College of Engineering
Computer Science & Eng. Dept.
Course: CMP 257L
Web App Programming Lab

Course Professor: Dr. Dana Dghaym
Lab Instructor: Khaldoon Alhusari
Office: EB2-117
Phone: 971-6-515-4819
e-mail: kalhusari@aus.edu
Semester: Fall 2025

Name: Fatmah Alyammahi ID: g0010002
Name: Maryam Ali ID: g00093757

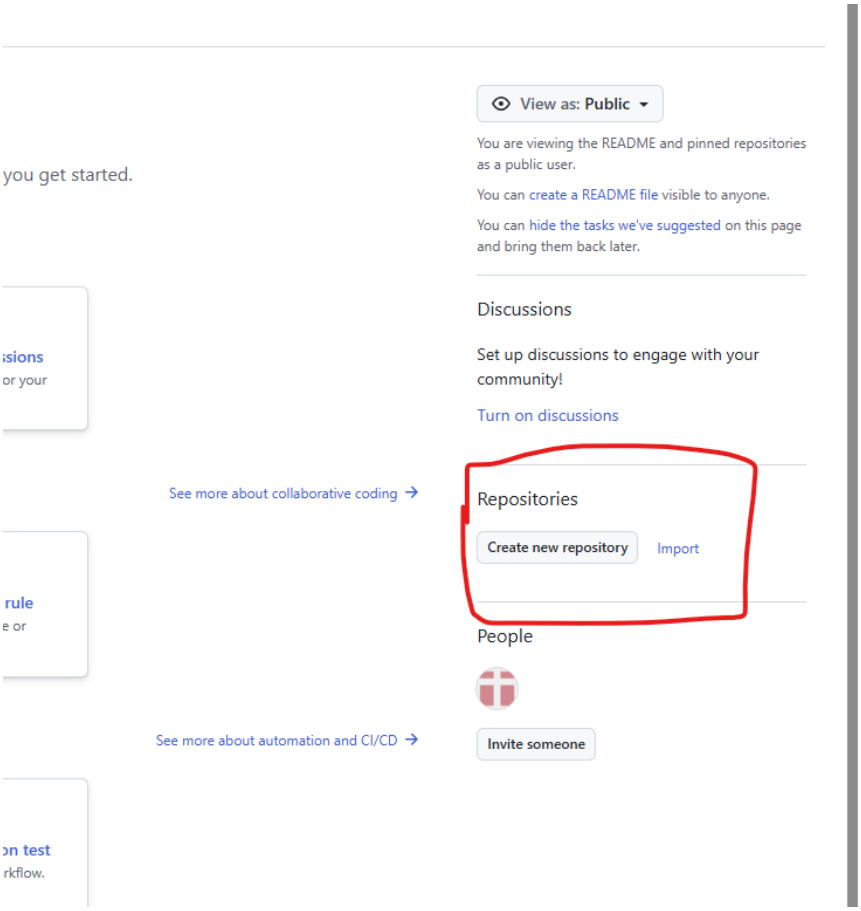
Lab 4: Introduction to GitHub and JavaScript

