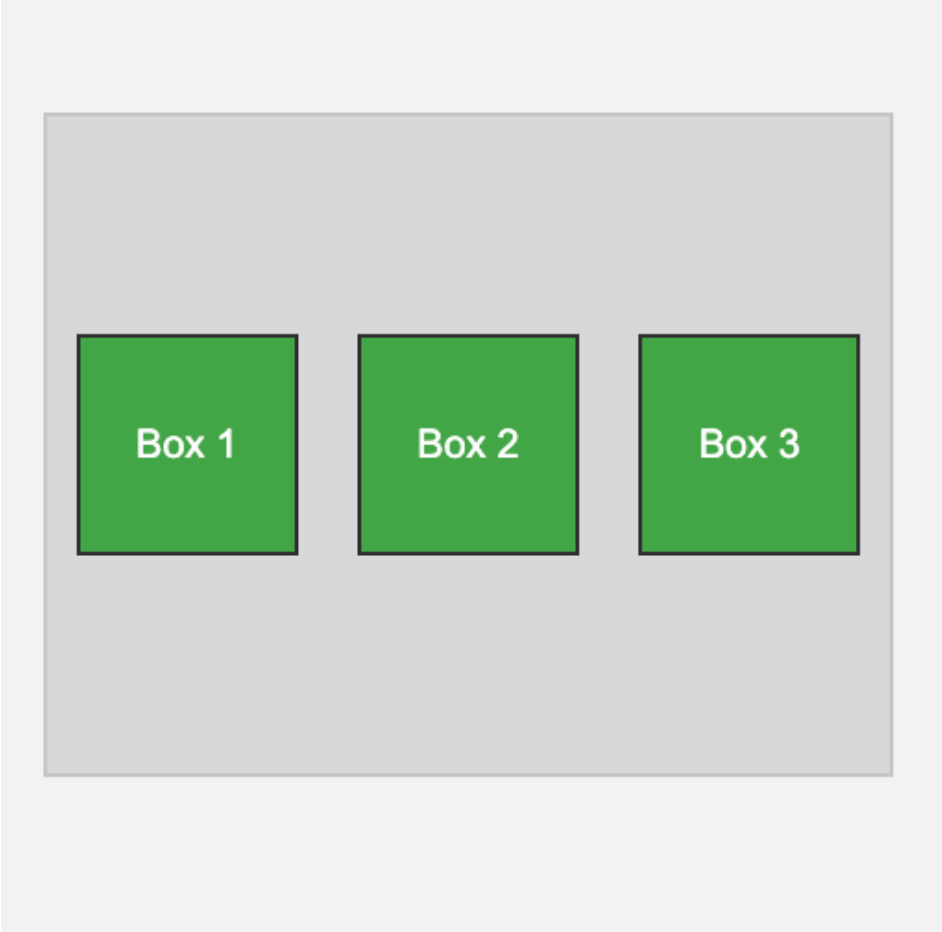
FLEXBOX AND GIT COLLABORATION

FULL STACK SKILLS BOOTCAMP

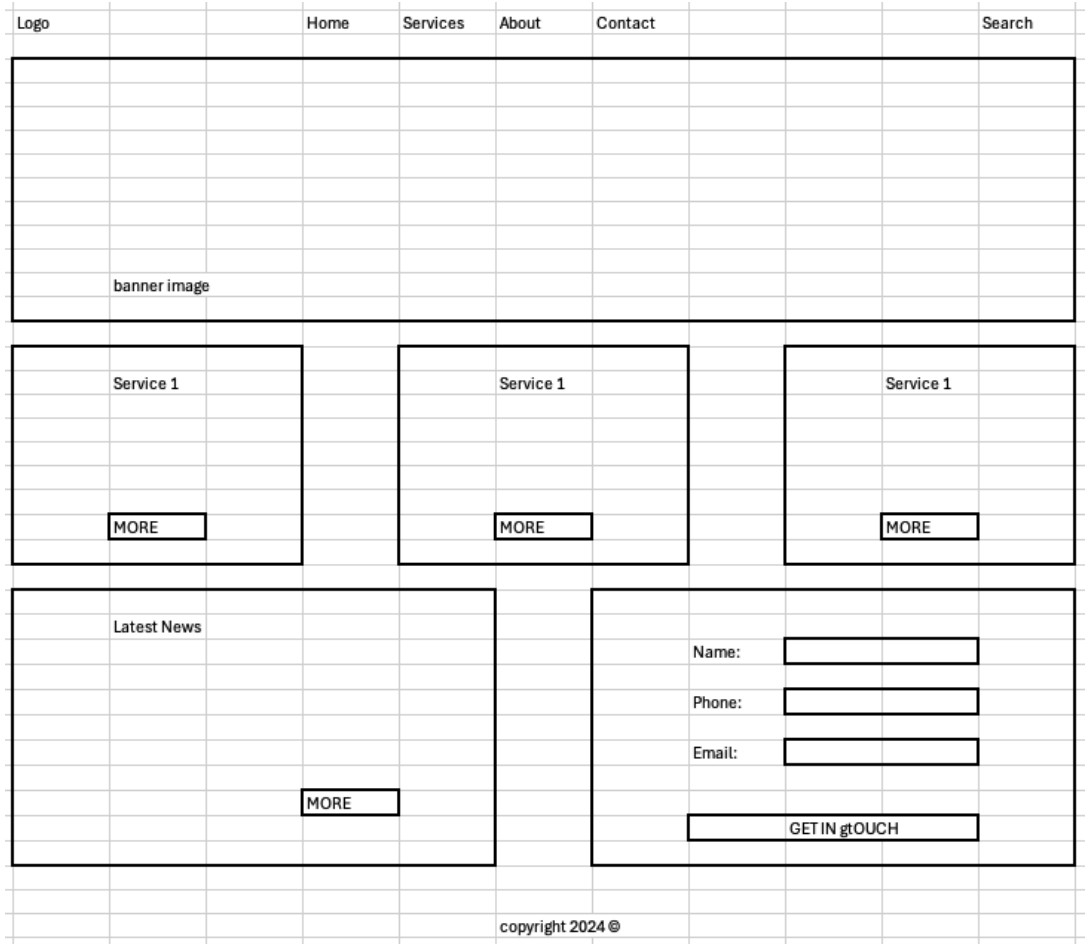
FLEXBOX AND GIT COLLABORATION

- **Lesson Overview:**
- In this lesson, we will be introduced to:
- Wireframing and layout planning using Microsoft Excel or Google Sheets.
- Flexbox fundamentals: display: flex, flex-direction, and flex-wrap.
- Flexbox alignment: justify-content, align-items, and understanding the flex axis.
- Git collaboration using branches and pull requests.



WIREFRAMING AND LAYOUT PLANNING

- Definition: Wireframing is an essential step in designing the structure of your website before diving into code.
- This gives you a visual blueprint of where elements will be placed and how they will be grouped.



