



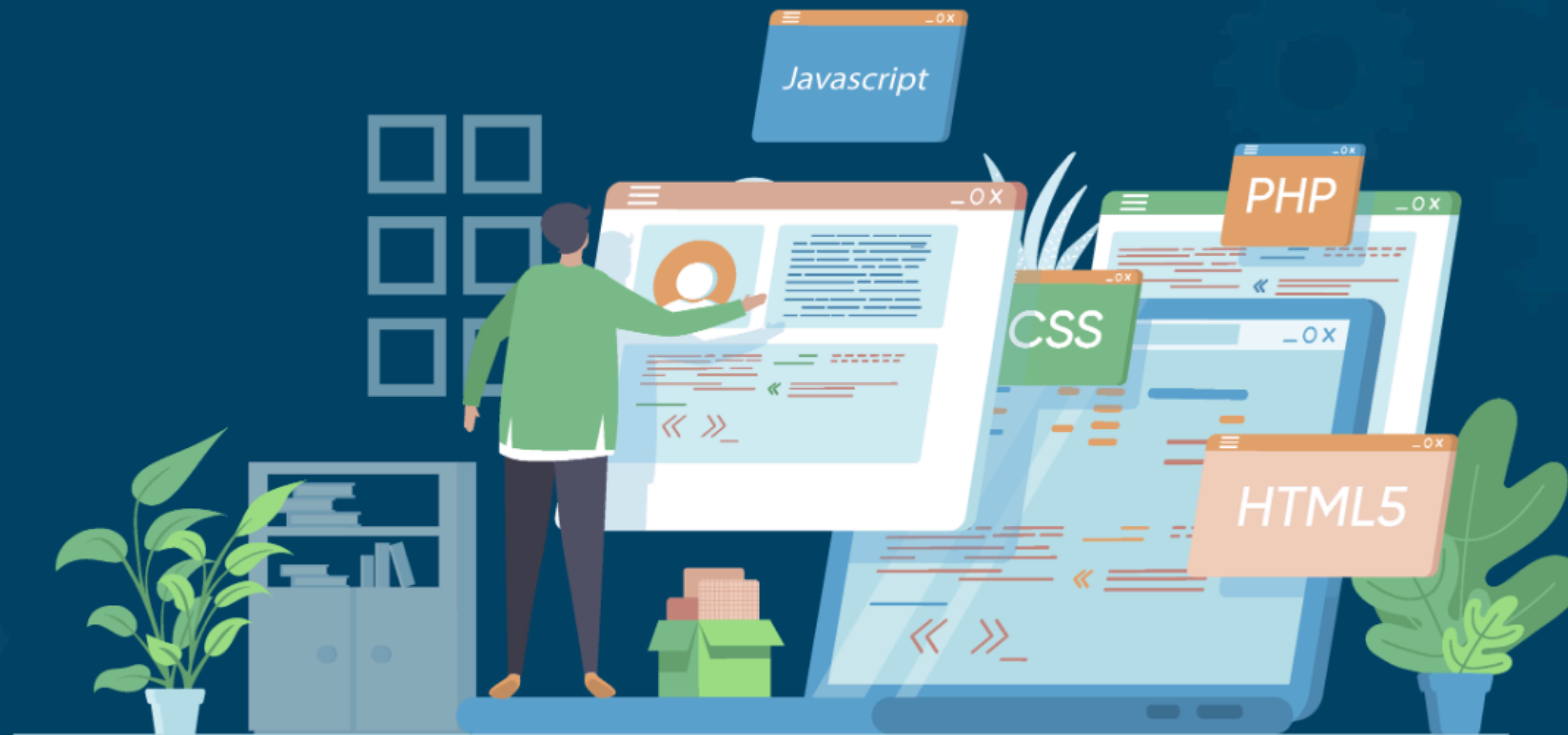
**edureka!**  
*a Veranda Enterprise*



# Full Stack Web Development Program



**edureka!**  
*a Veranda Enterprise*



# Day 13 - Working with Bootstrap Containers

# Titles

---

- Bootstrap Containers
- Bootstrap Grid System
- Grid System Classes



# Learning Objectives

---

By the end of this module, you will be able to:

