

COMPUTER PROGRAMMING (ICT)

Grade
11

QUARTER 2

MODULE

7

Inserting Box in HTML Document



0100001101101111011011010111000001110101
0111010001100101011100100010000001010000
0111001001101111011001110111001001100001
0110110101101101011010010110111001100111

Computer Programming (ICT) – Grade 11**Quarter 2 – Module 7: Inserting Box in HTML Documents.**

Republic Act 8293, section 176 states that: No copyright shall subsist in any work of the Government of the Philippines. However, prior approval of the government agency or office wherein the work is created shall be necessary for exploitation of such work for profit. Such agency or office may, among other things, impose as a condition the payment of royalties.

Borrowed materials (i.e., songs, stories, poems, pictures, photos, brand names, trademarks, etc.) included in this module are owned by their respective copyright holders. Every effort has been exerted to locate and seek permission to use these materials from their respective copyright owners. The publisher and authors do not represent nor claim ownership over them.

Published by the Department of Education Division of Pasig City

Development Team of the Module

Writer : Magiel L. Boncayao

Editor : Ma. Lerma Cantanero

Reviewer : Rowena O. Dimagiba

Illustrator:

Layout Artist:

Management Team: Ma. Evalou Concepcion A. Agustin
OIC-Schools Division Superintendent
Dr. Aurelio G. Alfonso
OIC-Assistant Schools Division Superintendent
Dr. Victor Javena
OIC – Chief Curriculum Implementation Division

Education Program Supervisors

Librada L. Agon EdD (EPP/TLE/TVL/TVE)
Liza A. Alvarez (Science/STEM/SSP)
Bernard R. Balitao (AP/HUMSS)
Joselito E. Calios (English/SPFL/GAS)
Norlyn D. Conde EdD (MAPEH/SPA/SPS/HOPE/A&D/Sports)
Wilma Q. Del Rosario (LRMS/ADM)
Ma. Teresita E. Herrera EdD (Filipino/GAS/Piling Larang)
Perlita M. Ignacio PhD (EsP)
Dulce O. Santos PhD (Kindergarten/MTB-MLE)
Teresita P. Tagulao EdD (Mathematics/ABM)

Printed in the Philippines by _____

Department of Education – Division of Pasig City

Office Address: Caruncho Avenue, San Nicolas, Pasig City

Telefax:

E-mail Address:



Introductory Message

For the Facilitator:

Welcome to the Computer Programming for the ICT Module on Inserting Box in HTML Documents

This module was collaboratively designed, developed and reviewed by educators from Schools Division Office of Pasig City headed by its Officer-In-Charge Schools Division Superintendent, Ma. Evalou Concepcion A. Agustin in partnership with the Local Government of Pasig through its mayor, Honorable Victor Ma. Regis N. Sotto. The writers utilized the standards set by the K to 12 Curriculum using the Most Essential Learning Competencies (MELC) while overcoming their personal, social, and economic constraints in schooling.

This learning material hopes to engage the learners into guided and independent learning activities at their own pace and time. Further, this also aims to help learners acquire the needed 21st century skills especially the 5 Cs namely: Communication, Collaboration, Creativity, Critical Thinking and Character while taking into consideration their needs and circumstances.

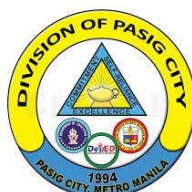
In addition to the material in the main text, you will also see this box in the body of the module:



Notes to the Teacher

This contains helpful tips or strategies that will help you in guiding the learners.

As a facilitator you are expected to orient the learners on how to use this module. You also need to keep track of the learners' progress while allowing them to manage their own learning. Moreover, you are expected to encourage and assist the learners as they do the tasks included in the module.



Computer Programming

11

Quarter 2

Self Learning Module 7

Inserting Box in HTML Documents

Writer: Magiel L. Boncayao

Editor: Ma. Lerma I. Cantanero

Validator/Reviewer: Rowena O. Dimagiba



For the Learner:

Welcome to the Computer Programming for the ICT Module on Inserting Box in HTML Documents

The hand is one of the most symbolized part of the human body. It is often used to depict skill, action and purpose. Through our hands we may learn, create and accomplish. Hence, the hand in this learning resource signifies that you as a learner is capable and empowered to successfully achieve the relevant competencies and skills at your own pace and time. Your academic success lies in your own hands!

This module was designed to provide you with fun and meaningful opportunities for guided and independent learning at your own pace and time. You will be enabled to process the contents of the learning material while being an active learner.

This module has the following parts and corresponding icons:



Expectation - These are what you will be able to know after completing the lessons in the module



Pre-test - This will measure your prior knowledge and the concepts to be mastered throughout the lesson.



Recap - This section will measure what learnings and skills that you understand from the previous lesson.



