>
    </tr>
  </table>
  <br />
  <input type="reset" value="Cancel">
  <input type="submit" value="Send">
</form>
```

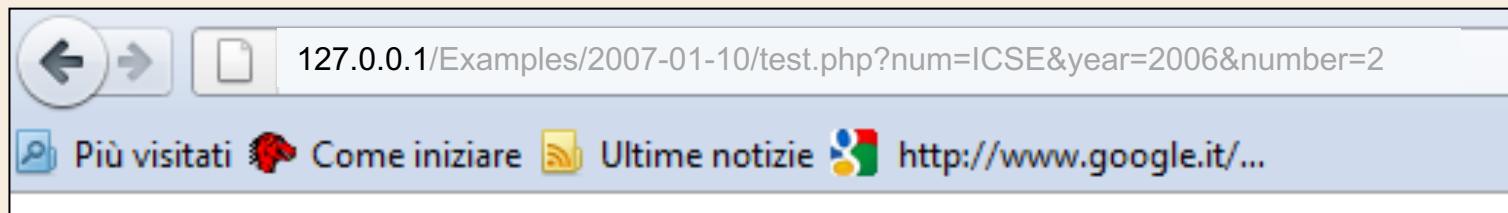
GET method example

Set data

Congress:

Year:

Articles: 1 2 3



Parameter passing

▷ GET method

- Some servers have some limitations about the GET method, and they don't allow to send forms having values greater than 255 total characters
- It is particularly suitable for forms having few fields and few data to send

```
http://127.0.0.1/Examples/order.php?art1=1&dim1=1&q1=1&dim2=2&q2=0&item3=1&dim3=1&q3=3&item4=1&dim4=3&q4=2&dim5=2&q5=0&pag=1&send=Send+1%27order
```

Parameter passing

▷ POST method

- Sending data is done in two phases: first the page on the server that has to compute data is contacted, then data are sent
- For this reason parameters don't appear in the query string and are not directly visible by the user
- Useful in many cases (e.g. password field)
- No limits on length of characters
- Even binary data can be sent over POST as well as textual data

POST method example

Set data

| | |
|-----------|--|
| Congress: | ICSE |
| Year: | 2006 ▾ |
| Articles: | <input type="radio"/> 1 <input checked="" type="radio"/> 2 <input type="radio"/> 3 |

```
<form method="post" action="test.php">
<table>
<tr>
    <td> Congress: </td>
    <td><input type="text" name="num" size="20"></td>
</tr>
<tr>
    <td> Year: </td>
    <td>
        <select name="year">
            <option value="2005">2005</option>
            <option value="2006">2006</option>
        </select>
    </td>
</tr>
<tr>
    <td> Articles: </td>
    <td>
        <input type="radio" name="number" value="1"> 1
        <input type="radio" name="number" value="2" checked> 2
        <input type="radio" name="number" value="3"> 3
    </td>
</tr>
</table>
<br />
<input type="reset" value="Cancel">
<input type="submit" value="Send">
</form>
```

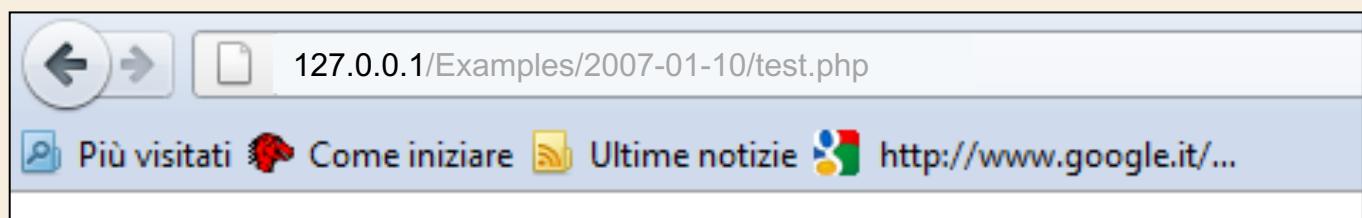
POST method example

Set data

Congress:

Year:

Articles: 1 2 3



GET VS POST

- ▷ Since the data sent using the GET method is appended directly to the URL, this solution IS NOT recommended for sensitive data (e.g. username and password)
- ▷ In case of a high number of parameters the adoption of GET method IS NOT recommended
- ▷ Some tags can be used only with the POST method (e.g. <input type="file">)
- ▷ The GET method can be helpful during the development of a web application