Web Programming

Introduction to Web Programming

Prof. Josué Obregón
Department of Industrial Engineering- ITM
Seoul National University of Science and Technology

Agenda

- Module tutor and students
- Course logistics
  - Goals
  - Methodology
  - Resources
- Contents of this course
- Introduction to Web Programming
  - HTML and CSS

Josué Obregón

- BSc in Computer Engineering,
- Universidad de San Carlos of Guatemala

- MSc in Industrial & Management Systems Engineering
  - Business Process Management Laboratory
  - Process mining applied to SNS

- PhD in Industrial & Management Systems Engineering
  - Industrial Artificial Intelligence Laboratory
  - Interpretable Machine Learning for tree ensembles

  - Assistant Professor in the Dept. of Industrial Engineering - ITM
  - Seoul National University of Science and Technology

- Tel: 7291
- Office: Changhak Hall, Room 334-1
- Office Hours: Tuesday 14:00 – 16:00
- Home-page: https://www.josue-obregon.com/
- Email: jobregon@seoultech.ac.kr
- Publications: Google Scholar Profile

Students

- Question for students
  - Any experience programming? What languages have you used or heard?
  - What website would you like to build?
- Class homepage
  - E-class (http://eclass.seoultech.ac.kr)
- Pre-requisites
  - Basic computer knowledge
  - Python knowledge

Course logistics

- The aims of the module are:
    1. Develop a deep understanding of web app design and implementation using Python, JavaScript, and SQL, alongside frameworks like Django, React, and Bootstrap.
    2. Acquire practical skills in applications design, scalability, security, and user experience to build robust web applications.
    3. Learn to effectively write and use APIs and create interactive user interfaces.
    4. Gain comprehensive knowledge and experience to design and deploy applications on the Internet by the end of the semester.

Class methodology

- There are two class components: Lectures and tutorial sessions (worksheets)
  - Lectures: Tuesday 10:00 – 12:50 (Room 107, Frontier Building)
  - Office Hours: Tuesday 14:00 – 17:00 (Room 334-1, Changhak Hall)
  - Tutorial sessions
- Tutorial sessions
  - Hands-on web programming exercises designed to reinforce the lectures' contents and encourage the students to experiment in real time
  - Worked during class
  - Submission: e-class until the end of the week (by Friday)
  - Every other week you might have additional exercises in the worksheet as homework

Code of conduct

- General rules
  - Class will start at 10:00
  - There will be a 10 minutes break every 50 minutes of class
  - Respect other fellow students and the instructor
- During class
  - Refrain from using any messaging app in your phone or computer
  - Participate actively in the class by answering questions and giving comments and opinions
- Homework
  - Do not copy (copying code is taken seriously)
  - I do not encourage you to use AI tools if you really want to learn how to program
  - Respect due dates

Class methodology

- Evaluation
  - Midterm: 30%
  - Attendance and participation 5%
  - Homework 15%
  - Final exam 30%
  - Term project 20%

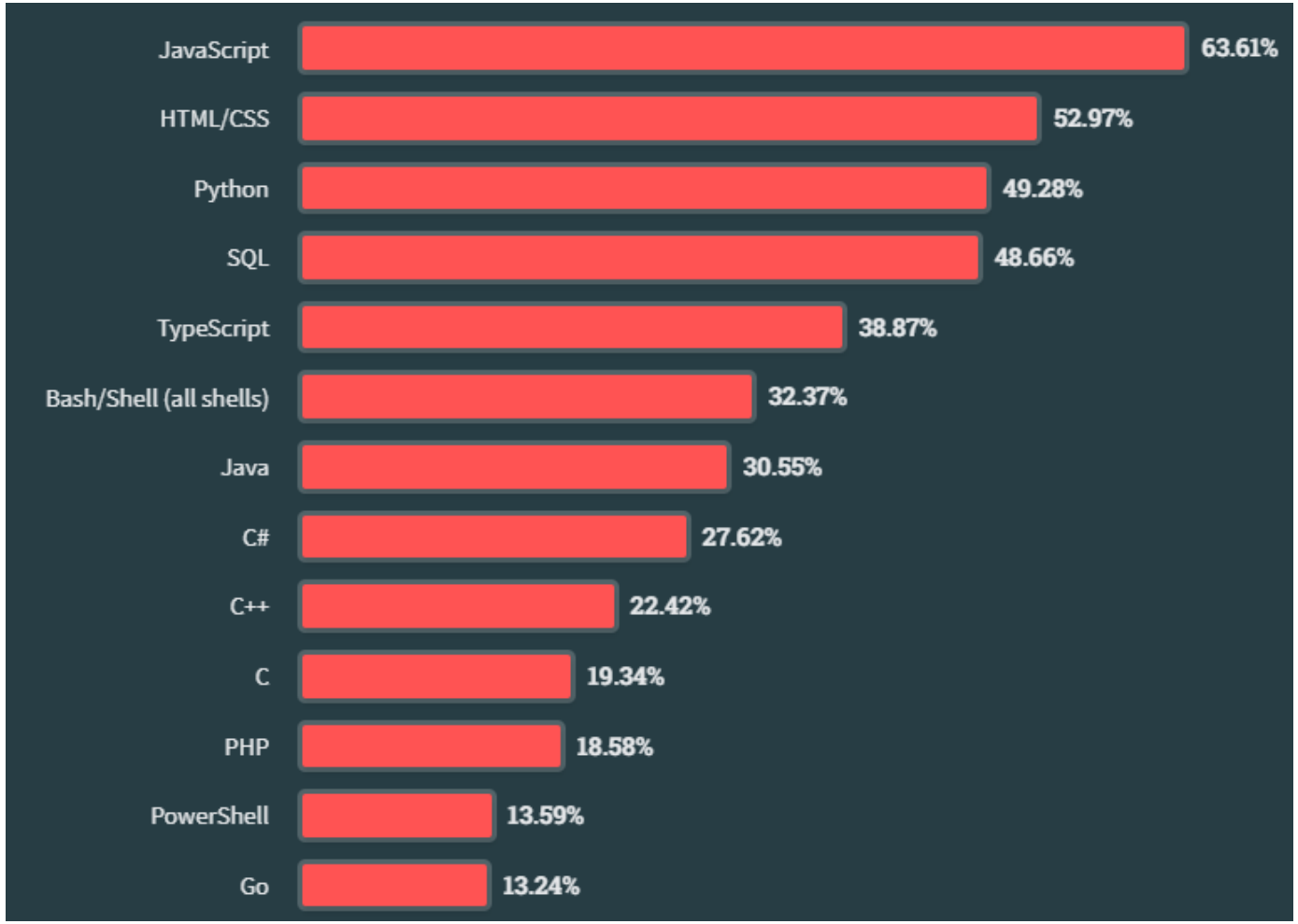
Resources

- This course is mainly based on

  - Lecture notes are available in the cs50 website

- Integrated Development Environment (IDE)
  - Visual Studio Code
- Other tools
  - Python
  - Git