- Develop custom layouts using Bootstrap containers and the grid system
- Implement Bootstrap containers and grid system in web design projects to create structured and responsive layouts

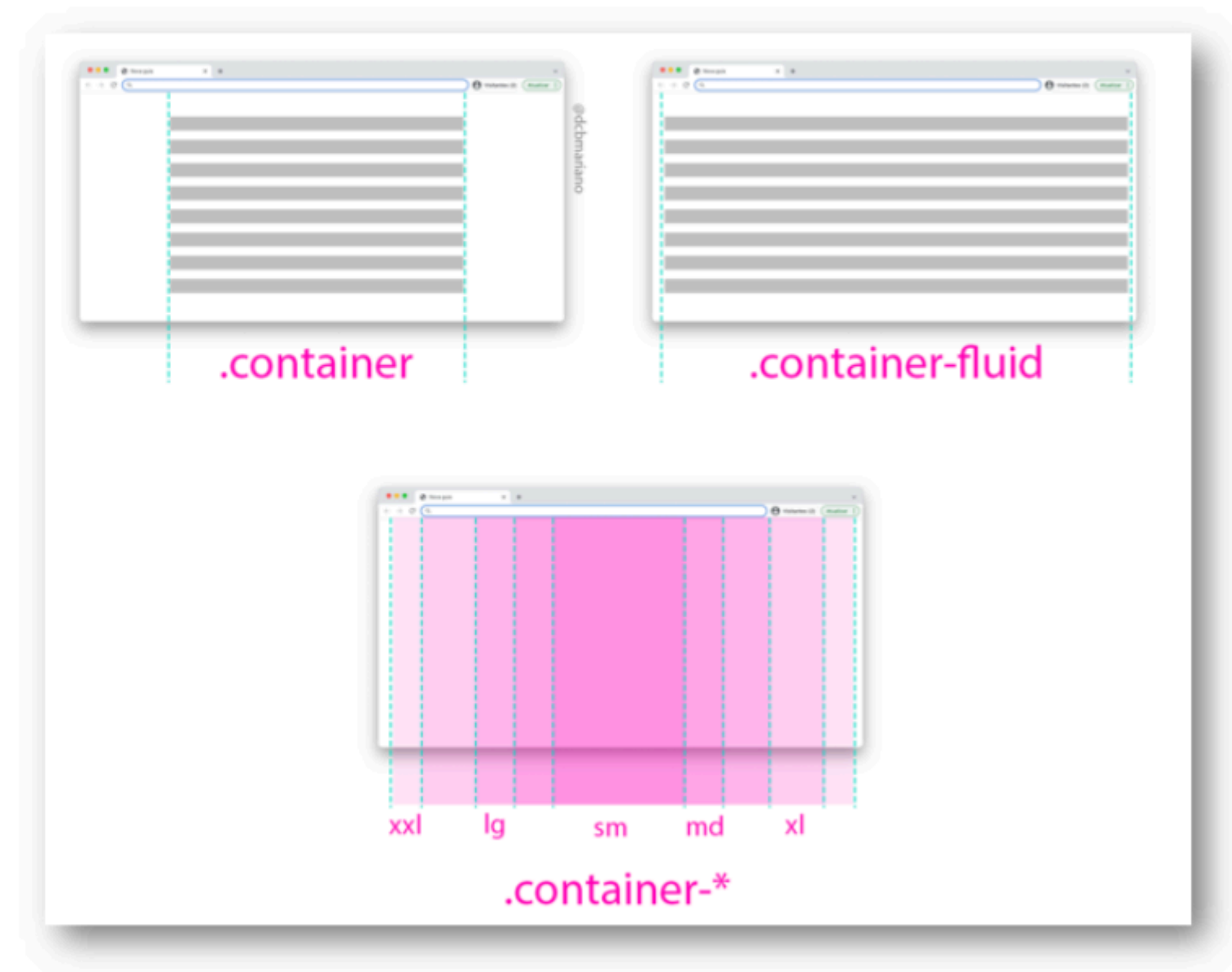




# Working with Bootstrap Containers

# Bootstrap Containers

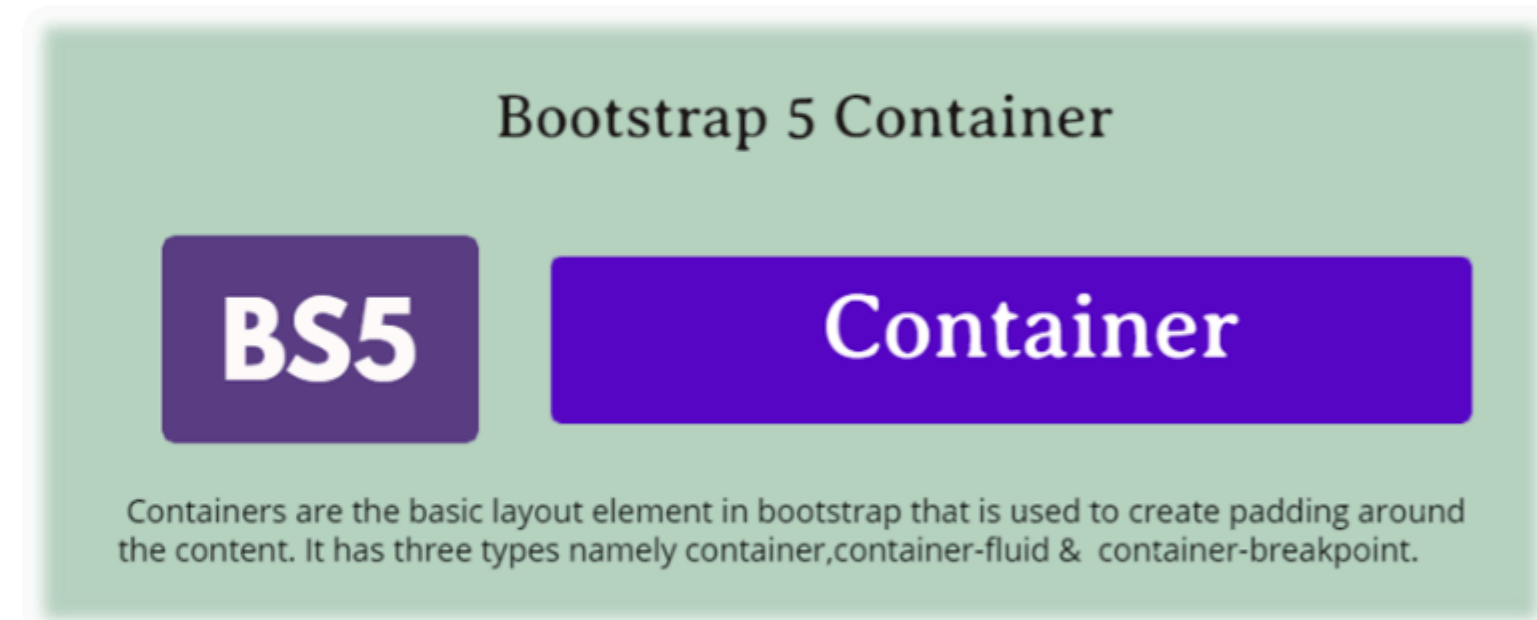
- Containers serve as the foundational framework for content, ensuring responsiveness in design.
- Crucial components within Bootstrap's grid system and containers become necessary when implementing the default grid layout.





# Bootstrap Containers (contd.)

- Bootstrap's grid system hinges on the arrangement of containers, rows, and columns, facilitating content arrangement and alignment.
- While nesting is possible, it's important to note that the majority of layouts can be efficiently designed without the need for nested containers.