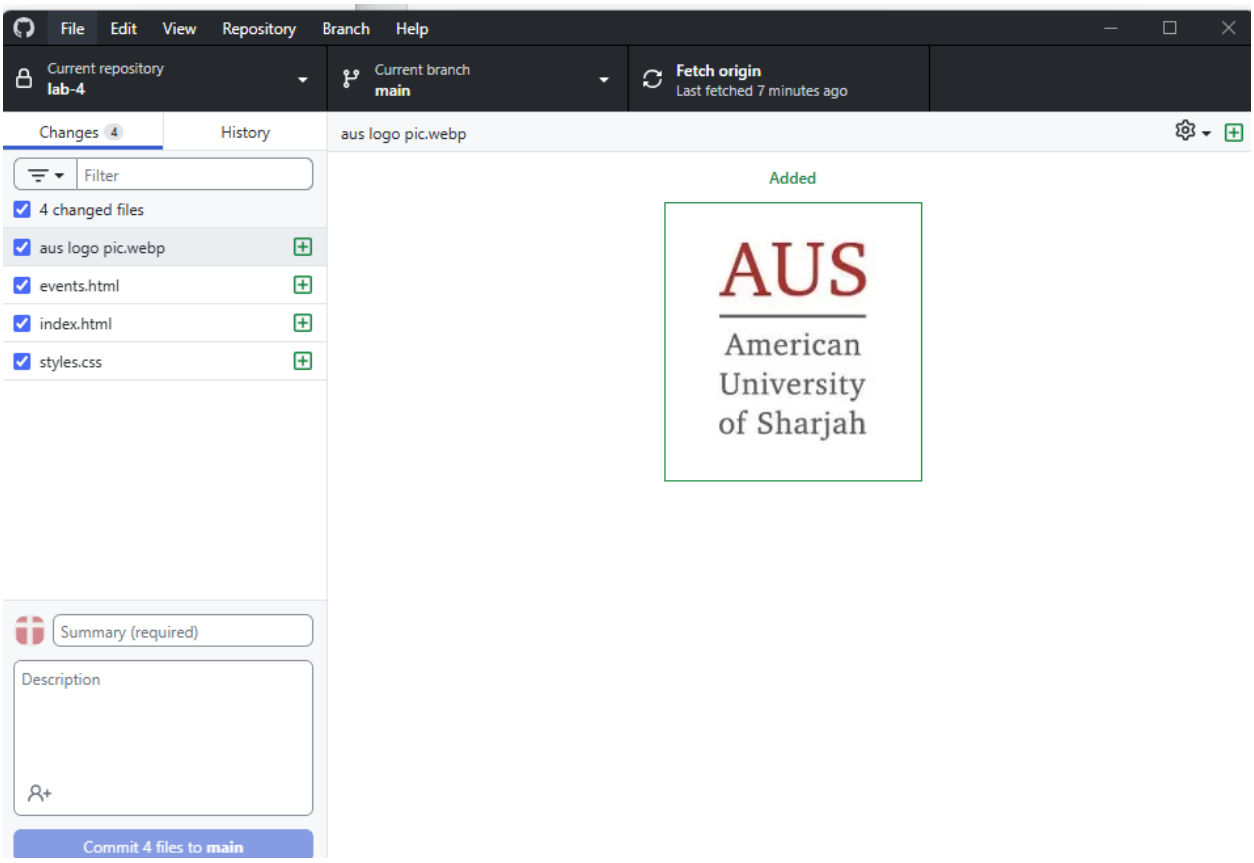
- **Objectives:**
  - Introduction to GitHub
  - Introduction to JavaScript

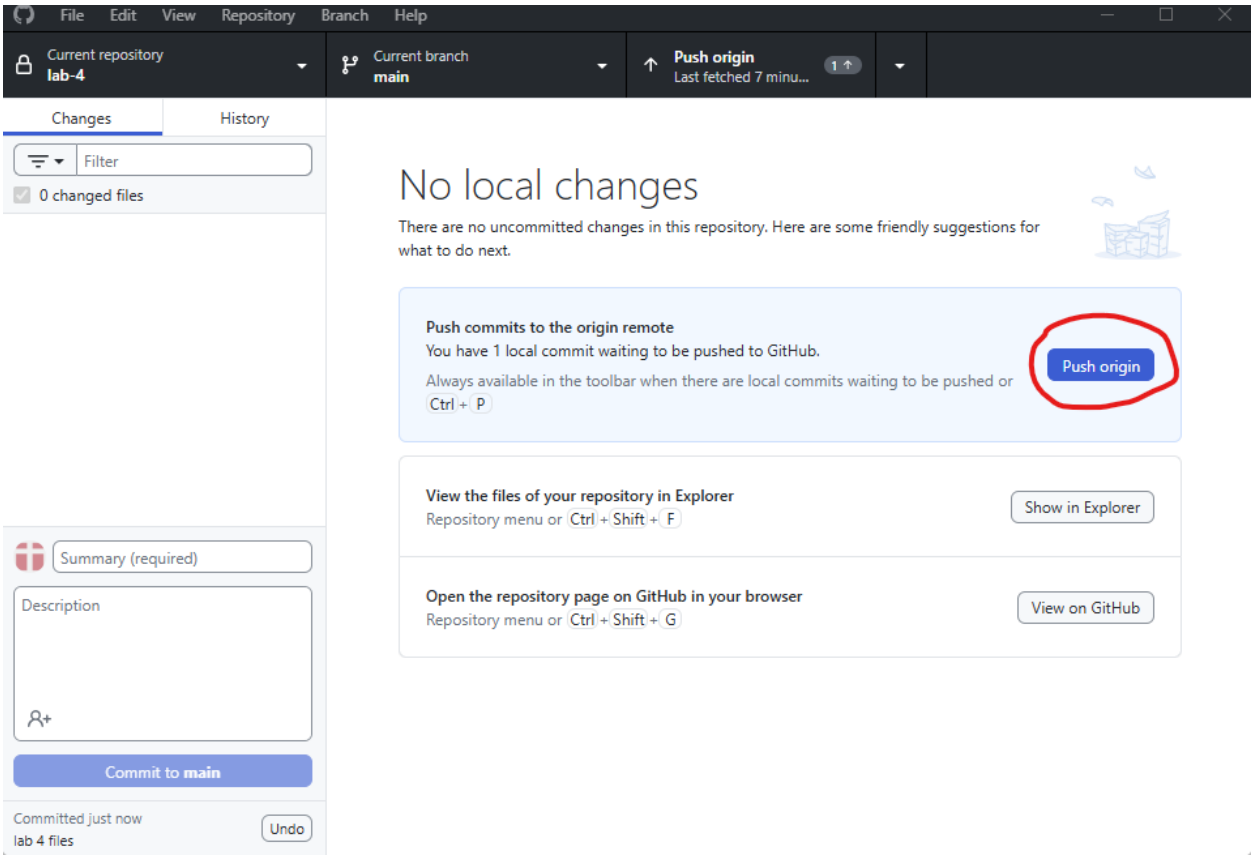
Exercise 1: Github [3 marks]