Quick global overview



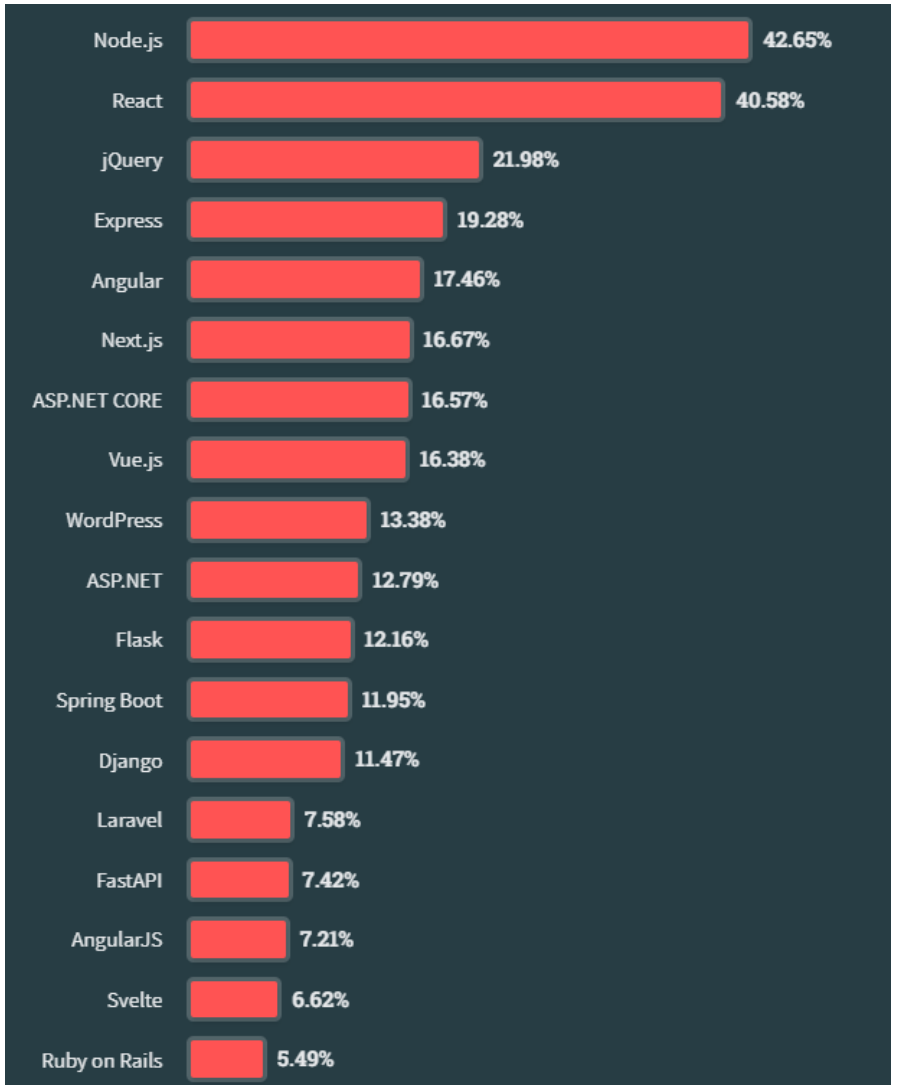
Programming, scripting, and markup languages

stack overflow

2023 Developer Survey

This report is based on a survey of 89,184 software developers from 185 countries around the world.

Quick global overview



stack overflow

2023 Developer Survey

Web frameworks and technologies

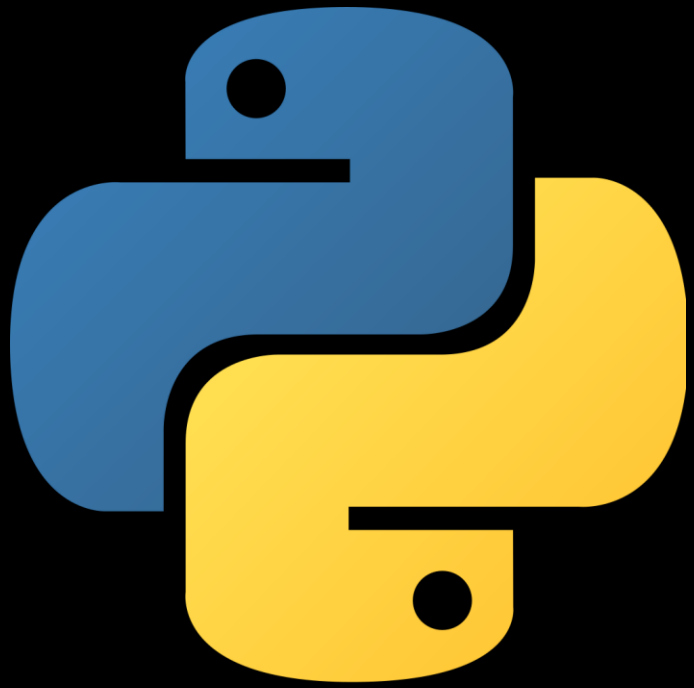
Web Programming
with Python and JavaScript