TOOLS FOR WIREFRAMING

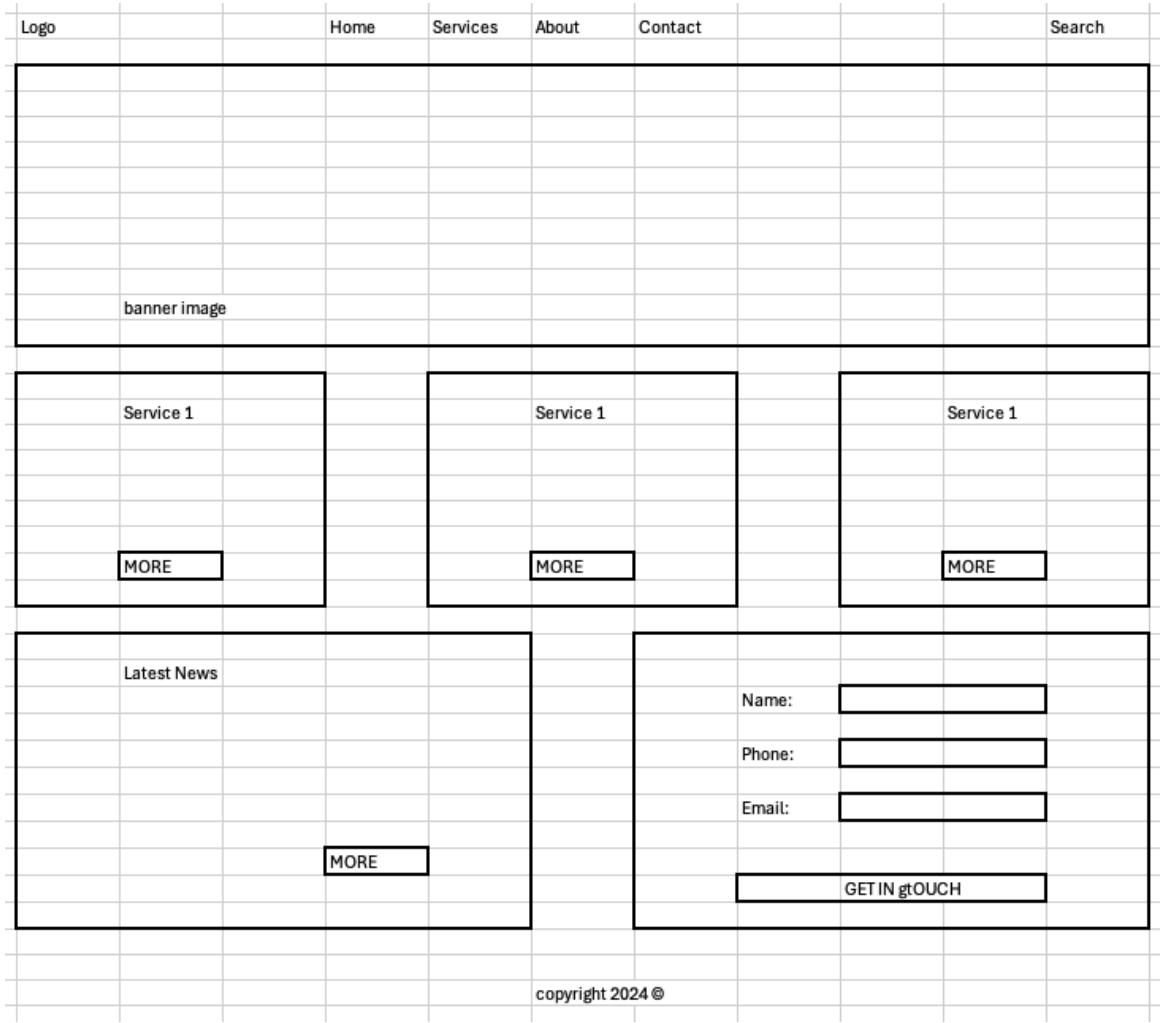
- **Why use tools like Microsoft Excel or Google Sheets?**
- Simple, grid-based layout planning.
- Easy to modify and collaborate on.
- Quick to set up without learning new design software.

Example:

Create a 3x3 grid to represent the website structure.

Plan for a header, content sections, and footer.

Demo...



FLEXBOX BASICS

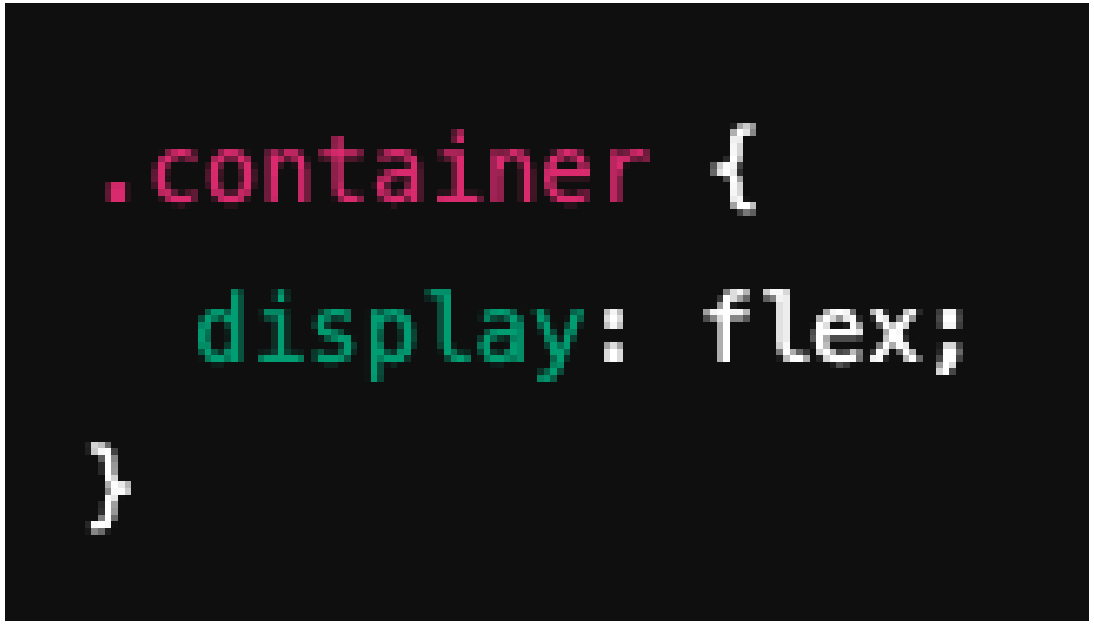
- Flexbox is a powerful tool for creating responsive layouts.
- It allows you to align elements dynamically across different screen sizes.

