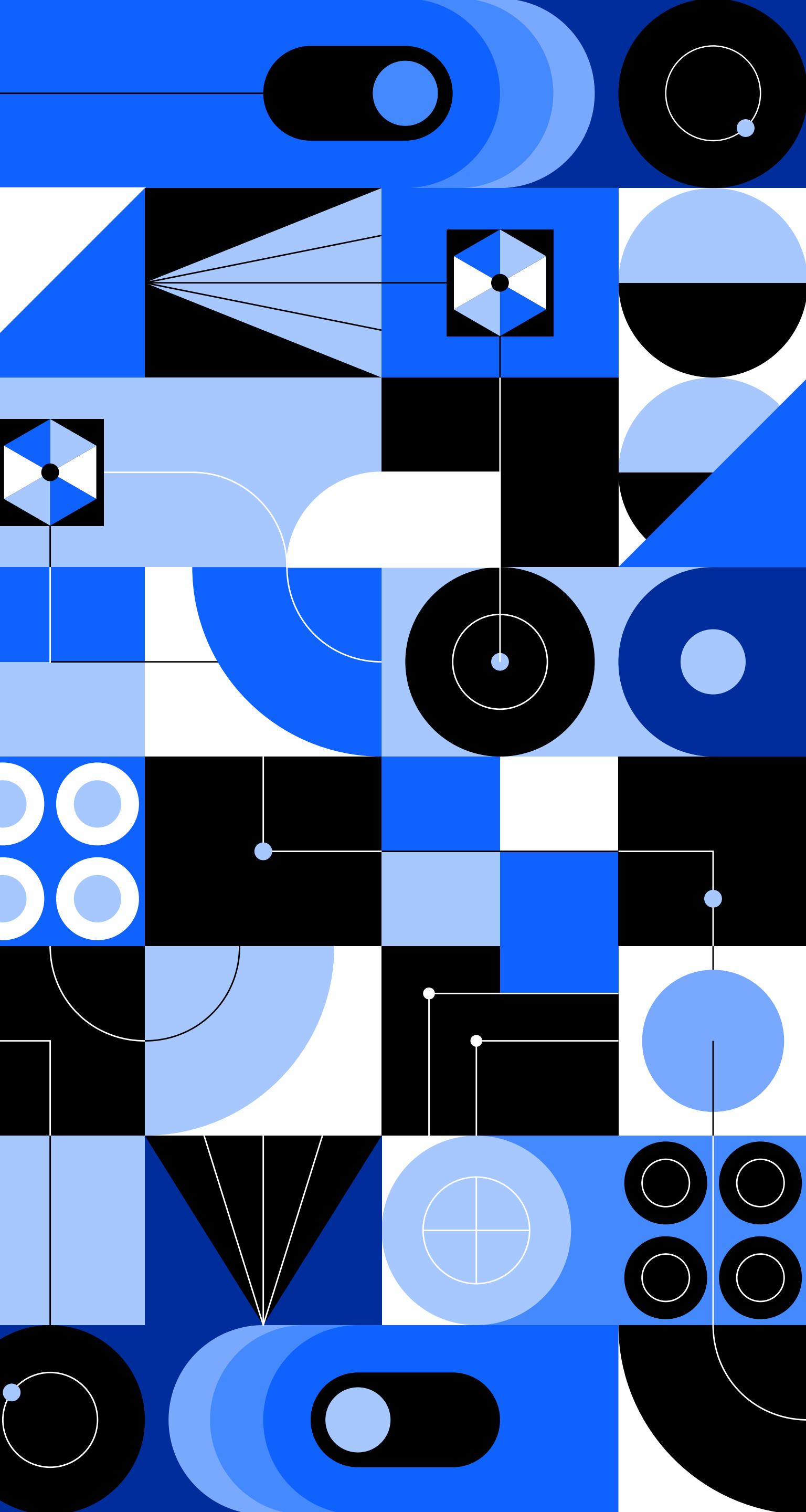


IBM Accelerate

Software Developer

Track

Wednesdays June 5th – July 24th
6:00pm – 8:00pm Eastern Time



June 7th, 2023

Frontend Technology Introduction

Welcome: Alan Bivens, Sujeily P. Fonseca-Gonzalez



Alan Bivens

VP, IBM MultiCloud Platform
SW Track Executive Sponsor

Sujeily P. Fonseca

Software Engineering
Manager, Technical Leader,
MultiCloud Saas Platform
SW Track Lead



Speakers: Kevin See, Joe Diaz



Kevin See

Software Developer,
MultiCloud Saas Platform

Joe Diaz

Site Reliability Engineer,
MultiCloud Saas Platform



IBM Accelerate : Talent Identification Program

2024 Software Track Learning Curriculum

Executive Sponsors: Alan Bivens and Tim Humphrey; Track Lead: Sujeily Fonseca

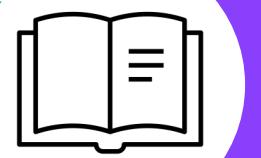


June 5 – Week 1

Frontend Technology Intro

Presenters:
Multiple

Foundational Skills Topic:
Corporate Communications



June 12 – Week 2

JavaScript, React, and Styling

Presenters:
Multiple

Foundational Skills Topic:
Growth Behaviors



June 19 – Week 3

Event Handling, Synchronicity & Testing

Presenters:
Multiple

Foundational Skills Topic:
Job Applications and Resumes



June 26 – Week 4

Client-Side versus Server-Side, Backend Intro, and Web App Security

Presenters:
Multiple

Foundational Skills Topic:
Personal Branding & LinkedIn Profile Refinement



July 3 – Week 5

Generative AI

Presenters:
Multiple

Foundational Skills Topic:
Interviewing



July 10 – Week 6

Functional Backend Hosts, SDKs, and Design Patterns

Presenters:
Multiple

Foundational Skills Topic:
Bringing Your Whole Self to Work



July 17 – Week 7

Security & Compliance

Presenters:
Multiple

Foundational Skills Topic:
Ask Me Anything



July 24 – Week 8

Cloud Native Development

Presenters:
Multiple

Foundational Skills Topic:
Graduation Celebration

Software Development

Executive Sponsor

Track Leader

College Sophomores

IBM Content
Instructors

IBM Coaches

IBM Coordinators

Software Development Session Structure –

Wednesdays, 6-8 PM Eastern

Pre-Session

- Up to 30 minutes of material to introduce that week's topic
- Specific tools needed for week
- Posted Monday before session

Lecture

- 1st hour of session
- Webex lecture with slides, demos
- Q&A ongoing via slack channel

Breakout & Lab

- 2nd hour of session
- Breakout rooms of 10 students with 2 coaches
- Practice exercise (lab) in GitHub

Post- session

- Attendance and NPS survey
- Optional practice test
- Up to 30 minutes of additional optional materials
- Lab solutions posted following week

Office Hour

- 1 hour
- Thursday (day after lecture)
- Optional session
- Hosted by that week's instructors

Slack Channel Q&A

- Ongoing Q&A
- Via slack channel
- Monitored by instructors & coaches

HackerRank Test

- Multiple choice
- 15-20 minutes
- After sessions 3, 6, 7, 8
- Must pass 3 of 4 to earn Badge
- Available after Thursday Office Hours
- Due the following Tuesday

Coding Practice Test

- Optional
- Given at end of Program
- Coding Test practice to prepare for Intern Application Assessment

IBM Accelerate Expectations & Badge Requirements

Attendance

- Attendance will be recorded each session
- Must **attend 6 of 7 sessions** (can miss 1 SW Track session)
- Encouraged to attend all sessions as they build upon each other

Participation

- We expect you to be an active learner
- Comment or ask questions during discussions
- Collaborate with peers in break-outs
- Take advantage of optional networking opportunities
- Recommended to have your camera on

SW Track Skills

- Track Assessment will be measured by 4 'graded' tests
- After sessions 3, 6, 7, and 8
- HackerRank
- Multiple choice
- 15-20 minutes
- Must **pass 3 of 4 tests** to earn Badge

Session Agenda

- 1 Frontend Technology Overview
- 2 React
- 3 Version Control
- 4 Basic Project Setup
- 5 Q&A
- 6 Lab

Frontend Technology by Overview



What Are Frontend Technologies?