Web Programming

HTML and CSS

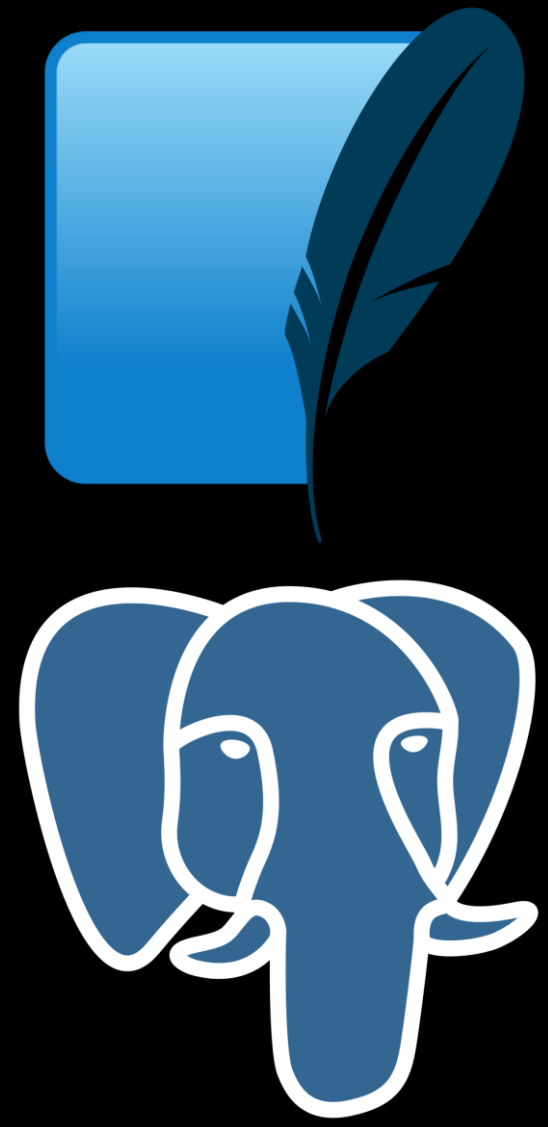
Git

Python

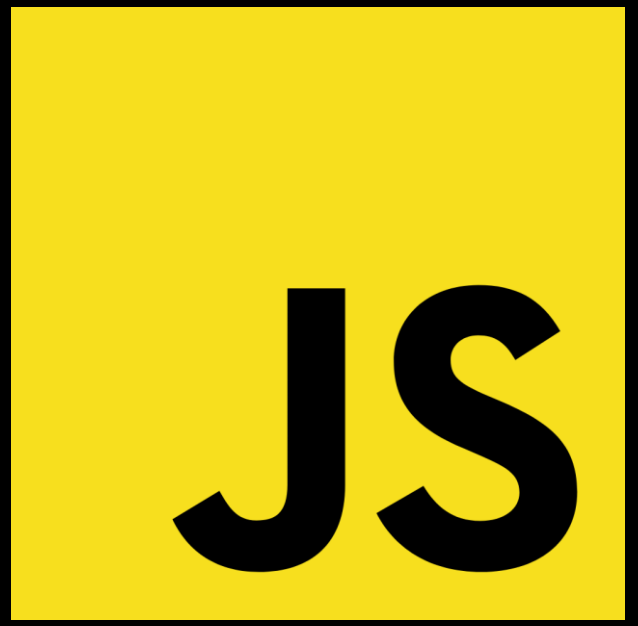


Django

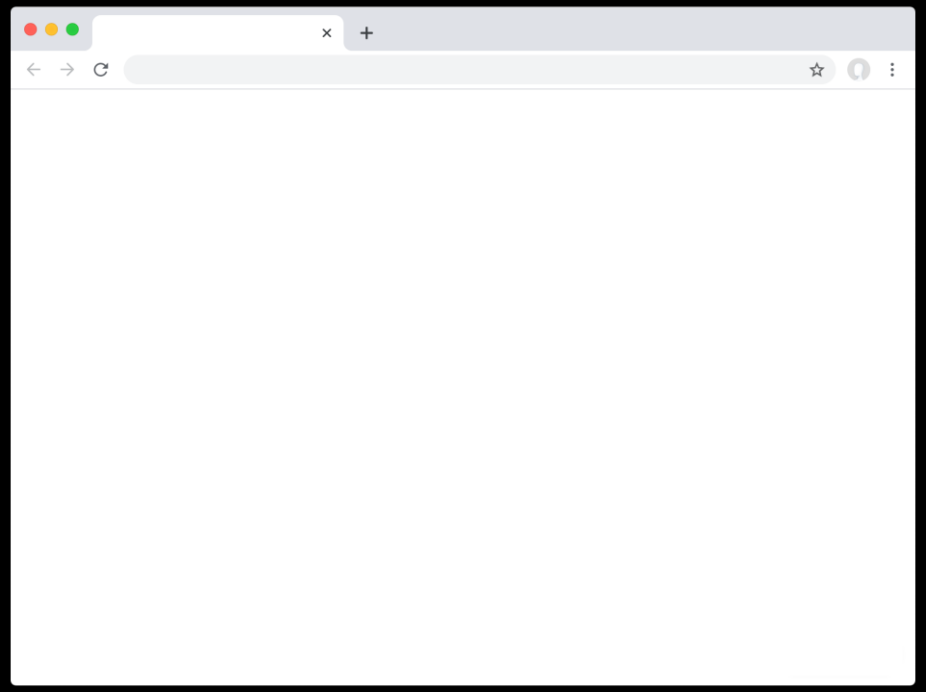
SQL, Models, and Migrations



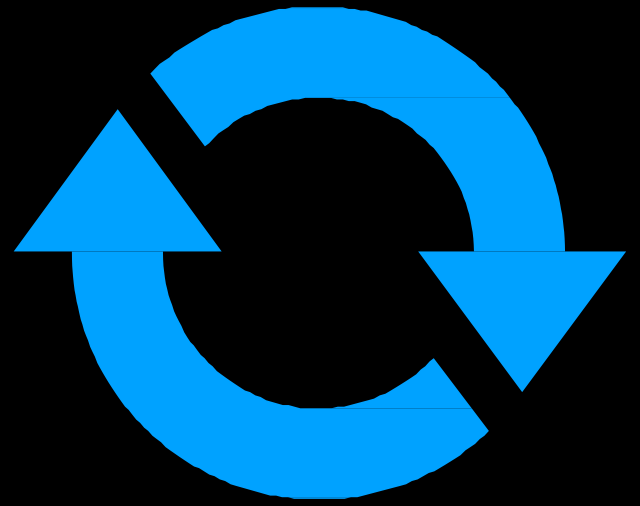
JavaScript



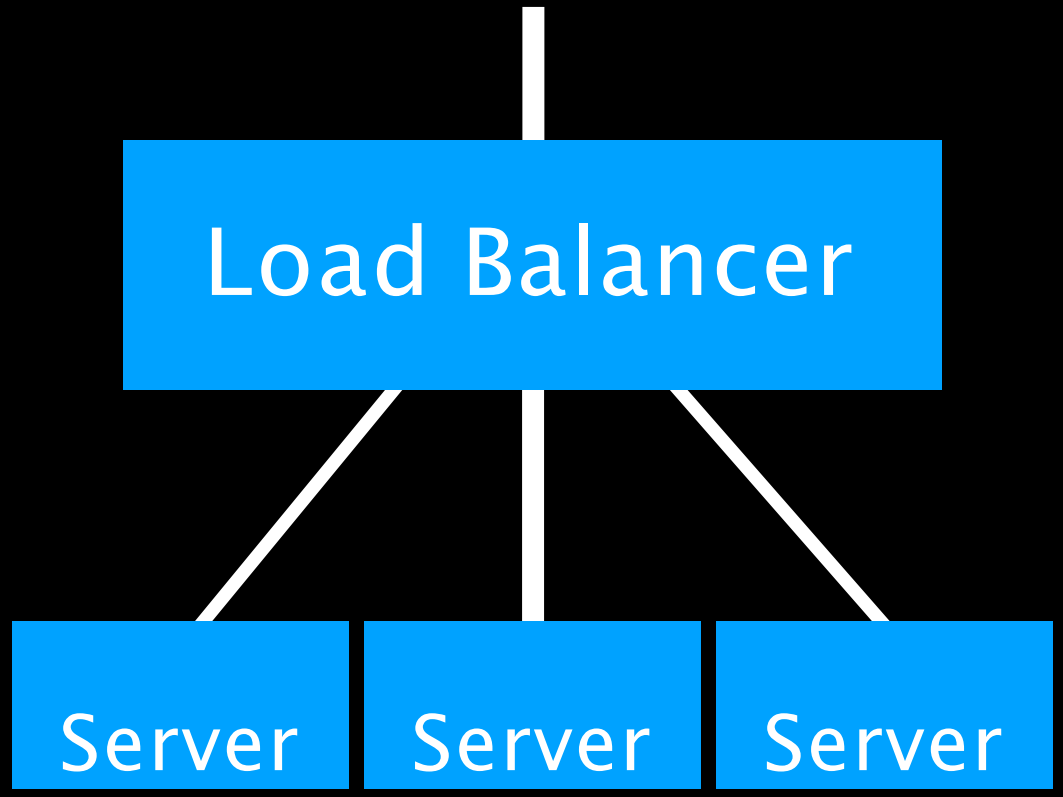
User Interfaces



Testing and CI/CD



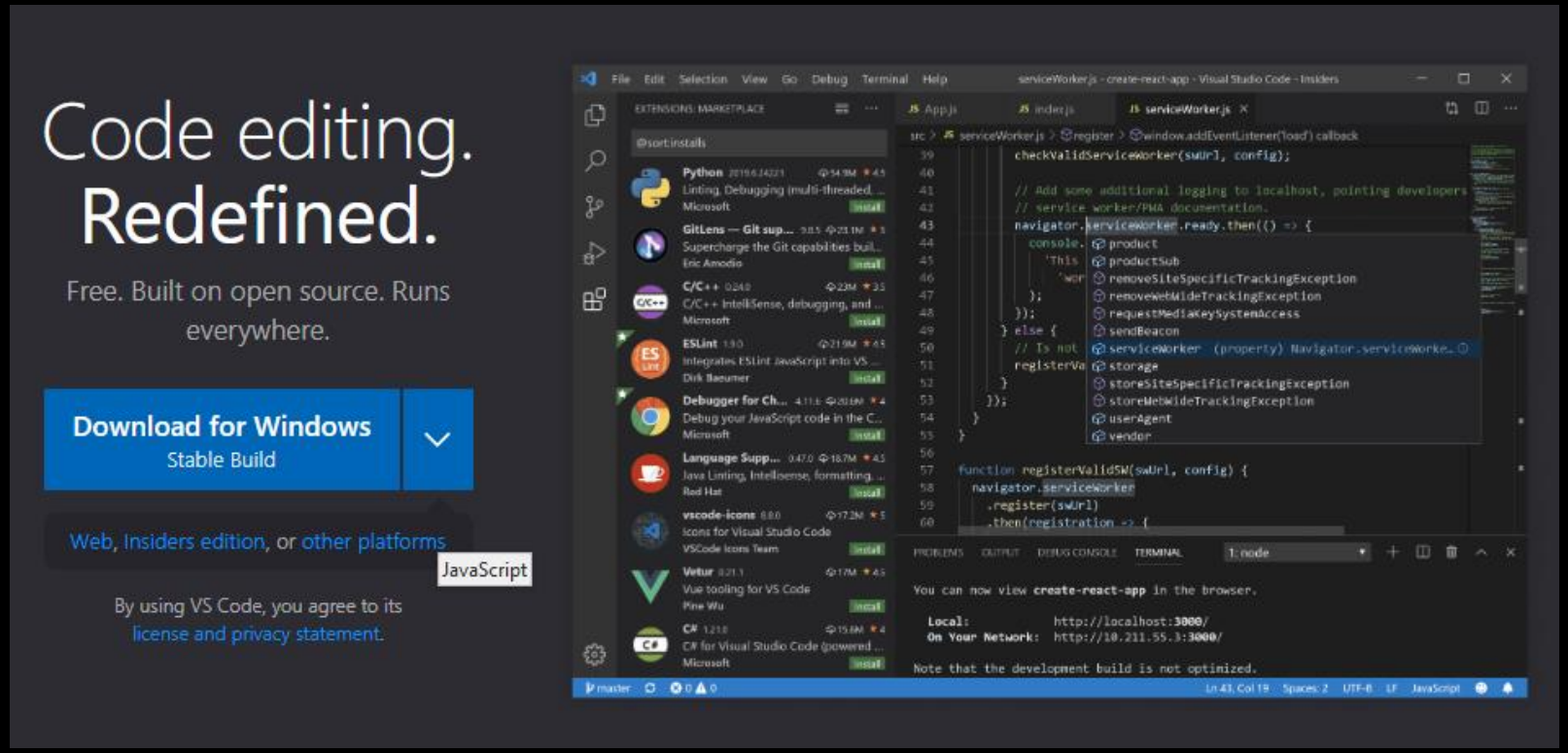
Scalability and Security



HTML and CSS

Visual Studio Code

https://code.visualstudio.com/



HTML

```
<!DOCTYPE html>
<html lang="en">
    <head>
        <title>Hello!</title>
    </head>
    <body>
        Hello, world!
    </body>
</html>
```

```
<!DOCTYPE html>
<html lang="en">
    <head>
        <title>Hello!</title>
    </head>
    <body>
        Hello, world!
    </body>
</html>
```

```
<!DOCTYPE html>
<html lang="en">
    <head>
        <title>Hello!</title>
    </head>
    <body>
        Hello, world!
    </body>
</html>
```

```
<!DOCTYPE html>
<html lang="en">
    <head>
        <title>Hello!</title>
    </head>
    <body>
        Hello, world!
    </body>
</html>
```

```
<!DOCTYPE html>
<html lang="en">
    <head>
        <title>Hello!</title>
    </head>
    <body>
        Hello, world!
    </body>
</html>
```

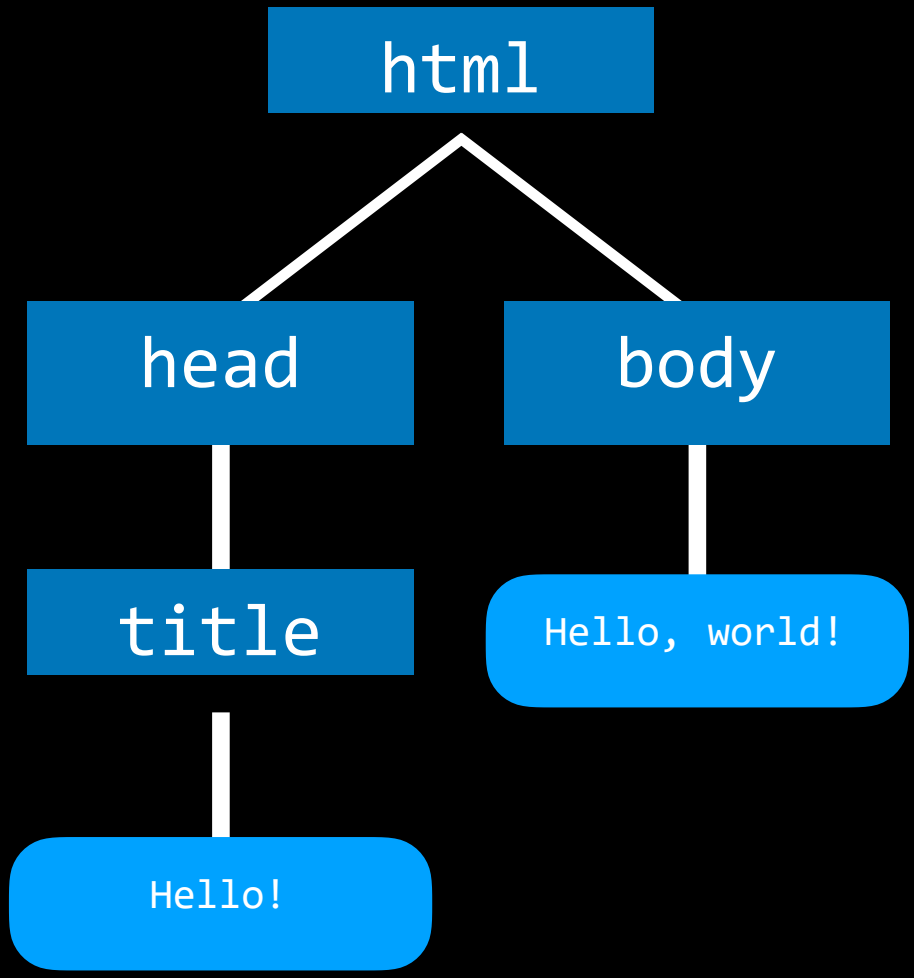
```
<!DOCTYPE html>
<html lang="en">
    <head>
        <title>Hello!</title>
    </head>
    <body>