Lesson- This section will discuss the topic for this module.



Activities - This is a set of activities you will perform.



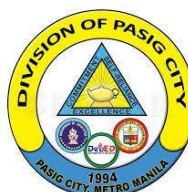
Wrap Up- This section summarizes the concepts and applications of the lessons.



Valuing- this part will check the integration of values in the learning competency.



Post-test - This will measure how much you have learned from the entire module. Ito po ang parts ng module.

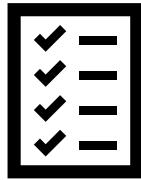




EXPECTATION

The students should be able to:

1. familiarize the different ways on how to put and position the box in a webpage
2. identify different ways on how to insert a box
3. apply the different ways on how to insert a box in making a webpage.

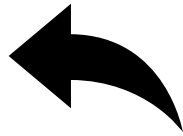


PRE-TEST

Instructions: Select the letter that corresponds to the correct answer.

1. This is a property that specifies the type of box used for an HTML element
A. Inline-flex B. Flex-container C. Flex-direction D. Display
2. It is used to ensures that the elements behave predictably when the page layout must accommodate different screen sizes and different display devices.
A. Flexbox B. Inline-flex C. Flex-direction D. Flex-container
3. Which of the following justify-content property value allows you to position the item at end of the container?
A. flex-end B. flex-start C. space-between D. space-around
4. Which of the following justify-content property value you to position the item with space between the lines?
A. flex-end B. flex-start C. space-between D. space-around
5. It is a property that specifies whether the flex items should wrap or not?
A. Flex-direction B. Inline-flex C. Flex-wrap D. Flex-container





RECAP

We discussed last time the CSS User interface. We learned that we can make our website more interesting by changing the appearance of our content using this kind of CSS properties.

In this activity you need to complete the syntax of the following to change appearance of your webpage by changing the resize to both, margin to 50px, outline-offset to 20px, border style dotted, border 5px and padding 30px.

```
<!DOCTYPE html>
<html>
<head>
<style>
div {
    margin: _____;
    border: _____;
    padding: _____;
    width: 300px;
    resize: _____;
    outline-offset: _____;
}
</style>
</head>
```



LESSON

CSS3 Flexbox

Flexible boxes, or flexbox, is a new layout mode in CSS3.

Use of flexbox ensures that elements behave predictably when the page layout must accommodate different screen sizes and different display devices.

CSS3 Flexbox Concepts

Flexbox consists of flex containers and flex items.

A flex container is declared by setting the display property of an element to either flex (rendered as a block) or inline-flex (rendered as inline).

Inside a flex container there is one or more flex items.

Note: Everything outside a flex container and inside a flex item is rendered as usual. Flexbox defines how flex items are laid out inside a flex container.



Flex items are positioned inside a flex container along a flex line. By default there is only one flex line per flex container.

The following example shows three flex items. They are positioned by default: along the horizontal flex line, from left to right:

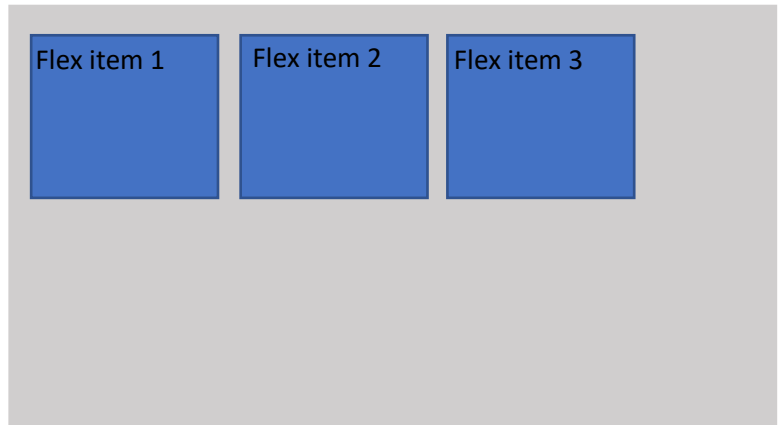
```
<!DOCTYPE html>
<html>
<head>
<style>
.flex-container {
  display: -webkit-flex;
  display: flex;
  width: 400px;
  height: 250px;
  background-color: lightgrey;
}

.flex-item {
  background-color: cornflowerblue;
  width: 100px;
  height: 100px;
  margin: 10px;
}
</style>
</head>
<body>

<div class="flex-container">
  <div class="flex-item">flex item 1</div>
  <div class="flex-item">flex item 2</div>
  <div class="flex-item">flex item 3</div>
</div>

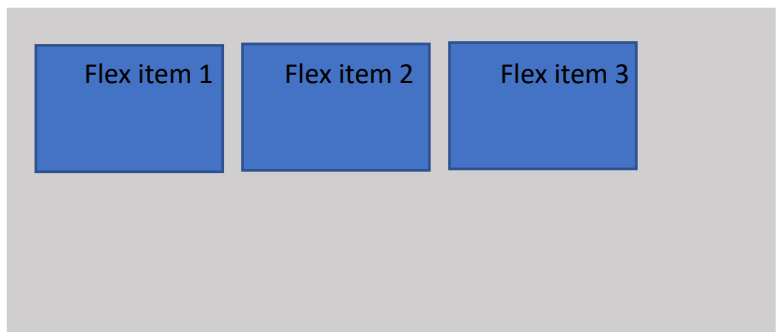
</body>
</html>
```