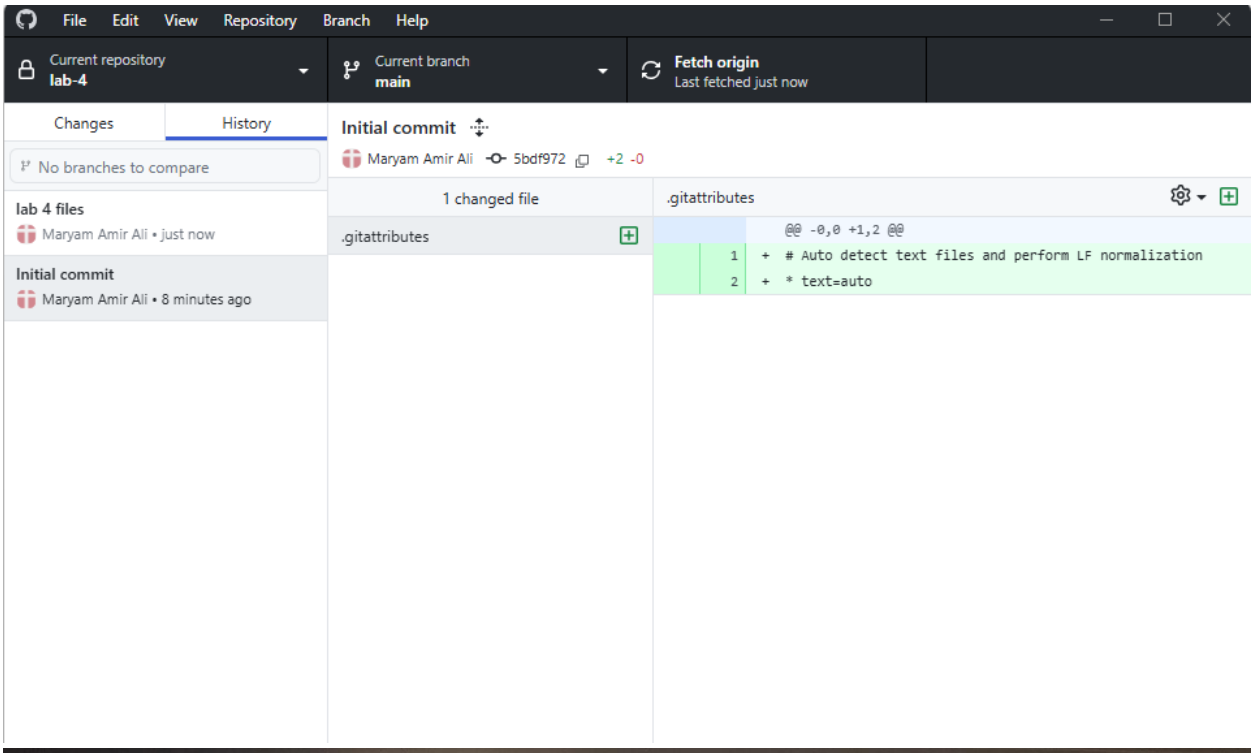
Use the project you've been working on for the last few labs for this exercise.
Using the accompanying document with GitHub instructions, create a repository for your website. Add your lab partner to the repository and then have him/her clone your repository, make a small change, and push.
Pull the updated code on your side.
Give screenshots of all the steps you followed, including creating the repo, cloning, committing, pushing, and pulling.