# Bootstrap Containers (contd.)

There are three containers available in Bootstrap:

Container name	Container description
<b>.container</b>	This class provides a responsive, fixed-width container. Its max-width changes at each breakpoint.
<b>.container-fluid</b>	This class provides a full-width container, spanning the entire width of the viewport.
<b>.container-{breakpoint}</b>	These containers allow you to specify a class that is 100% wide until the specified breakpoint is reached. We apply max-widths for each of the higher breakpoints.



# Bootstrap Containers (contd.)

Code Example:- index.html

```
<body>

  <div class="container bg-info">
    <h2>Container Example</h2>
    <p>This content is placed within a container.</p>
  </div>

  <div class="container-fluid bg-light">
    <h2>Container Fluid Example</h2>
    <p>This content is placed within a fluid container.</p>
  </div>

  <!-- Link to Bootstrap JS (for optional components like dropdowns, modals, etc.) -->
  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.1/dist/js/bootstrap.bundle.min.js"
    integrity="sha384-HwwvtgBNo3bZJJLYd8oVXjrBZt8cqVSpeBNS5n7C8IVInixGAoxmnlMuBnhbgrkm"
    crossorigin="anonymous"></script>

</body>
```

# Bootstrap Containers (contd.)

---



Output

## Container Example

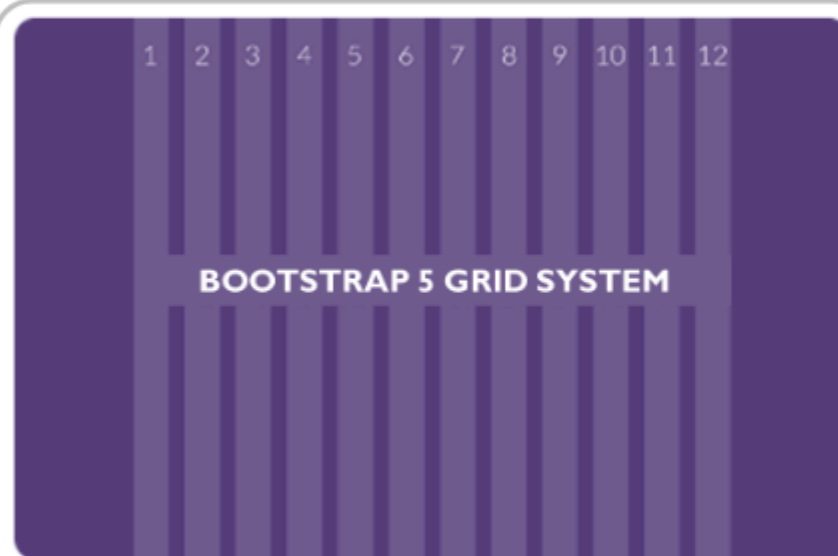
This content is placed within a container.

## Container Fluid Example

This content is placed within a fluid container.

# Bootstrap Grid System

The Bootstrap grid system serves as a fundamental structure for arranging content within a web page layout and is composed of rows and columns that facilitate the organization and responsiveness of the content.