The tools, programming languages, frameworks, and libraries used in frontend development.

For the creation of the visual and interactive components of a software application or website.

Frontend developers typically work with HTML, CSS, and JavaScript.



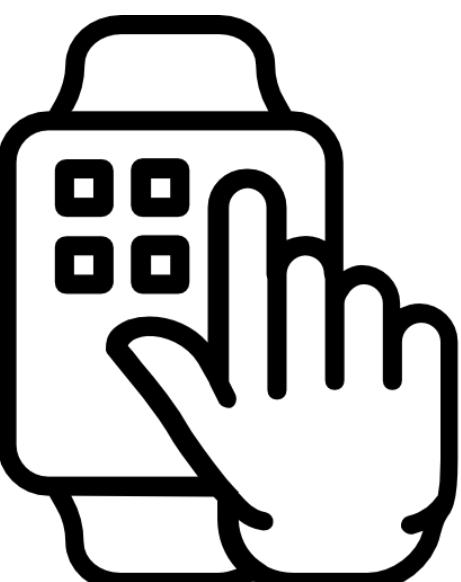
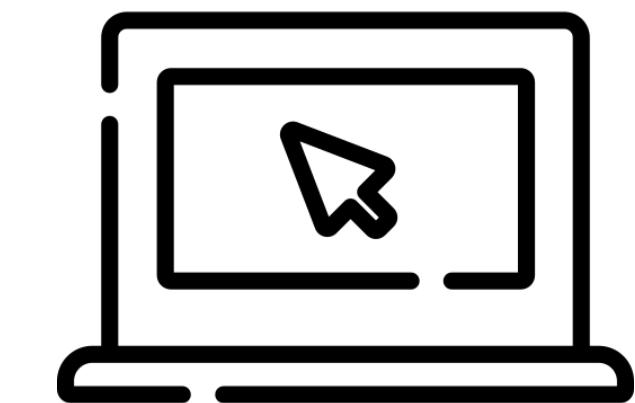
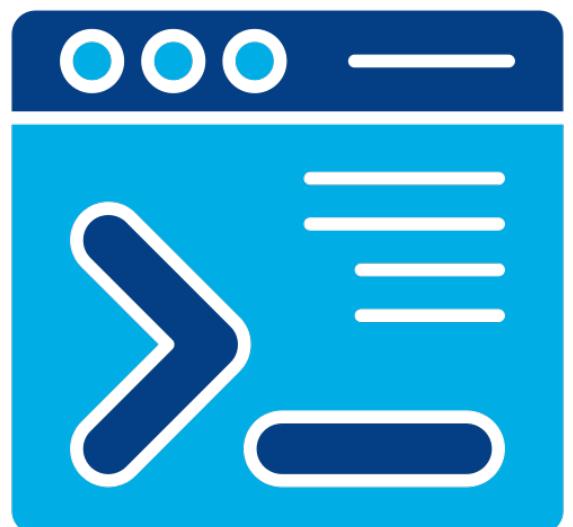
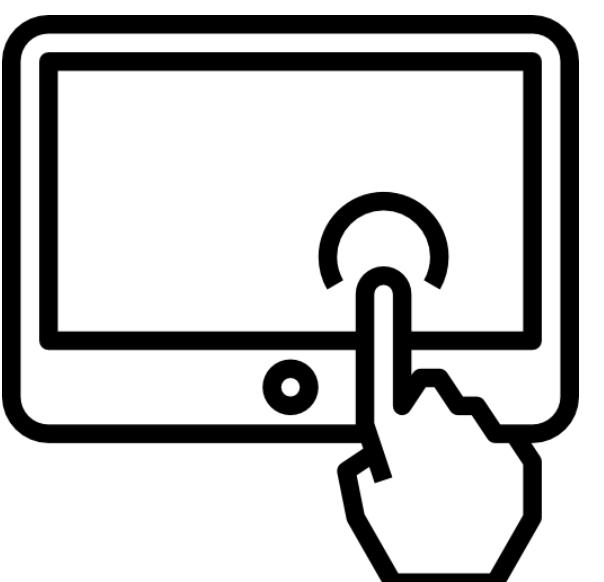
User Interface (UI)

A client-side layer of an application or system

The User Interface (UI) refers to the visual and interactive components the output of frontend development

Some different types of UIs include

- Graphical User Interface (GUI)
- Command-Line Interface (CLI)
- Touch User Interface (TUI)
- Voice User Interface (VUI)