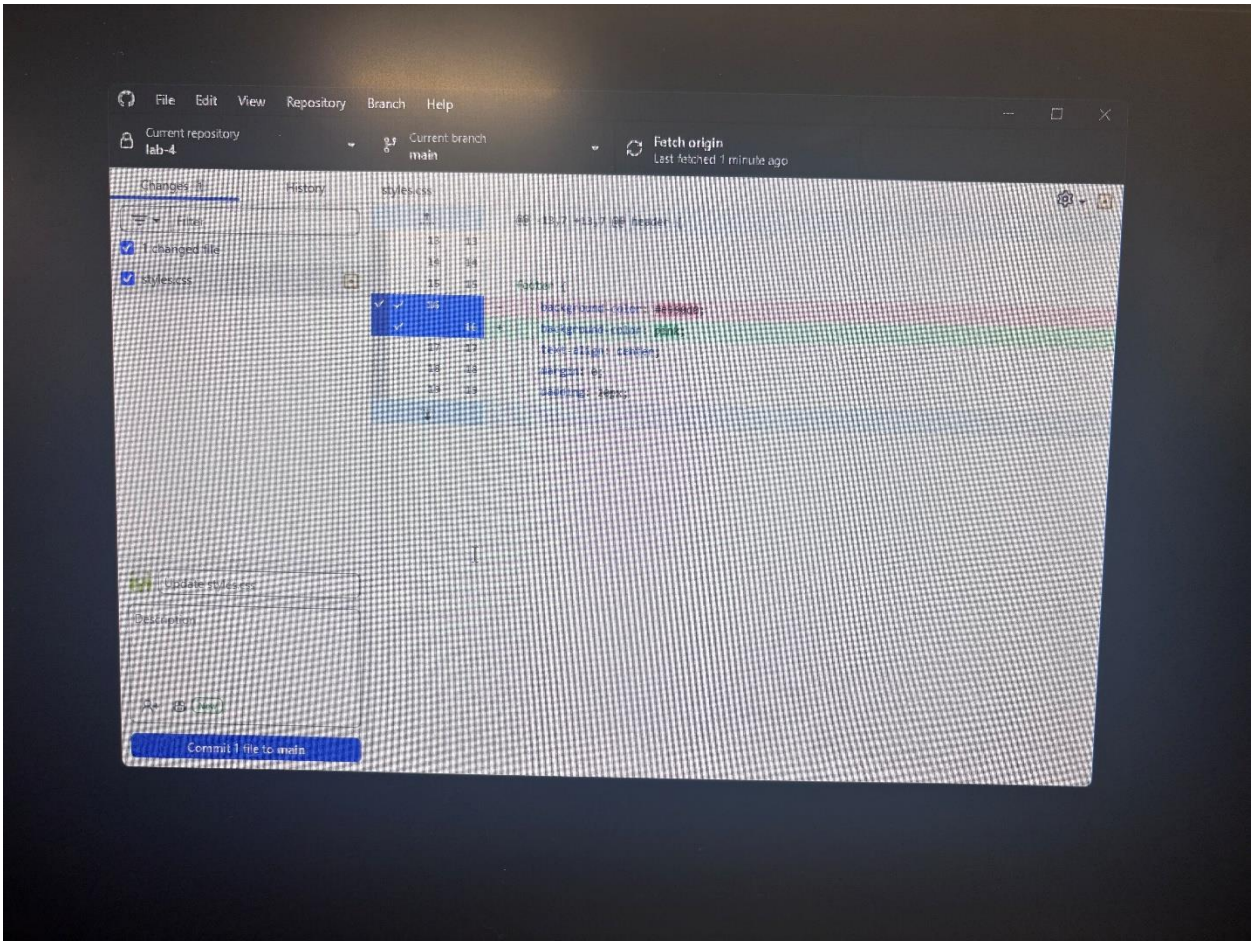
Screenshot:

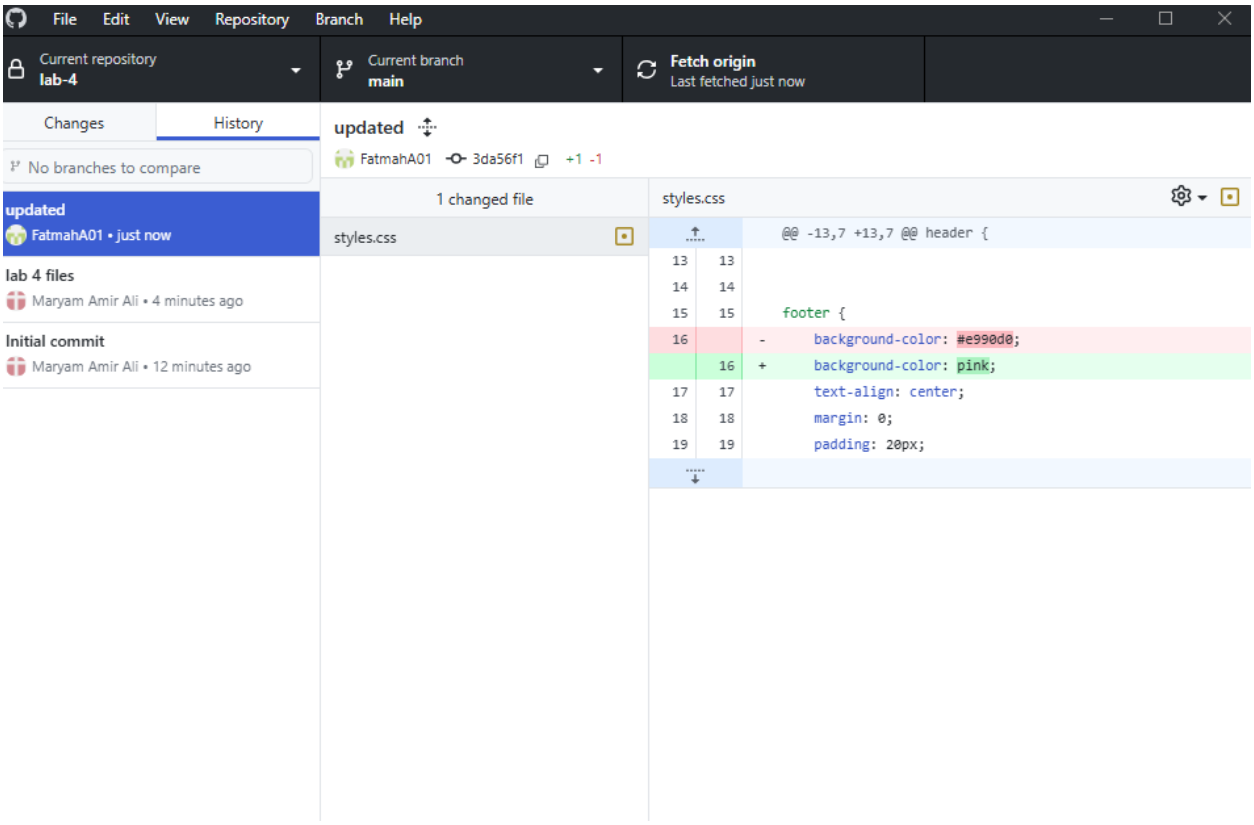

you get started.
View as: Public
You are viewing the README and pinned repositories as a public user.
You can create a README file visible to anyone.
You can hide the tasks we've suggested on this page and bring them back later.
Discussions
Set up discussions to engage with your community!
Turn on discussions
See more about collaborative coding
Repositories
Create new repository
Import
People
See more about automation and CI/CD
Invite someone


File
Edit
View
Repository
Branch
Help
Current repository
lab-4
Current branch
main
Fetch origin
Last fetched 7 minutes ago
Changes 4
History
aus logo pic.webp
Filter
4 changed files
aus logo pic.webp
events.html
index.html
styles.css
Added
AUS
American
University
of Sharjah
Summary (required)
Description
Commit 4 files to main


File
Edit
View
Repository
Branch
Help
Current repository
lab-4
Current branch
main
Push origin
Last fetched 7 minu...
1
Changes
History
Filter
0 changed files
No local changes
There are no uncommitted changes in this repository. Here are some friendly suggestions for what to do next.
Push commits to the origin remote
You have 1 local commit waiting to be pushed to GitHub.
Always available in the toolbar when there are local commits waiting to be pushed or Ctrl + P
Push origin
View the files of your repository in Explorer
Repository menu or Ctrl + Shift + F
Show in Explorer
Open the repository page on GitHub in your browser
Repository menu or Ctrl + Shift + G
View on GitHub
Summary (required)
Description
Commit to main
Committed just now
lab 4 files
Undo


File Edit View Repository Branch Help
Current repository
lab-4
Current branch
main
Fetch origin
Last fetched just now
Changes
History
No branches to compare
lab 4 files
Maryam Amir Ali • just now
Initial commit
Maryam Amir Ali • 8 minutes ago
Initial commit
Maryam Amir Ali 5bdf972 +2 -0
1 changed file
.gitattributes
.gitattributes
@@ -0,0 +1,2 @@
1 + # Auto detect text files and perform LF normalization
2 + * text=auto


File Edit View Repository Branch Help
Current repository
lab-4
Current branch
main
Fetch origin
Last fetched 1 minute ago
Commit 1 file to main



Exercise 2: JavaScript [7 marks]

Solve the following exercises using any JavaScript compiler of your choice. You can use the console within your browser or online compilers such as https://jsbin.com/

For each sub exercise, paste the code and a screenshot of your output.