        Hello, world!
    </body>
</html>
```

```
<!DOCTYPE html>
<html lang="en">
    <head>
        <title>Hello!</title>
    </head>
    <body>
        Hello, world!
    </body>
</html>
```

```
<!DOCTYPE html>
<html lang="en">
    <head>
        <title>Hello!</title>
    </head>
    <body>
        Hello, world!
    </body>
</html>
```

Document Object Model

```
<!DOCTYPE html>
<html lang="en">
    <head>
        <title>Hello!</title>
    </head>
    <body>
        Hello, world!
    </body>
</html>
```



Common HTML Tags

- <h1>, <h2>, ..., <h6>
- ,
- <img>
- <a>
- <table>
- <form>
- ...

CSS

Common CSS Properties

- color
- text-align
- width, height
- margin, padding
- font-family, font-size, font-weight
- border
- ...

Identifying Elements

- `div`
- `span`
- `id`
- `class`

Specificity

1.inline
2. id
3. class
4. type

Specificity

```
<div id="foo">
    Hello!
</div>
```

```
div {
  color: blue;
}
```

Hello!

Specificity

```
<div id="foo">
    Hello!
</div>
```

```
div {
  color: blue;
}

#foo {
  color: red;
}
```

Hello!

Specificity

```
<div id="foo">
    Hello!
</div>
```

```
#foo {
  color: red;
}

div {
  color: blue;
}
```

Hello!

CSS Selectors

`a, b`	Multiple Element Selector
`a b`	Descendant Selector
`a > b`	Child Selector
`a + b`	Adjacent Sibling Selector
`[a=b]`	Attribute Selector
`a:b`	Pseudoclass Selector
`a::b`	Pseudoelement Selector