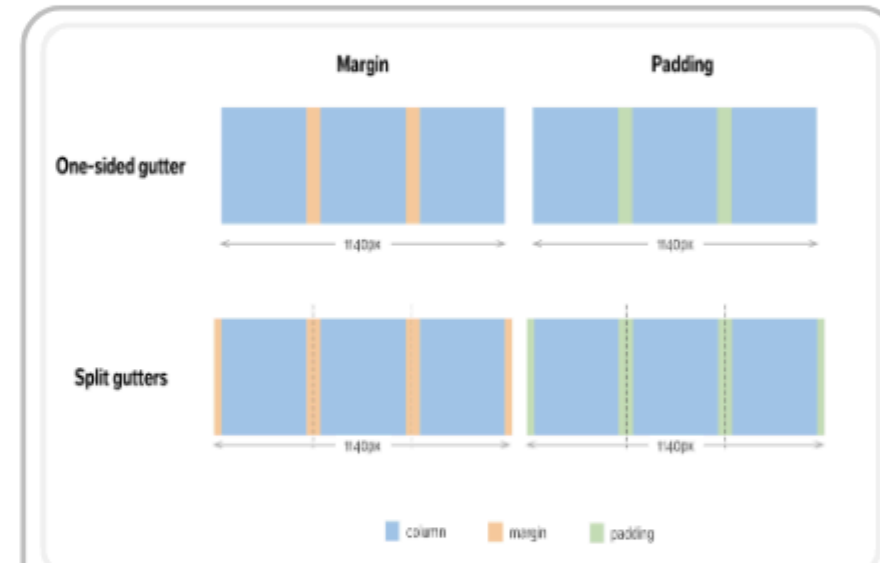
## Rows and Columns

The grid system is based on rows and columns. Rows are containers for columns, which can be sized and positioned within the row to create the desired layout.



## Columns Counts & Class

Bootstrap rows can have up to 12 columns, defined by classes indicating the column count. For instance, use `.col-6` for two equal-width columns.

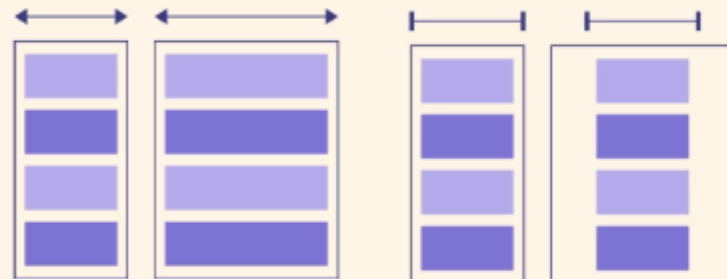


## Gutters and Padding

Columns have gutters for spacing between them, which makes the layout visually pleasing. The no-gutters class can remove margins and padding.

# Bootstrap Grid System (contd.)

## Flexible and Fluid Grids



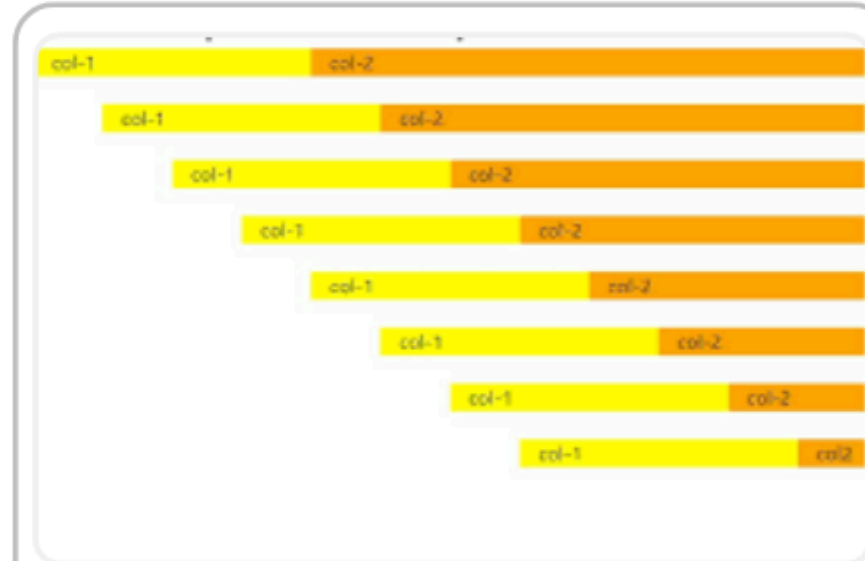
### Fluid and Relative Sizing

Columns' widths are specified in percentages, ensuring their fluidity and responsiveness relative to their parent element.



### Responsive Design

The grid system is important for responsive designs. Bootstrap has responsive classes for different screen sizes, ensuring optimal appearance on various devices.



### Offsetting Columns

Bootstrap grid system allows offsetting columns to create unique layouts by adjusting their positioning within a row.

# Bootstrap Grid System (contd.)

The grid system of Bootstrap enables the creation of up to **12** columns on a page. You can combine these columns to form wider ones if you do not need to use all **12** individually.