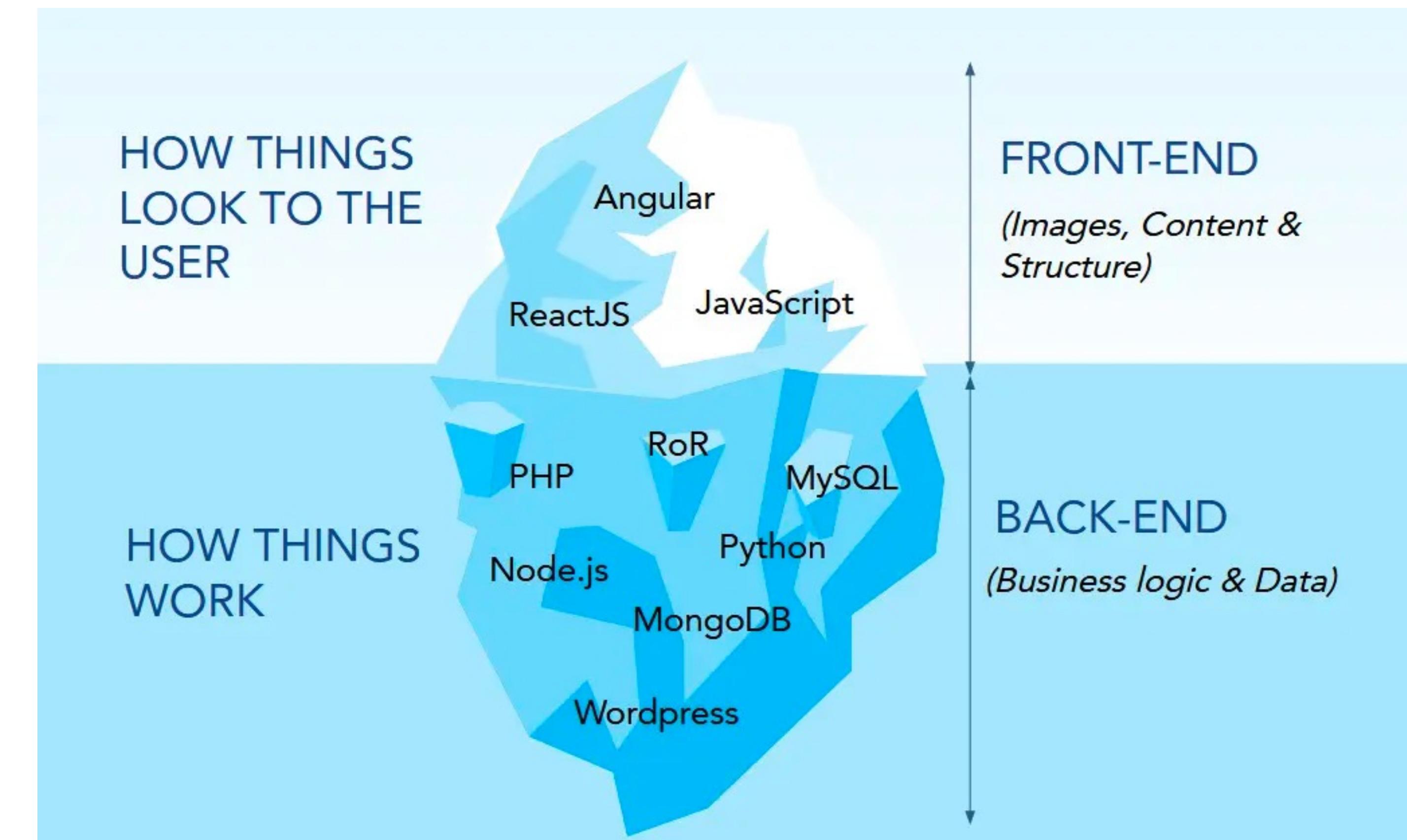


Web Development

In the IBM Accelerate program we are walking through the development of a web application.

Web development can be split into two main components:

- Frontend development (Client-Side)
- Back-end development (Server-Side)



Web Development Tools

At the heart of web development are three main languages:

HTML

- The content of a webpage

CSS

- Provides styling and layout

Javascript

- Controls behavior of DOM elements

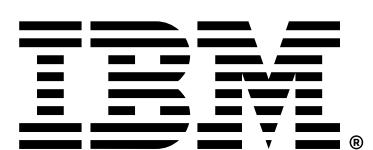


HTML

HyperText Markup Language

- HTML is the standard markup language for creating web page applications.
- It tells the browser how to display contents on a webpage (the structure).

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4  <title>Advanced Demo with DOM Elements</title>
5  <link rel="stylesheet" type="text/css" href="styles.css">
6  </head>
7  <body>
8  <h1>Advanced Demo with DOM Elements</h1>
9
10 <div class="form-container">
11   <label for="colorSelect">Select a color:</label>
12   <select id="colorSelect" class="dropdown">
13     <option value="blue">Blue</option>
14     <option value="green">Green</option>
15     <option value="red">Red</option>
16   </select>
17
18   <label for="nameInput">Enter your name:</label>
19   <input type="text" id="nameInput" class="inputField">
20
21   <label for="messageTextArea">Enter your message:</label>
22   <textarea id="messageTextArea" class="textField" placeholder="Enter your message">
23   </textarea>
```



CSS

Cascade Style Sheet

```
1  /*  
2   selector {  
3     property: value;  
4     property: value;  
5     .....  
6     property: value;  
7   }  
8 */  
9  
10 <body> {  
11   background-color: aquamarine;  
12 }  
13  
14 <h1> {  
15   color: olivedrab;  
16   text-align: center;  
17 }  
18  
19 <p> {  
20   font-family: Arial, Helvetica, sans-serif;  
21   font-size: 20px;  
22 }
```

- Specifically describes *how* information is displayed and not *what* information is displayed.
- Can be included in the HTML document or used in a separate .css file.
- A CSS file has one or more rules that start with a selector.
- A selector specifies an HTML element and applies a style property to it.

JavaScript

```
var variables = runtimeScene.getVariables();
var speech = variables.get("speech").getAsString();

speak(speech);

// say a message
function speak(text, callback) {
    var u = new SpeechSynthesisUtterance();
    u.text = text;
    u.lang = 'en-US';

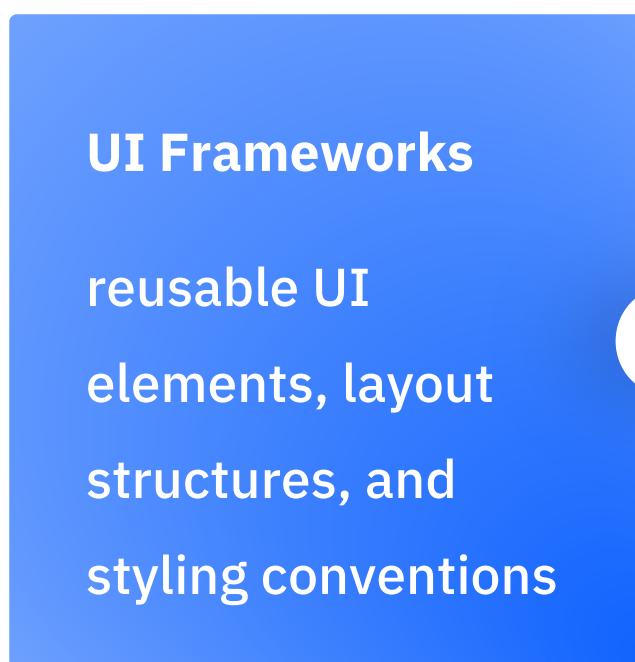
    u.onend = function () {
        if (callback) {
            callback();
        }
    }

    u.onerror = function (e) {
        if (callback) {
            callback(e);
        }
    }
}
```

- JavaScript is a scripting language
- It gives the computer a series of instructions to perform
- Interacts with the user, the page, and the internet

Frameworks

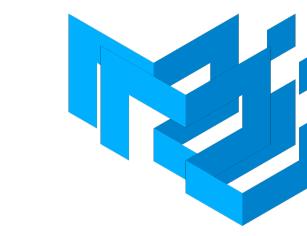
A framework is a software development platform or environment that provides a structured and reusable set of tools, components, and libraries for building applications.



Bootstrap



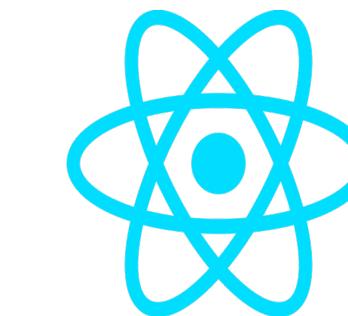
IBM Carbon



Material UI



Angular



React



Vue

HTML, JS, CSS DEMO

A screenshot of a web browser window displaying a CodePen demo page. The browser interface includes a header with tabs, a search bar, and user account options. The main content area shows an HTML editor with a dark theme, displaying a basic HTML structure. To the right of the editor is a preview area titled "HTML, JavaScript, CSS Demo". This preview area contains a form with dropdown and input fields, and a large text area. Below the form is a section with three blue links labeled "Link 1", "Link 2", and "Link 3". At the bottom of the preview area are two buttons: "Click Me" (blue) and "Click Me - HTML Body" (green). The bottom of the browser window features navigation links for "Console", "Assets", "Comments", "Keys", and sharing options.

Accelerate Week 1

https://codepen.io/Kulube/pen/PoyaaKO?editors=1000

Accelerate Week 1

HTML