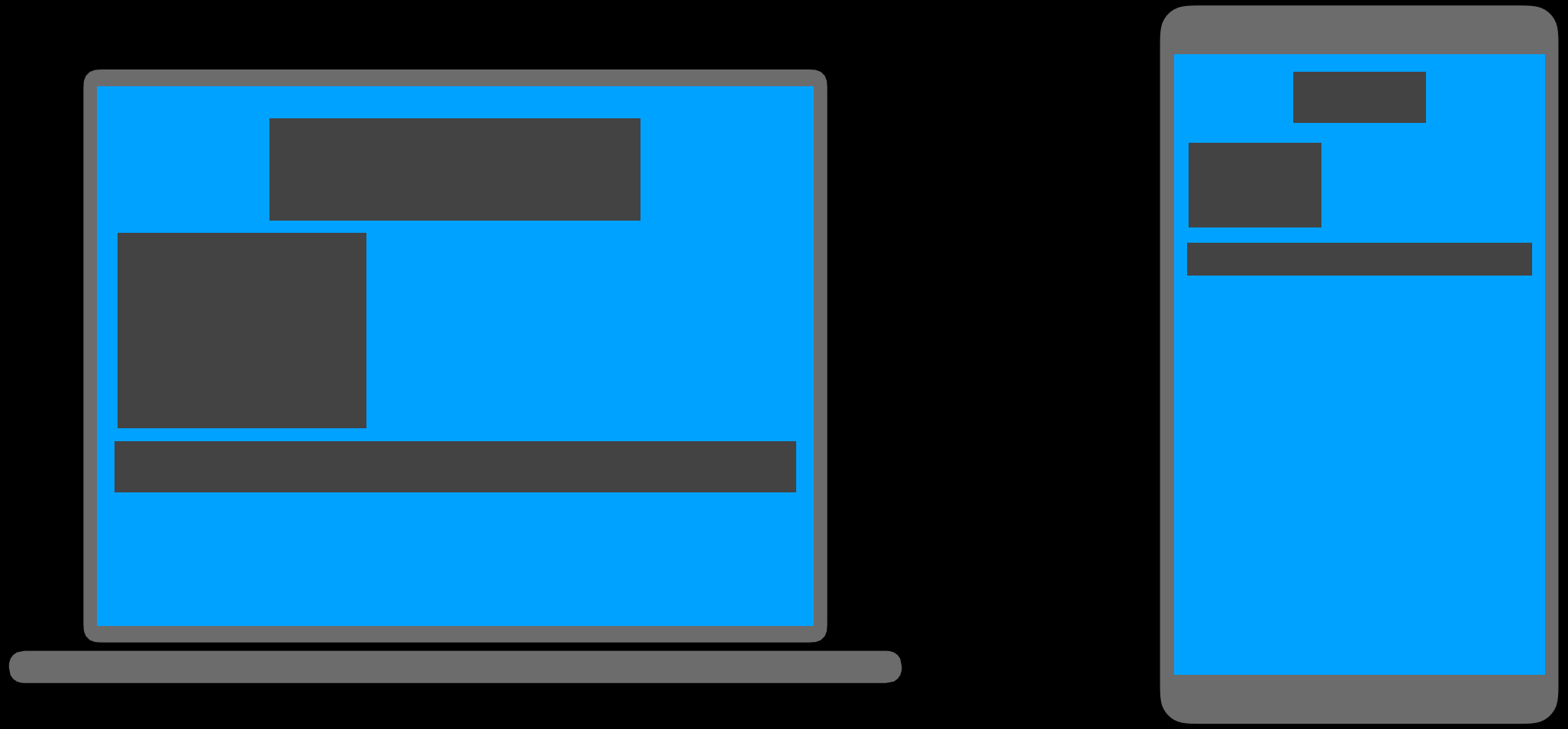
Responsive Design

Responsive Design

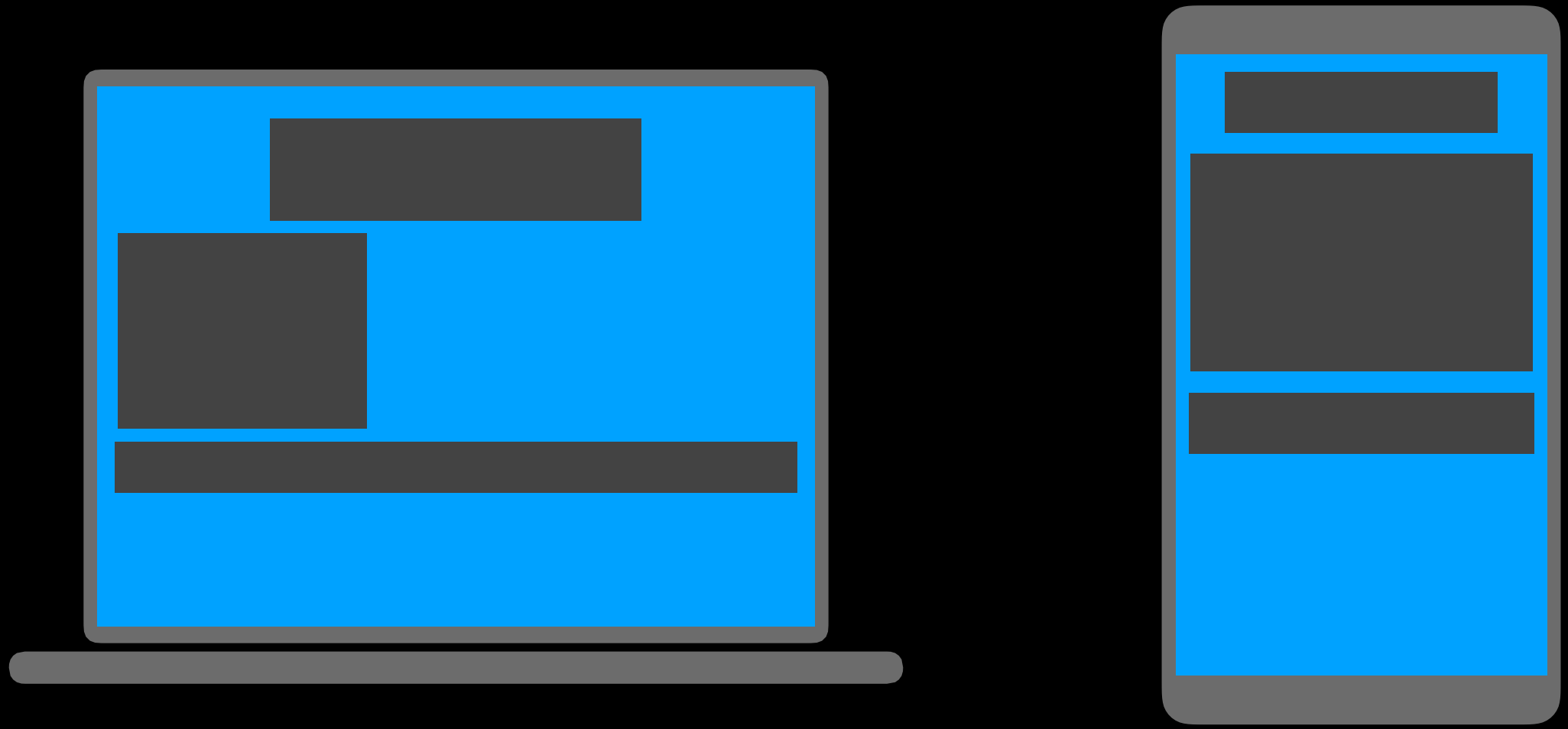
- `viewport`
- Media Queries
- Flexbox
- Grids