2a. How Many Pizzas?

- Imagine you have **3 slices of pizza**, but you're really hungry and order **2 more pizzas**. Each pizza has **8 slices**. You want to calculate the total number of slices you have.
- Create a program that initializes variables for initial and final number of slices and calculates how many pizza slices you have in total.
- Print out a message to the console informing the user of the number of slices eg. "I have X slices of pizza. Time for a pizza party!"

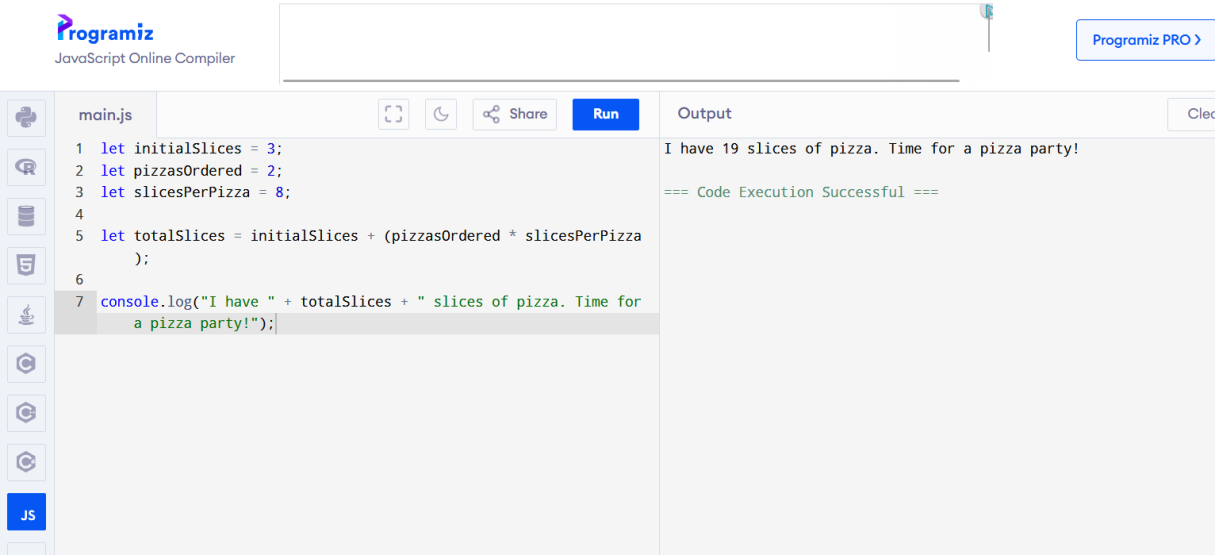
Code:

```
let initialSlices = 3;
let pizzasOrdered = 2;
let slicesPerPizza = 8;
```

let totalSlices = initialSlices + (pizzasOrdered * slicesPerPizza);

console.log("I have " + totalSlices + " slices of pizza. Time for a pizza party!");

Screenshot:



2b. The Procrastination Timer

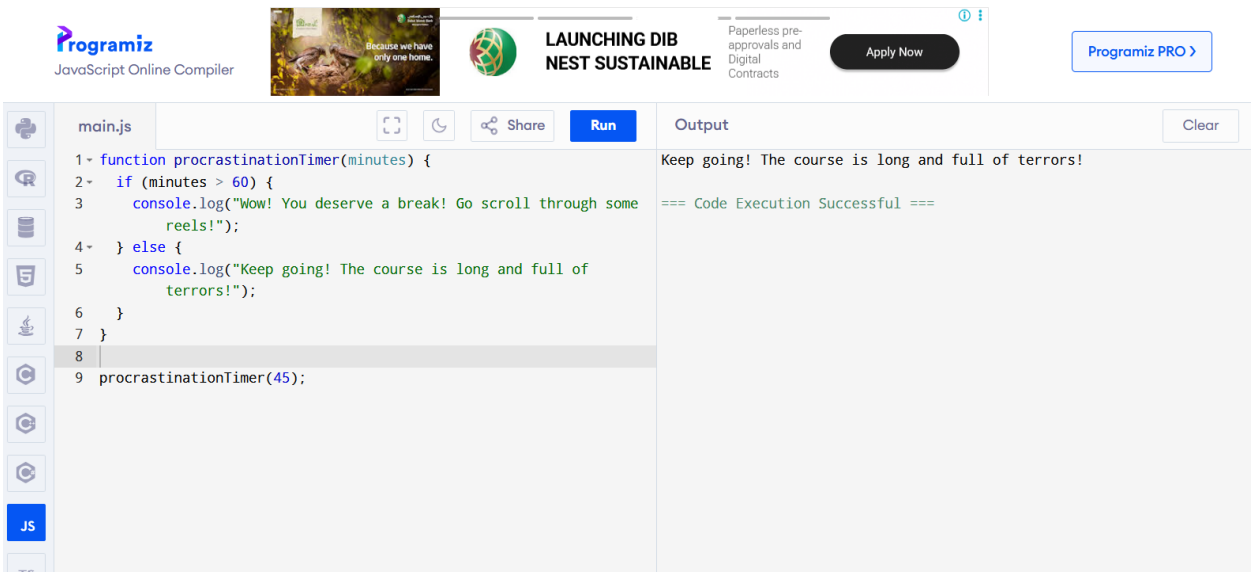
- Write a function that takes the number of minutes you've already studied as input.
- If you've studied for more than 60 minutes, log "Wow! You deserve a break! Go scroll through some reels!"
- If you've studied for less than 60 minutes, log "Keep going! The course is long and full of terrors!"

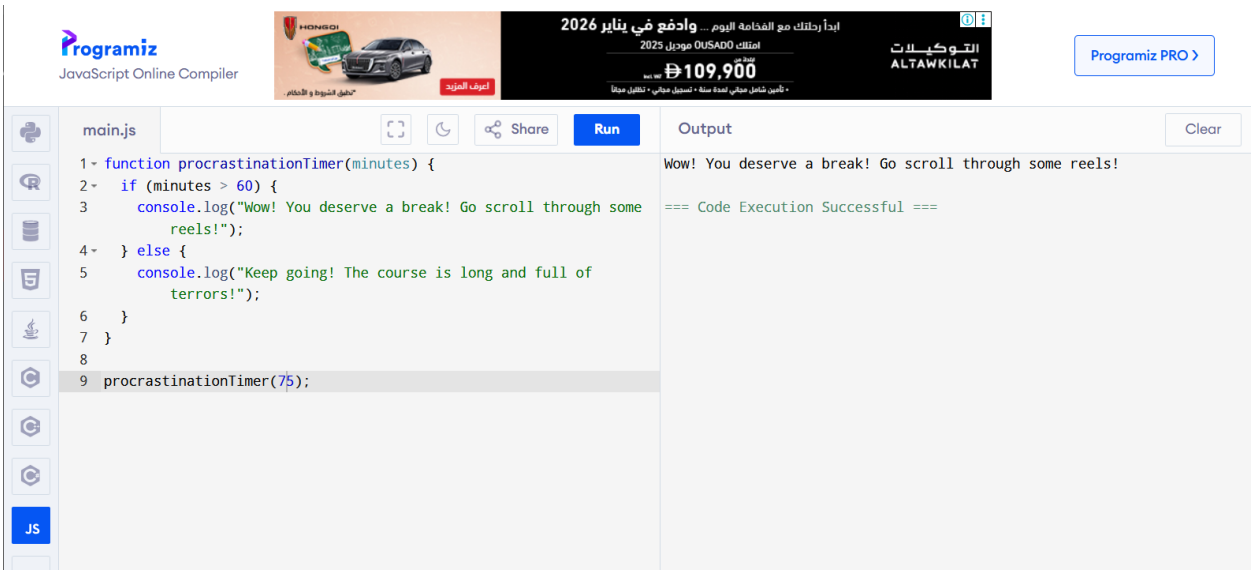
Code:

```
function procrastinationTimer(minutes) {
  if (minutes > 60) {
    console.log("Wow! You deserve a break! Go scroll through some reels!");
  } else {
    console.log("Keep going! The course is long and full of terrors!");
  }
}

procrastinationTimer(75);
```

Screenshot:





2c. Coffee Addiction Tracker

- You need to stay awake for **5 days** of finals week, and you drink **3 cups of coffee per day** to survive.
- Write a while loop that calculates how many cups of coffee you'll drink in total over the 5 days.
- Display an alert like "You'll need X cups of coffee to survive finals week!"