```
1 <!DOCTYPE html>
2 <html>
3 <head></head>
4 <body></body>
5 </html>
```

HTML, JavaScript, CSS Demo

Select a color:

Blue

Enter your name:

Enter your message here:

Enter your message

Link 1 Link 2 Link 3

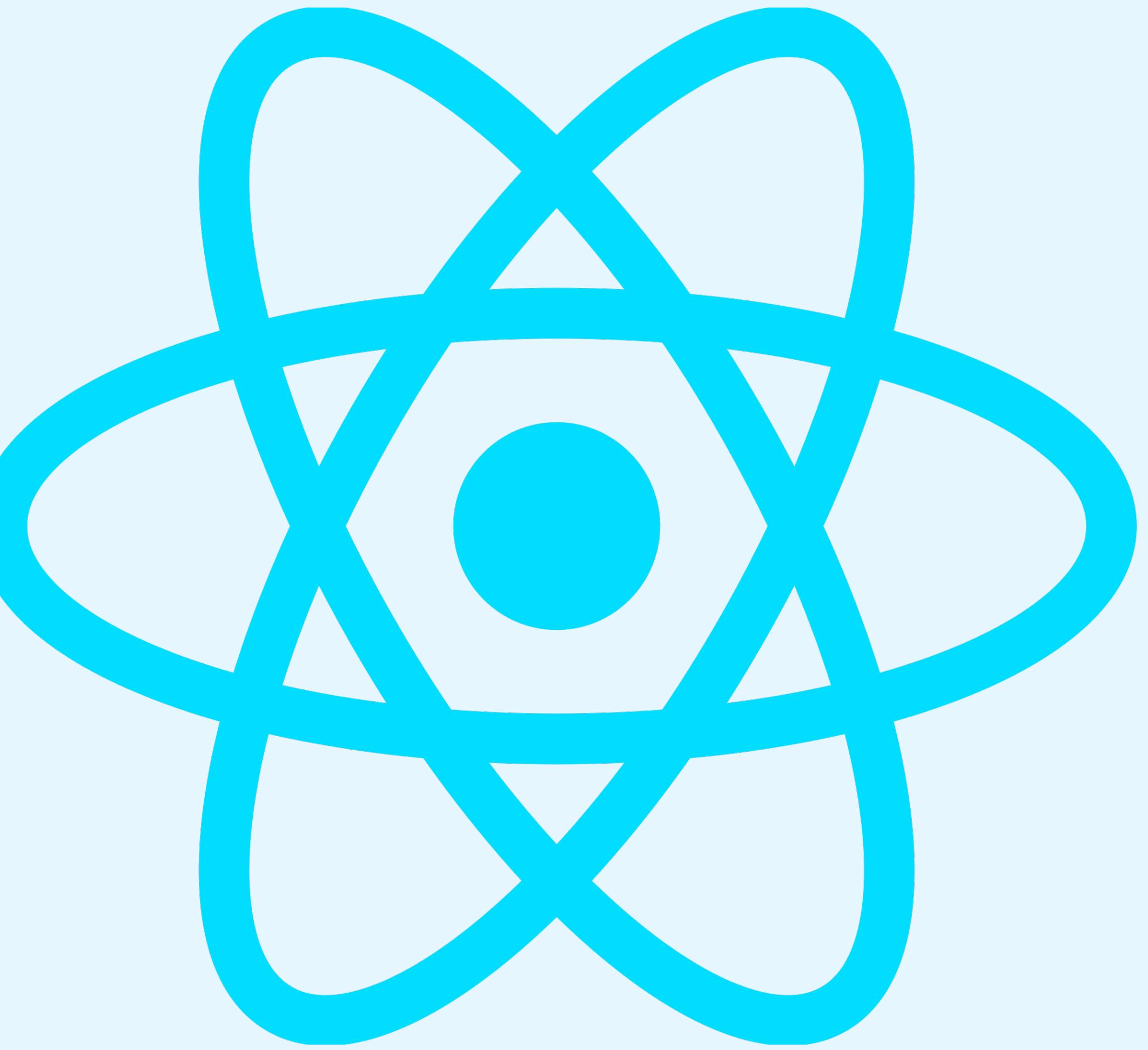
Click Me Click Me - HTML Body

Console Assets Comments Keys

Fork Embed Export Share



React



What is React and why should you use it?

What is React?

React is a JavaScript library developed by Facebook that is used for building UI components.

Why React?

- Simplicity
- Flexible
- Easy to learn
- Performance
- Virtual DOM
- Large community for support / resources

React Examples

React DOM takes care of updating the browser DOM to match the React Elements.

```
import react from "react";
import ReactDOM from "react-dom";

function Music() {
  return <h1>Micheal Jackson - Human Nature</h1>;
}

ReactDOM.render(<Music />, document.getElementById("root"));
```

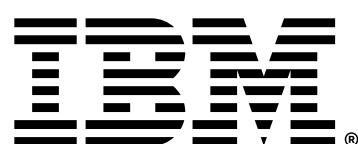
Whenever the state changes on a component, the React DOM will automatically re-render and update the browser DOM.

```
import react, { useState } from 'react';

function LikeButton() {
  const [likeCount, setLikeCount] = useState(0);