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

Viewport



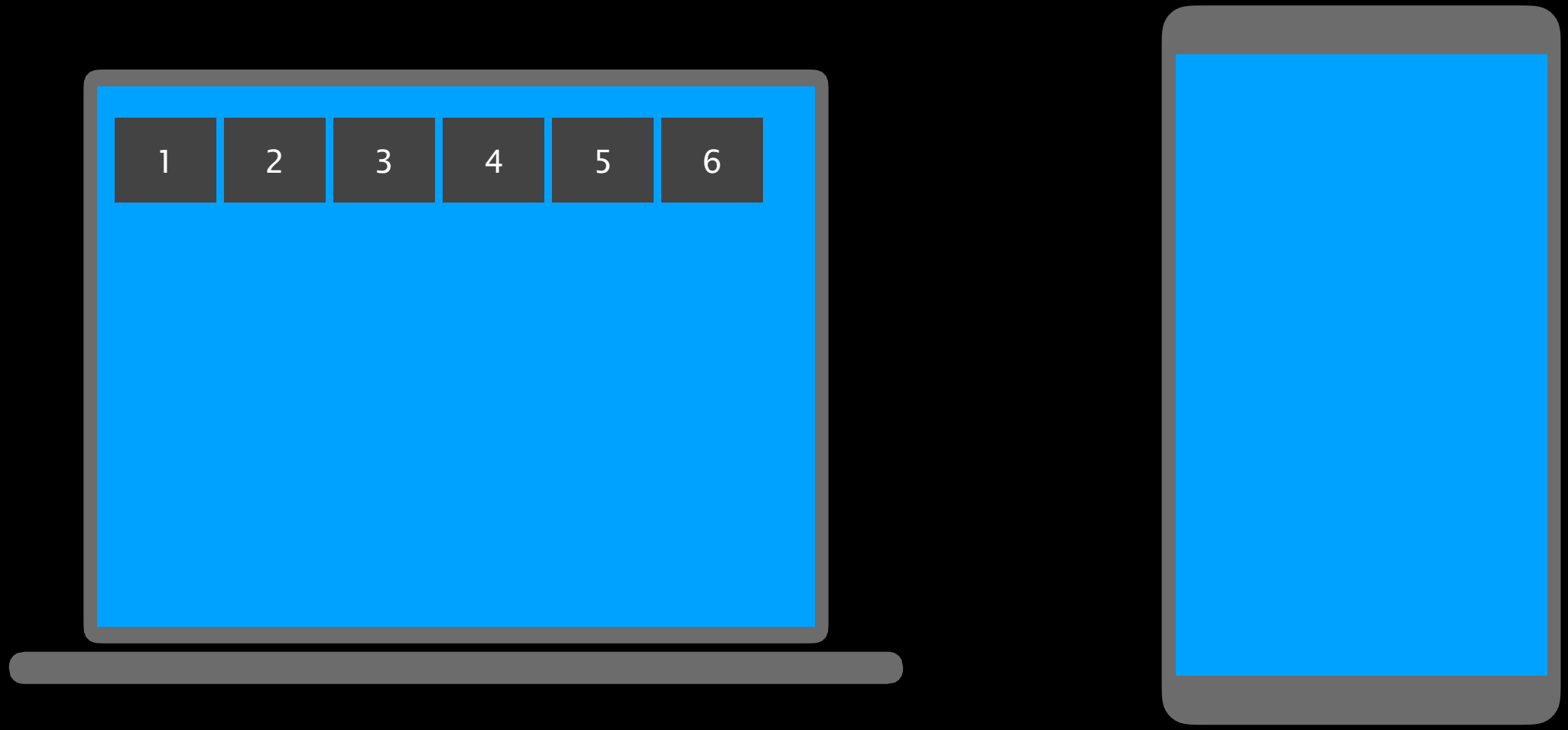
Viewport



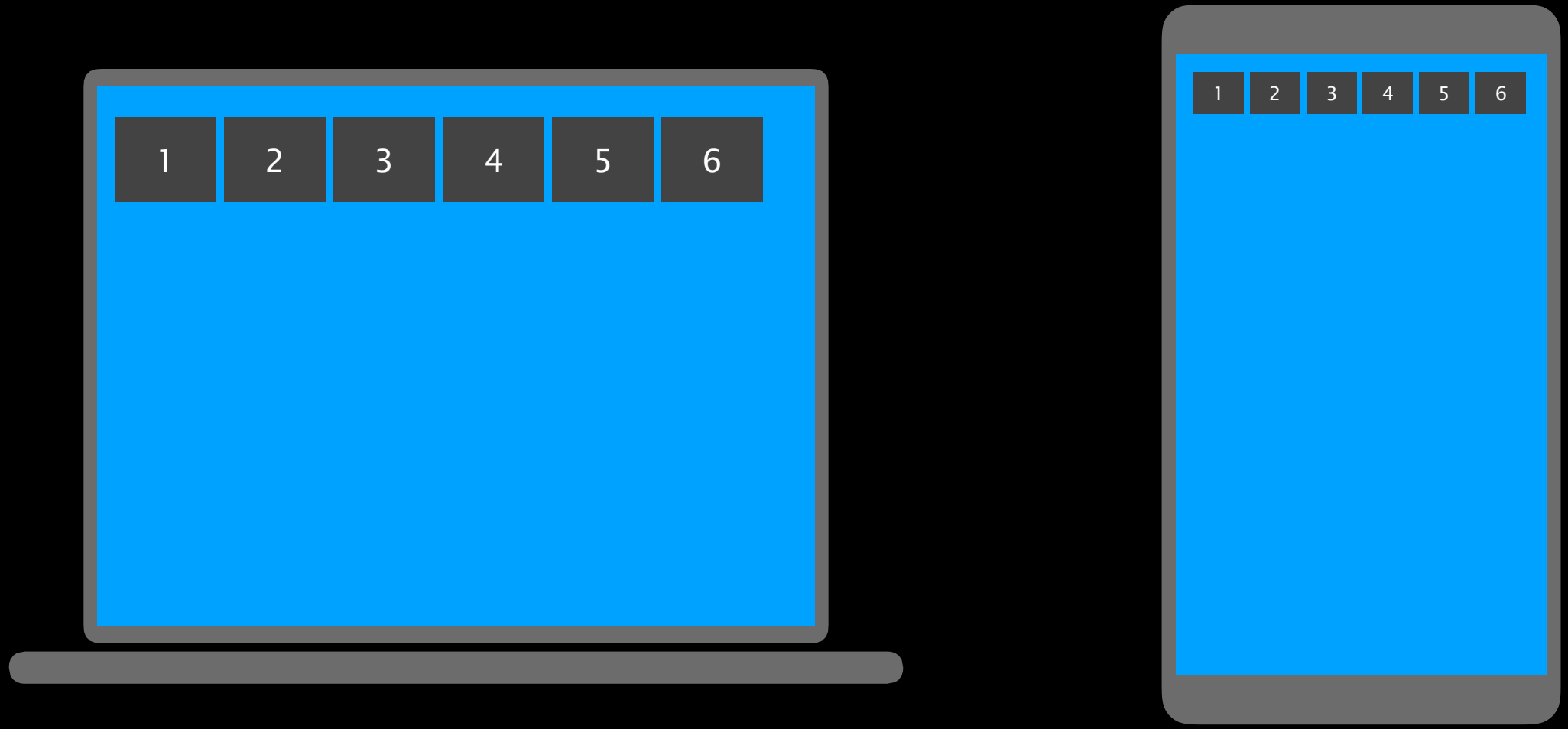
Media Queries

- Media Types: `print`, `screen`, …
- Media Features: `height`, `width`, `orientation`, …