Results:



To change the direction of the flex line, just set the `direction` property to `rtl` (right-to-left), the text is drawn right to left, and also the flex line changes direction, which will change the page layout:

```
body {
  direction: rtl;
}
```



Flex Direction

The flex-direction property specifies the direction of the flexible items inside the flex container. The default value of flex-direction is row (left-to-right, top-to-bottom)

The other values are as follows:

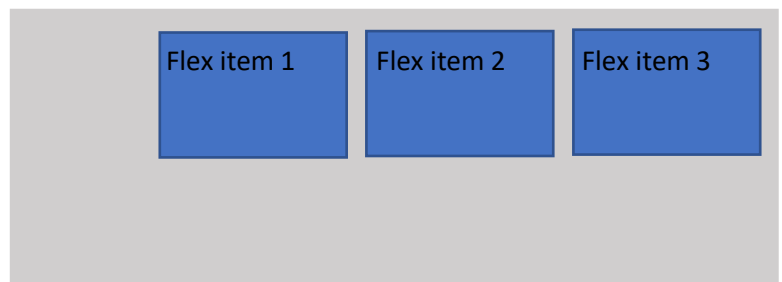
- **row-reverse** - If the writing-mode (direction) is left to right, the flex items will be laid out right to left
- **column** - If the writing system is horizontal, the flex items will be laid out vertically
- **column-reverse** - Same as column, but reversed

The following example shows the result of using the row-reverse value:

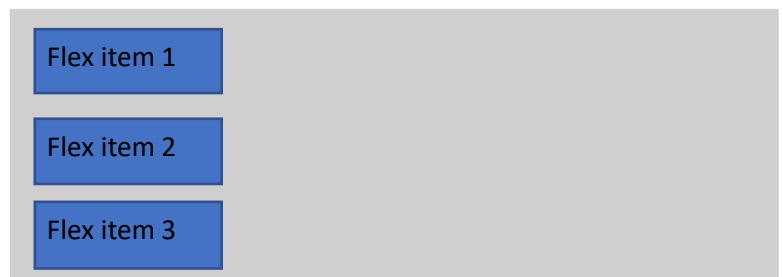
Syntax:

```
flex-direction: row-reverse;
```

Results:



```
flex-direction: column;
```



```
flex-direction: column-reverse;
```



The justify-content Property

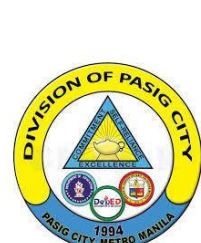
The justify-content property horizontally aligns the flexible container's items when the items do not use all available space on the main-axis.

The possible values are as follows:

- **flex-start** - Default value. Items are positioned at the beginning of the container
- **flex-end** - Items are positioned at the end of the container
- **center** - Items are positioned at the center of the container
- **space-between** - Items are positioned with space between the lines
- **space-around** - Items are positioned with space before, between, and after the lines

Here is the example syntax for **flex-start**

```
.flex-container {  
  display: flex;
```



```
display: flex;
justify-content: flex-end;
width: 400px;
height: 250px;
background-color: lightgrey;}
```

Here is the example syntax for **center**

```
.flex-container {
justify-content: center;
}
```

Here is the example syntax for **space-between**

```
.flex-container {
justify-content: space-between;
}
```

Here is the example syntax for **space-around**

```
.flex-container {
justify-content: space-around;
}
```

The flex-wrap Property

The flex-wrap property specifies whether the flex items should wrap or not, if there is not enough room for them on one flex line.