DISPLAY: FLEX

- display: flex

  display: flex turns a container into a Flexbox container.

- This affects how its child elements are laid out.
- Items inside .container will now follow Flexbox rules for alignment and positioning.



FLEX-DIRECTION

- flex-direction:
  defines the direction in which the flex items are placed in the flex container.
- row (default): Horizontal layout.
- column: Vertical layout.
- row-reverse: Reverses the order of items horizontally.
- column-reverse: Reverses the order of items vertically.
- **Example:**

```
.container {
  display: flex;
  flex-direction: column;
}
```

FLEX-WRAP

- By default, Flexbox tries to fit items on a single line.

  flex-wrap allows items to wrap onto multiple lines.

- nowrap: All items stay on one line.
- wrap: Items will wrap to the next line if they overflow.
- wrap-reverse: Items wrap in reverse order.

  Demo...

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

FLEXBOX ALIGNMENT

- **justify-content** aligns flex items along the main axis (horizontal if flex-direction: row, vertical if flex-direction: column).
- flex-start: Align items to the start.
- center: Center items.
- space-between: Distribute items with space between them.
- space-around: Space around items, with even margins.

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

ALIGN-ITEMS

- **align-items** aligns items along the cross axis (perpendicular to the main axis).
- flex-start: Align items to the top.
- center: Center items vertically.
- stretch: Items stretch to fill container.

```
.container {
  display: flex;
  align-items: center;
}
```

THE FLEX AXIS

- **Main Axis**: Defined by flex-direction. Horizontal if row, vertical if column.
- **Cross Axis**: Perpendicular to the main axis.
- **Tip**: Understanding the axes helps you control alignment using justify-content (main axis) and align-items (cross axis).

  Demo...

GIT COLLABORATION

- Git enables team collaboration by allowing multiple people to work on the same project.
- This section covers branching, pull requests, and merging.

GIT BRANCHING

- **Why use branches?**
- Each developer can work on a feature without affecting the main codebase.
- Avoids conflicts by isolating work.
- **Creating a branch:**

```
git checkout -b feature-branch
```

COMMITTING AND PUSHING CHANGES

- Once you've made changes, commit and push them to the remote repository.

```
git add .
git commit -m "Added new feature"
git push origin feature-branch
```

PULL REQUESTS AND MERGING

- A **pull request (PR)** is how changes from a feature branch get reviewed and merged into the main branch.
- **Pull Request Steps:**

1. Push your branch to GitHub.
2. Open a pull request for review.
3. Once approved, merge the branch.

```
git merge feature-branch
```

BRANCH CLEANUP

- After merging, delete the old branch to keep the repository clean.

```
git branch -d feature-branch
```

RECAP

- **Wireframing** helps plan your website's layout.
- **Flexbox** provides a powerful, flexible layout system.
- **CSS transitions and transformations** bring subtle animations and effects.
- **Git collaboration** ensures smooth teamwork through branches and pull requests.

- **Website Wireframe Beginner's Guide**: https://blog.hubspot.com/website/website-wireframe
- **CSS Tricks: A Complete Guide to Flexbox:** https://css-tricks.com/snippets/css/a-guide-to-flexbox/
- **Git Branches in a Nutshell** https://git-scm.com/book/en/v2/Git-Branching-Branches-in-a-Nutshell

Step8Up Academy

QUESTIONS?