  function handleClick() {
    setLikeCount(likeCount + 1);
  }

  return {
    <button onClick={handleClick}>
      Song has {likeCount} likes!
    </button>
  };
}
```



React Project Structure

Each project structure varies and there's no one way to organize your project.

Components – Custom components

Pages – Each page in the app

Hooks – Custom hooks

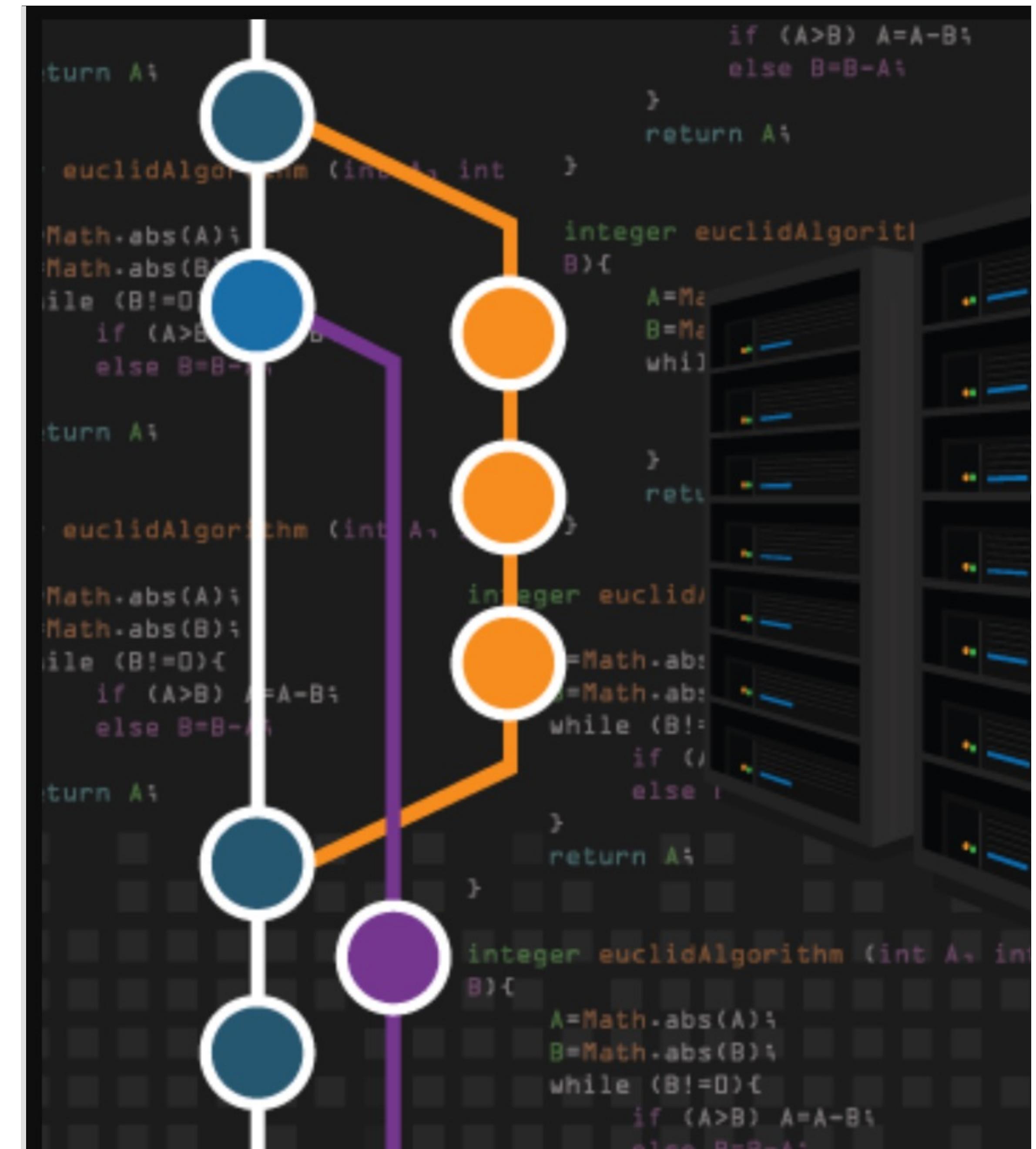
Styles – Styling files like CSS or SCSS

Tests – Test suite of the project