The possible values are as follows:

- **nowrap** - Default value. The flexible items will not wrap (this is the default for flex-wrap)
- **wrap** - The flexible items will wrap if necessary
- **wrap-reverse** - The flexible items will wrap, if necessary, in reverse order

Here is the example syntax for **nowrap**

```
.flex-container {
display: flex;
display: flex;
flex-wrap: nowrap;
width: 300px;
height: 250px;
background-color: lightgrey;
}
```

Here is the example syntax for **wrap**

```
.flex-container {
flex-wrap: wrap;
}
```

Here is the example syntax for **wrap-reverse**

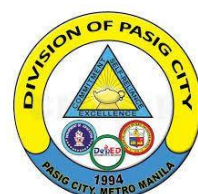
```
.flex-container {
flex-wrap: wrap-reverse;
}
```

Flex Item Properties

Ordering

The order property specifies the order of a flexible item relative to the rest of the flexible items inside the same container:

<style>



```

.first {
    order: -1;
}
</style>
<body>
<div class="flex-container">
    <div class="flex-item">flex item 1</div>
    <div class="flex-item first">flex item 2</div>
    <div class="flex-item">flex item 3</div>
</div>
</body>

```

Perfect Centering

In the following example we will solve an almost daily problem: perfect centering.

It is very easy with flexbox. Setting `margin: auto;` will make the item perfectly centered in both axis:

```

.flex-item {
    background-color: cornflowerblue;
    width: 75px;
    height: 75px;
    margin: auto;
}

```

align-self

The `align-self` property of flex items overrides the flex container's `align-items` property for that item. It has the same possible values as the `align-items` property.

The following example sets different `align-self` values to each flex item:

```

.flex-item {
    background-color: cornflowerblue;
    width: 60px;
    min-height: 100px;
    margin: 10px;
}

.item1 {
    align-self: flex-start;
}
.item2 {
    align-self: flex-end;
}

.item3 {

```



```
    align-self: center;
}
```

```
.item4 {
    align-self: baseline;
}
```

Flex Property

The flex property specifies the length of the flex item, relative to the rest of the flex items inside the same container

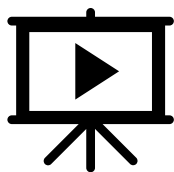
In the following example, the first flex item will consume 2/4 of the free space, and the other two flex items will consume 1/4 of the free space each:

```
.flex-item {
    background-color: cornflowerblue;
    margin: 10px;
}
```

```
.item1 {
    flex: 2;
}
```

```
.item2 {
    flex: 1;
}
```

```
.item3 {
    flex: 1;
}
```

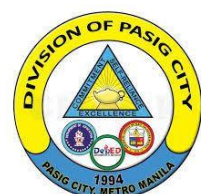


ACTIVITIES

CSS3 Flexbox Properties

The following table lists the CSS properties what you need to do is to explain the descriptions and function of each CSS Flexbox properties. For this activity the student will get 20 points

| Property | Description and Function |
|-----------------------|--------------------------|
| <u>display</u> | |
| <u>flex-direction</u> | |



justify-content

align-items

flex-wrap

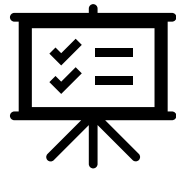
align-content

flex-flow

order

align-self

flex



WRAP-UP

In this module we discussed about flexbox to ensure that elements behave predictably when the page layout must accommodate different screen sizes and different display devices.

For this activity students will get 5 points if they give the correct answer. Make the box in center position and in a row reverse direction. Complete the following syntax below and put the answer on the space provided.

```
.flex-container {  
  display: flex;  
  _____;  
  width: 400px;  
  height: 250px;  
  background-color: lightgrey;  
  _____  
}
```





Instructions: Read carefully the following questions and provide two to three sentences answer for each number.

1. Why is it important to apply flexbox in your boxes?

2. What do you think are the advantages of having CSS flexbox, when making a website?



POST TEST

INSTRUCTIONS: Select the letter that corresponds to the correct answer.

1. It is a property that specifies whether the flex items should wrap or not?
a. A. Flex-direction B. Inline-flex C. Flex-wrap D. Flex-container
2. It is used to ensure that the elements behave predictably when the page layout must accommodate different screen sizes and different display devices.
a. A. Flexbox B. Inline-flex C. Flex-direction D. Flex-container
3. Which of the following justify-content property value allows you to position the item with space between the lines?
a. A. flex-end B. flex-start C. space-between D. space-around
4. This is a property that specifies the type of box used for an HTML element
a. A. Inline-flex B. Flex-container C. Flex-direction D. Display
5. Which of the following justify-content property value allows you to position the item at end of the container?
a. A. flex-end B. flex-start C. space-between D. space-around





KEY TO CORRECTION

| | | | | | | | | |
|-----------|---------------------------------|-------------------------|------------|------|------|------|------|------|
| Pre-test: | 1. D | 1. C | Post-test: | 1. C | 2. A | 3. C | 4. D | 5. A |
| | 2. A | | | | | | | |
| | 3. A | | | | | | | |
| | 4. C | | | | | | | |
| | 5. C | | | | | | | |
| Wrap-up | 1. Flex-direction; row-reverse; | 2. Align-items: center; | | | | | | |

REFERENCES

Websites

- Accessed September, 3 2020 2:30 am
<https://www.w3schools.com/>