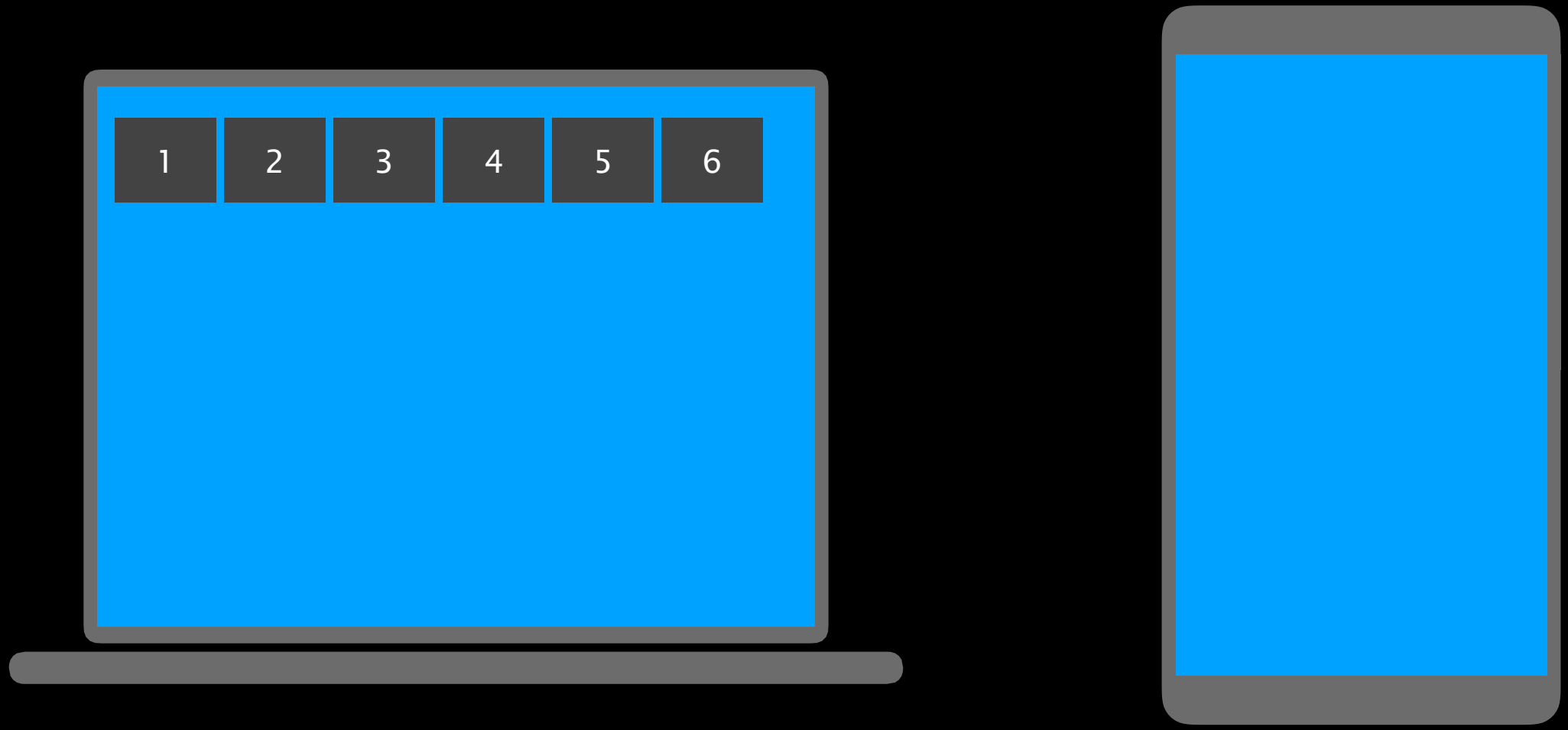
Flexbox



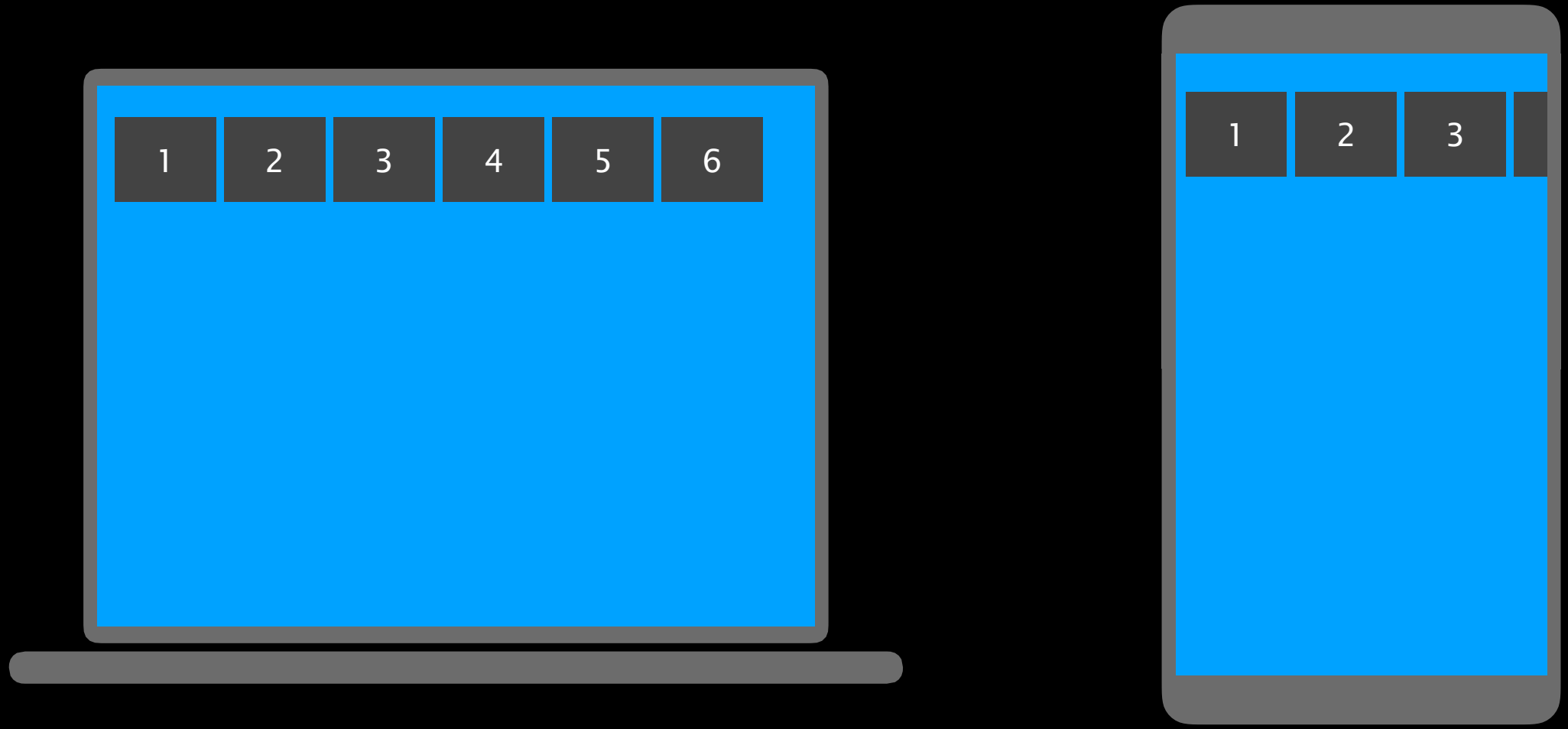
Flexbox



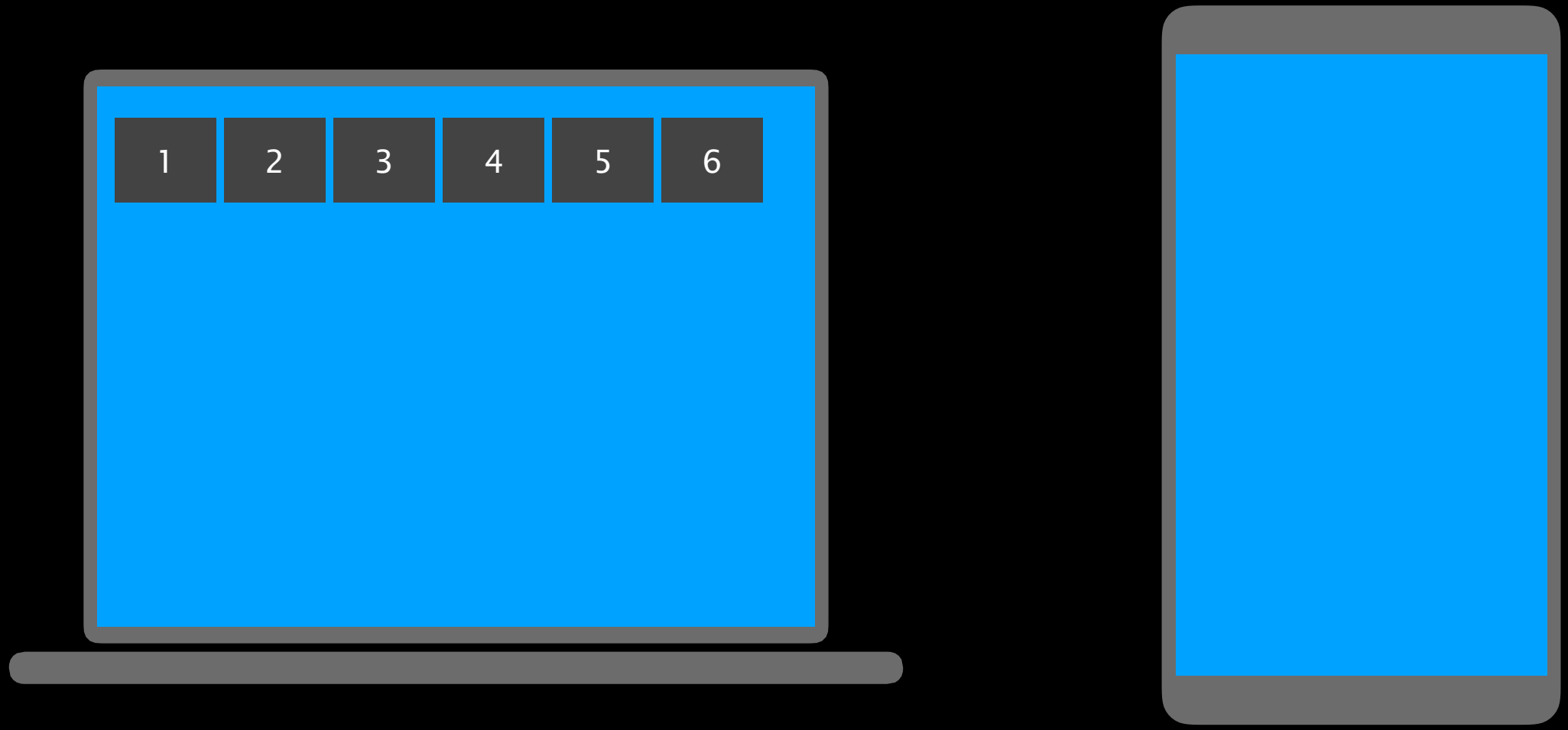
Flexbox



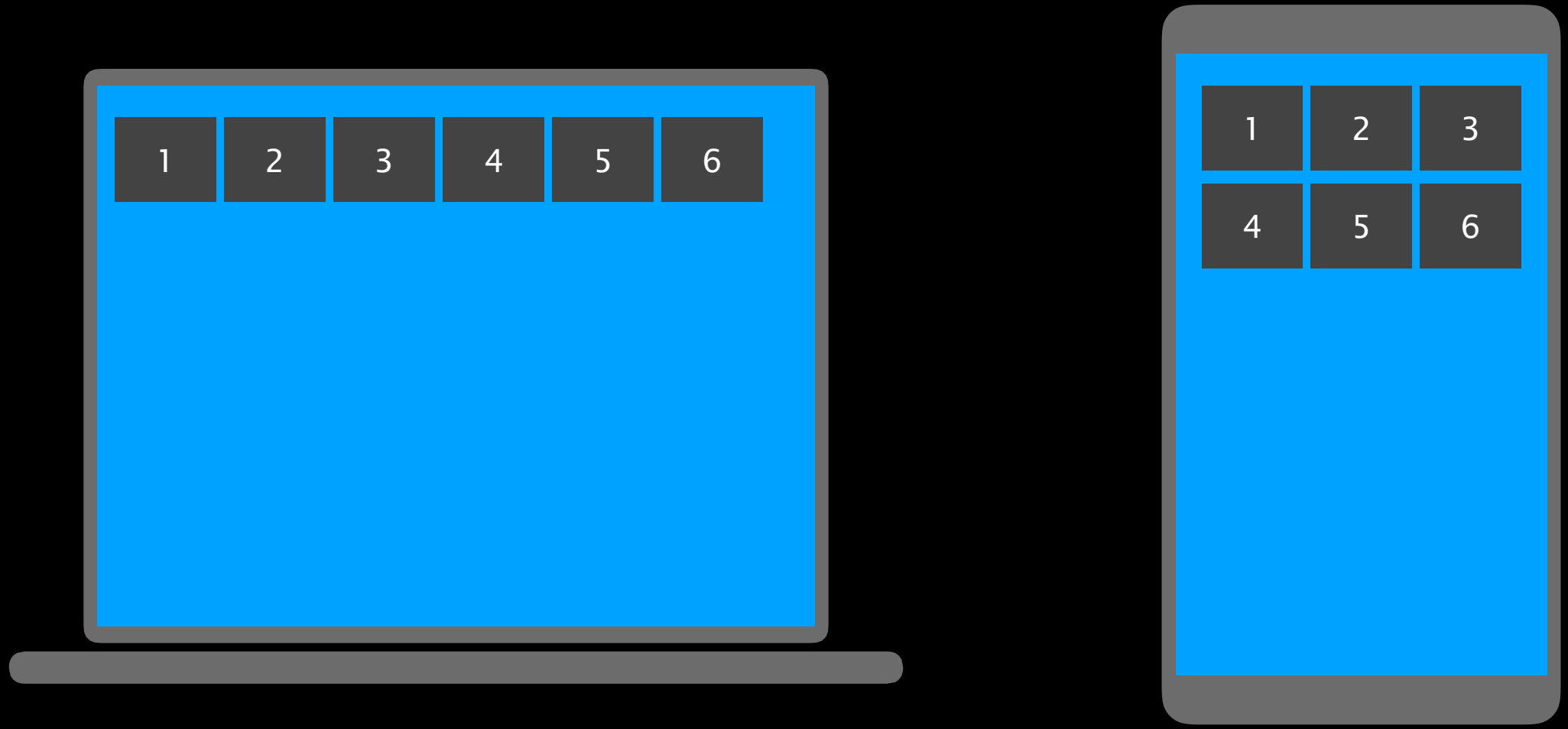
Flexbox



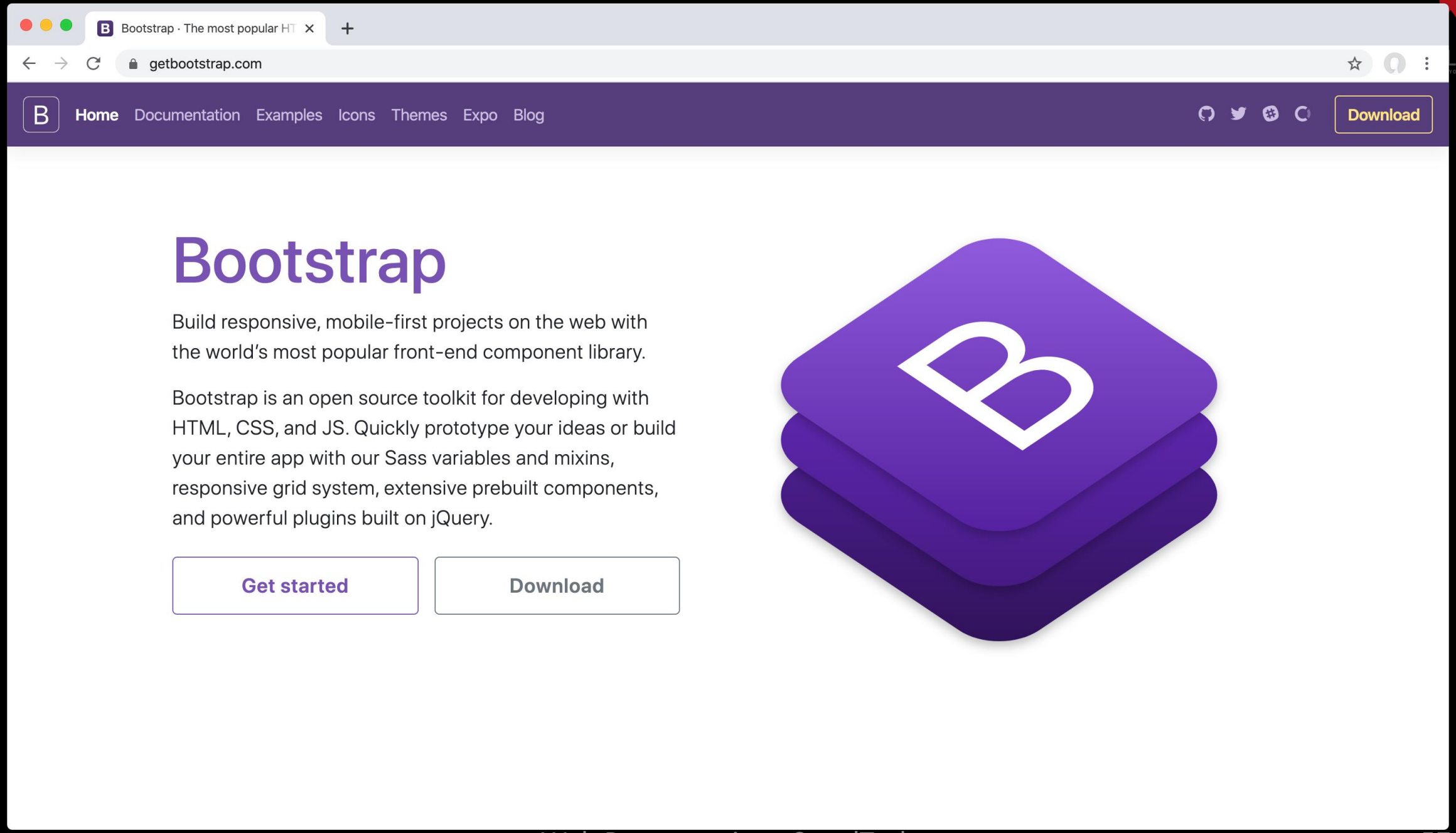
Flexbox



Flexbox



Bootstrap


Bootstrap · The most popular HT
getbootstrap.com
Home
Documentation
Examples
Icons
Themes
Expo
Blog
Download
Bootstrap
Build responsive, mobile-first projects on the web with the world's most popular front-end component library.
Bootstrap is an open source toolkit for developing with HTML, CSS, and JS. Quickly prototype your ideas or build your entire app with our Sass variables and mixins, responsive grid system, extensive prebuilt components, and powerful plugins built on jQuery.
Get started
Download

Sass

Web Programming
with Python and JavaScript