```
> node_modules  
> public  
└ src  
    └ components  
        JS Button.js  
        JS FormInput.js  
    > hooks  
    └ pages  
        JS UserProfile.js  
    > styles  
    > tests  
# App.css  
JS App.js  
JS App.test.js  
# index.css  
⚙️ index.jsx  
JS setupTests.js  
{ } package-lock.json  
{ } package.json  
ⓘ README.md
```



Version Control



The practice of
tracking and managing
changes to software
and code



Why is version control useful?

Tracking

- Who made a change to the code
- What the change was
- When the change occurred

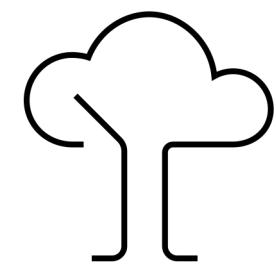
Managing

- Create independent streams of changes
- Merge changes back together
- Create releases, Kanban boards, issues, and more

Deploying

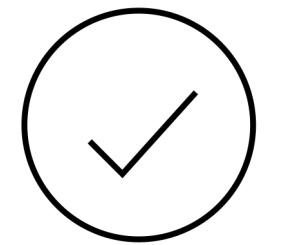
- Automate verification of code
- Automate the building of code
- Automate the testing of code

Common Terms



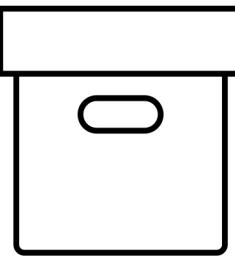
Branch

A new line of development that diverges from the main line of development.



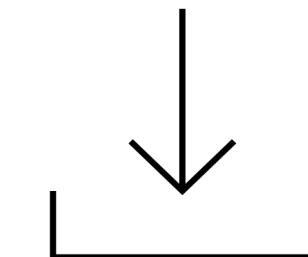
Commit

Save the current state of your changes.



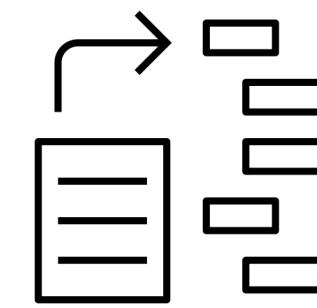
Repository (repo)

Used to organize projects and can contain folders, files, images, videos, spreadsheets, and more.



Clone

Creates a copy of a repo on your local machine.



Merge

The processing of combining multiple changes made to different copies of the same file in the source repo.

How does it work?

1

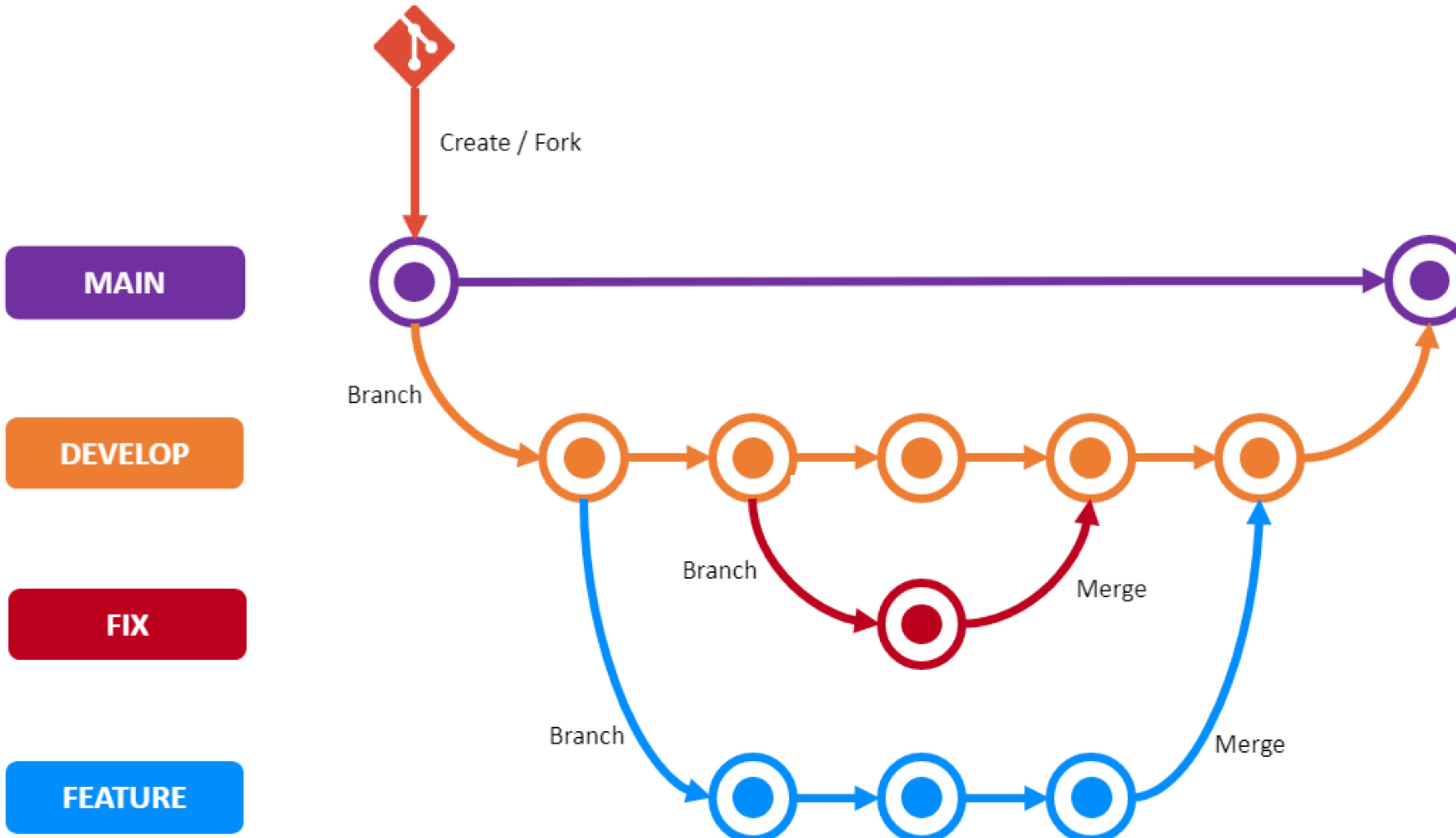
Create a branch

2

Make and commit changes

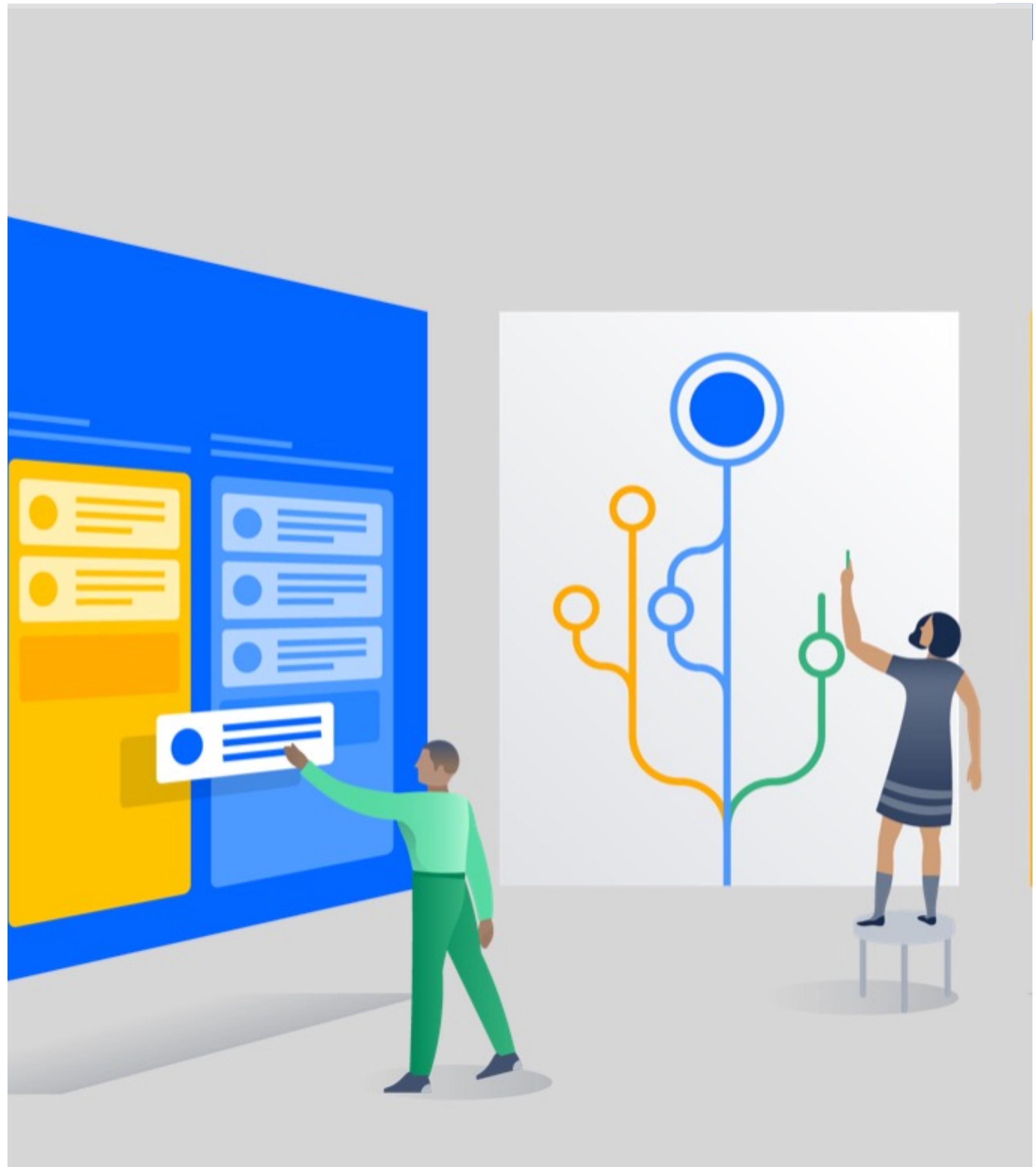
3

Open a pull request



The IBM logo consists of the letters "IBM" in a bold, stylized font made of horizontal bars. A registered trademark symbol (®) is located at the bottom right of the letters.

Basic Project Setup



Steps

1.

Download and install
node and npm

2.

Create a React
project

3.

Connect version
control

Download the Node.js installer

Go to <https://nodejs.org/en/download> and download the matching installer for your PC.



Install Node.js and NPM

Run the installation in the terminal
command prompt:

```
npm install
```



Verify the installation

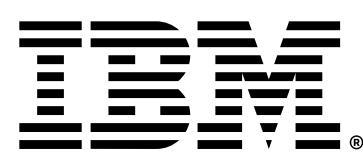
Run the following in a terminal command prompt:

npm -v

node -v

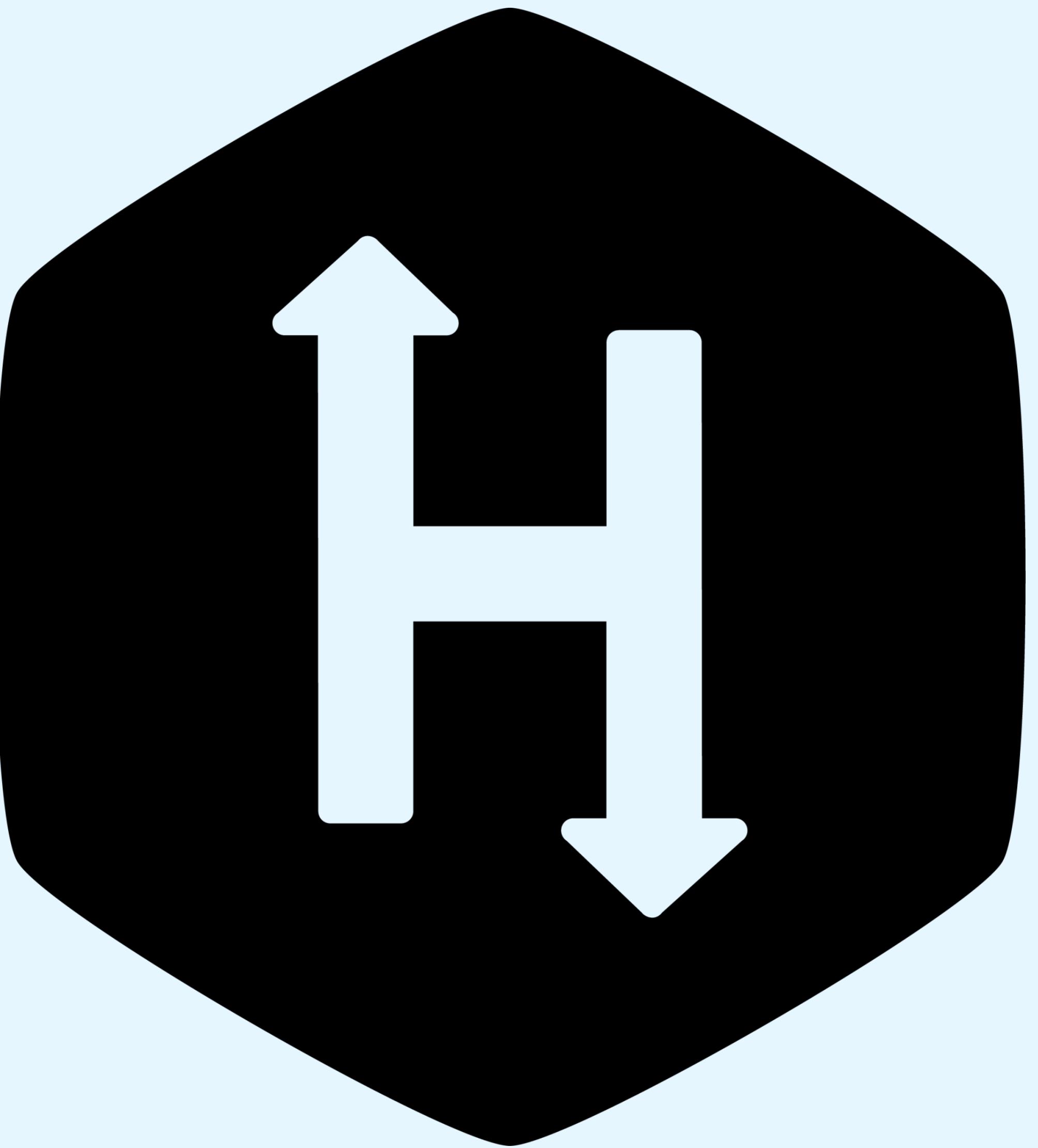


Q&A



HackerRank

Practice Test



HackerRank Practice Test



Hey Sujeily.fonseca,

Welcome to 2024 IBM Accelerate: Software Week 1

Test duration	No. of questions
45 mins	5 questions

Instructions

1. This is a timed test. Please make sure you are not interrupted during the test, as the timer cannot be paused once started.
2. Please ensure you have a stable internet connection.
3. We recommend you to try the [sample test](#) for a couple of minutes, before taking the main test.
4. Before taking the test, please go through the [FAQs](#) to resolve your queries related to the test or the HackerRank platform.

[Continue](#)

[Try Sample Test](#)

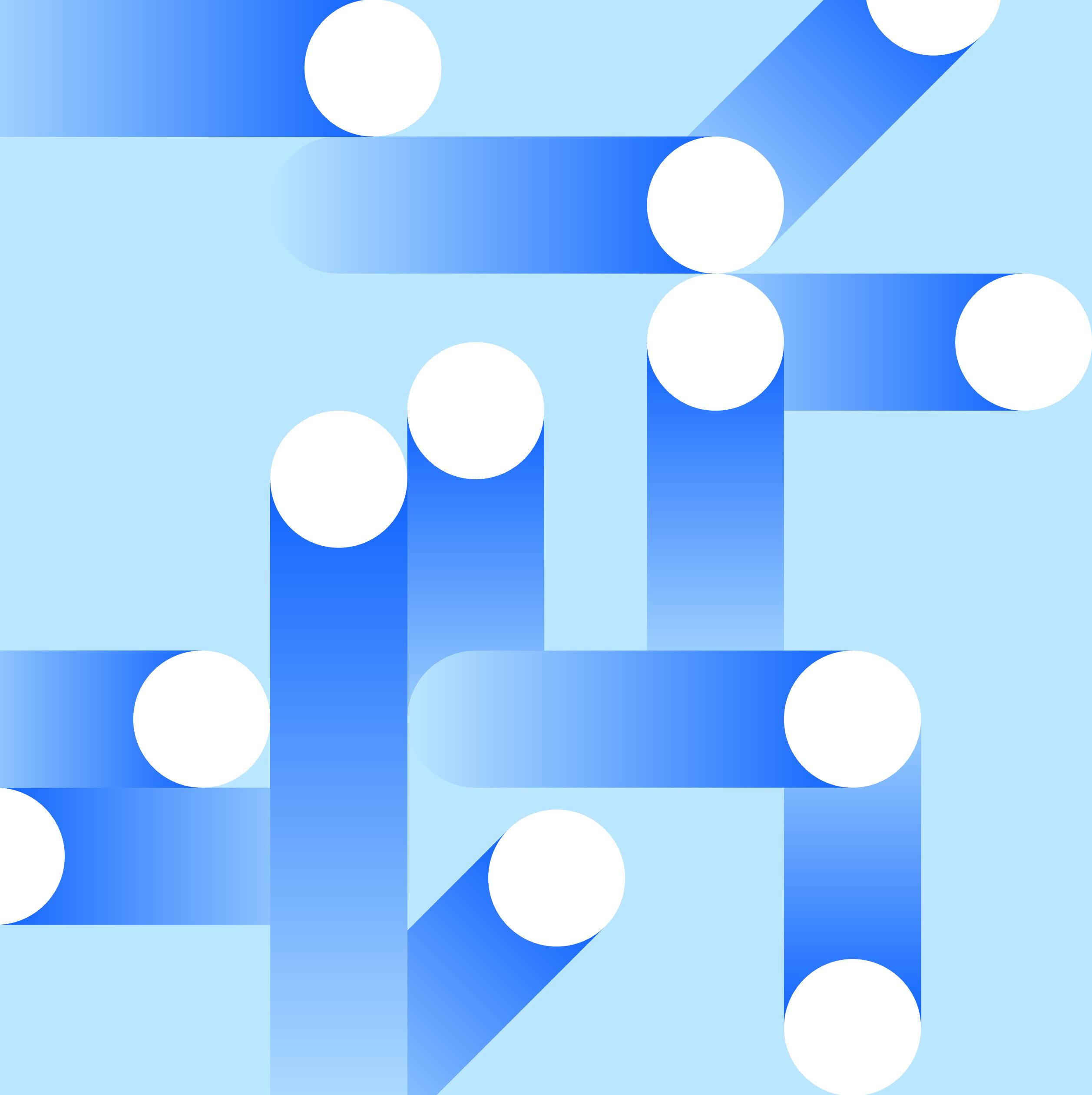
[Platform Help](#) | [Execution Environment](#) | [FAQ](#)

Project/Lab and GitHub Classroom



Lab/Project 2: To-do-list

Due: June 11, 2024



To-do-list Lab:

Review

- Submissions for Lab/Project 1 will close June 11, 2024, 11:59 PM ET
 - We will pull the code from your master/main branch at this time to provide feedback.
 - Make sure you merge any branches before then.
- Use the Slack channel and office hours for any questions.



To-do-list Lab:

Instructions

Feature Requirements:

- Design your personalize about me page.
- Take in and display to do list items.

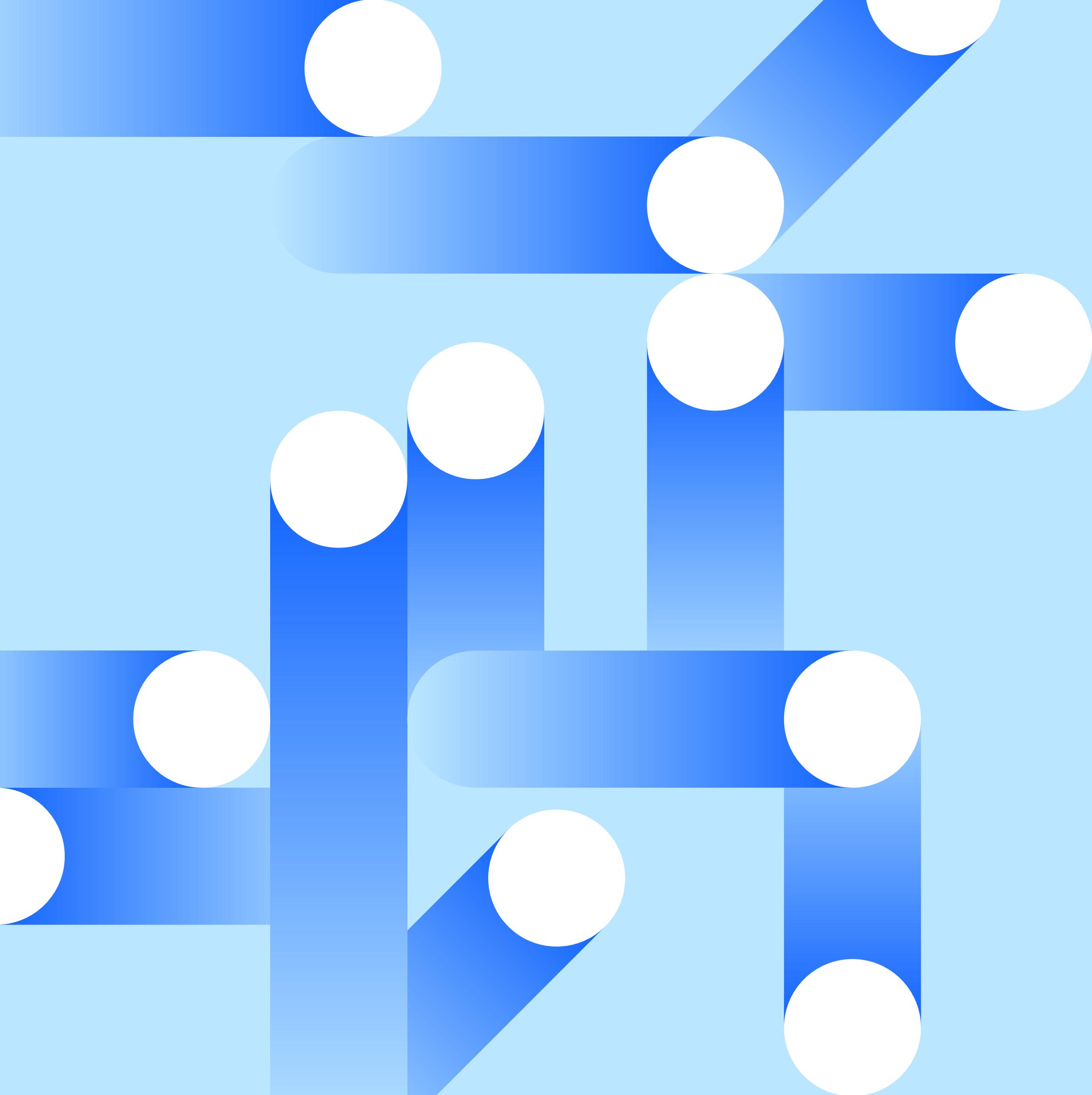
Implementation Requirements:

- Use [Material UI components](#) at least once throughout the app.
- Implement at least one functional component.

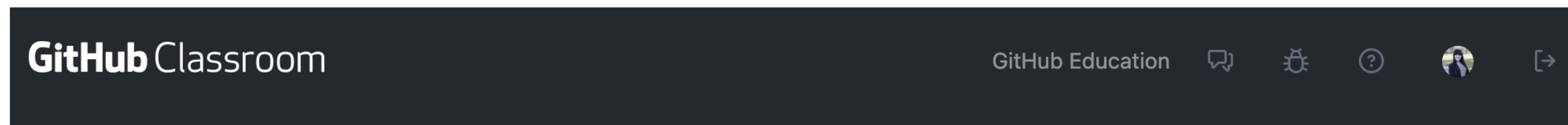


GitHub Classroom:

Let's walk through the process together!



GitHub Classroom: Let's walk through the process together!



2024-IBM-Accelerate-SW-Track-classroom

