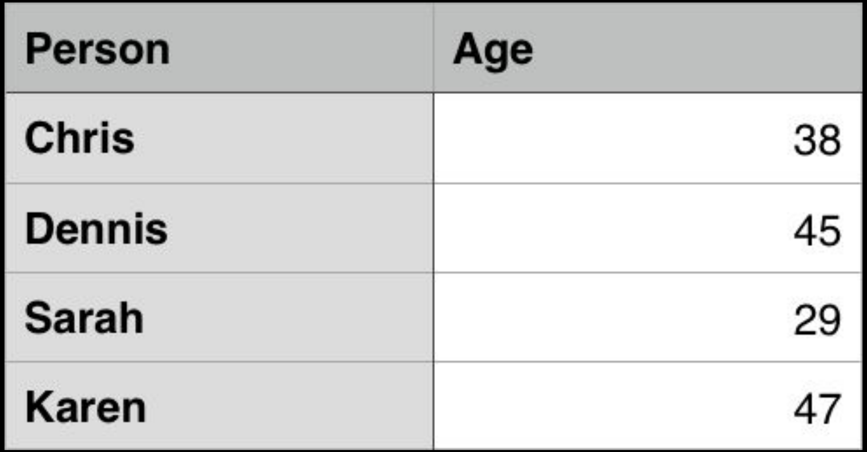
# HTML Tables

# What is a table?

- A table is a structured set of data made up of rows and columns.
- HTML tables allows the developer to arrange the data into rows and columns.
- A table allows the user to quickly and easily look up values that indicate some kind of connection between different types of data.
- Example -



| Person | Age |
|---|---|
| Chris | 38 |
| Dennis | 45 |
| Sarah | 29 |
| Karen | 47 |

# How does a table work?

- The point of a table is that it is rigid.
- Information is easily interpreted by making visual associations between row and column headers.
- When done correctly, a successful HTML table should enhance the experience of the users.

# When should you NOT use HTML tables?

- HTML tables should be used for tabular data — this is what they are designed for.
- Unfortunately, a lot of people used to use HTML tables to lay out web pages, e.g. one row to contain the header, one row to contain the content columns, one row to contain the footer, etc.
- In short, using tables for layout rather than CSS layout techniques is a bad idea.

# Creating a table

- The content of every table is enclosed by these two tags:

```
<table></table>
```

- The smallest container inside a table is a table cell, which is created by a <td> element ('td' stands for 'table data'):

```
<td>Hi, I'm your first cell.</td>
```

- If we want a row of four cells, we need to copy these tags three times. Update the contents of your table to look like so:

```
<td>Hi, I'm your first cell.</td>

<td>I'm your second cell.</td>

<td>I'm your third cell.</td>

<td>I'm your fourth cell.</td>
```

- The cells are not placed underneath each other, rather they are automatically aligned with each other on the same row. Each <td> element creates a single cell and together they make up the first row. Every cell we add makes the row grow longer.

- To stop this row from growing and start placing subsequent cells on a second row, we need to use the <tr> element ('tr' stands for 'table row').
- Place the four cells you've already created inside <tr> tags, like so:

```
<tr>
<td>Hi, I'm your first cell.</td>
<td>I'm your second cell.</td>
<td>I'm your third cell.</td>
<td>I'm your fourth cell.</td>
</tr>
```

# Adding table headers

- Headers are special cells that go at the start of a row or column and define the type of data that row or column contains.
- It is easier to find the data you are looking for when the headers clearly stand out, and the design just generally looks better.
- *Note - Table headings come with some default styling — they are bold and centered even if you don't add your own styling to the table, to help them stand out.*

# Allowing cells to span multiple rows and columns

- Sometimes we want cells to span multiple rows or columns.
- Take the following simple example, which shows the names of common animals. In some cases, we want to show the names of the males and females next to the animal name. Sometimes we don't, and in such cases we just want the animal name to span the whole table.
- Table headers and cells have the colspan and rowspan attributes, which allow the table entry to span across multiple rows or columns.
- Both accept a unitless number value, which equals the number of rows or columns you want spanned. For example, colspan="2" makes a cell span two columns.

# Providing common styling to columns

- HTML has a method of defining styling information for an entire column of data all in one place — the <col> and <colgroup> elements.
- These exist because it can be a bit annoying and inefficient having to specify styling on columns — you generally have to specify your styling information on every <td> or <th> in the column, or use a complex selector such as: nth-child.
- *Note - Styling columns like this is limited to a few properties: border, background, width, and visibility. To set other properties you'll have to either style every <td> or <th> in the column, or use a complex selector such as: nth-child.*