Accept the assignment – [to-do-list_week-1](#)

Once you accept this assignment, you will be granted access to the [to-do-list-week-1-sujeilyfonseca](#) repository in the [2024-IBM-Accelerate-SW-Track](#) organization on GitHub.

[Accept this assignment](#)

- Short link: <https://ibm.biz/accelerate-software-week1>
- Direct link: <https://classroom.github.com/a/rsLZezik>

GitHub Classroom: Automated feedback and code validation

The screenshot shows a GitHub repository page for '2023-IBM-Accelerate-SW-Track / to-do-list'. The repository has 17 branches and 0 tags. The 'Code' tab is selected. A message indicates the branch is 56 commits ahead of main. A commit by sujellyfonseca is shown, updating README.md. The commit message is 'docs: Updated README.md (#9)'. The commit was made 2 days ago by user 789d810 with 59 commits. The commit details show files like public, src, .gitignore, README.md, package-lock.json, and package.json being updated. Below the commit list is a file editor for README.md, which contains the heading 'Project Week 2: To-do list application (Cont.)' and an 'Introduction' section. The introduction text describes the goal of Project Week 2, mentioning the need to implement more Material UI components and give the application a sleek appearance. The repository's 'About' section notes there is no description, website, or topics provided. It includes links to Readme, Activity, and a star count of 0. The 'Releases' section shows no releases published and a link to 'Create a new release'. The 'Packages' section shows no packages published and a link to 'Publish your first package'. The 'Languages' section shows the code distribution: JavaScript (47.1%), HTML (42.0%), and CSS (10.9%).

- Automated grading/feedback was enabled.
- It will also give help you understand how you're doing.
- You can run the tests locally by executing “npm test”.