Code:

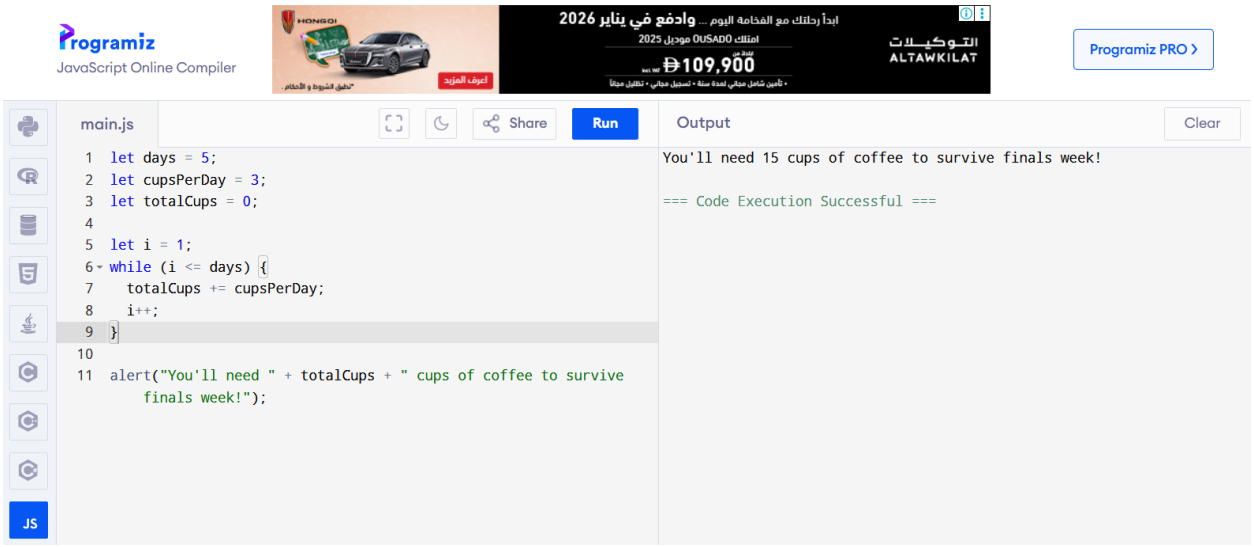
```
let days = 5;
let cupsPerDay = 3;
```

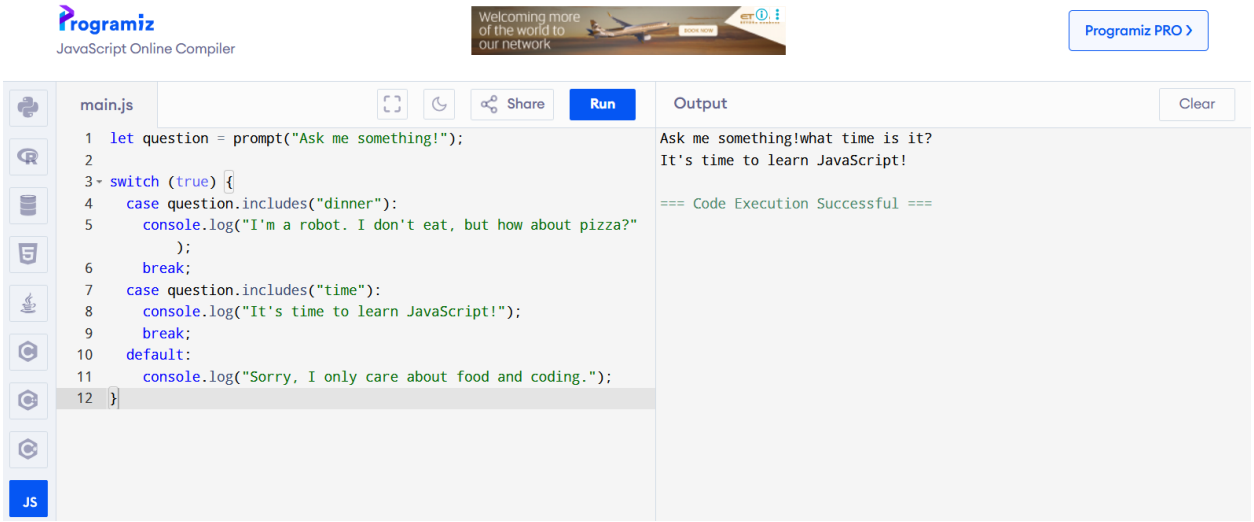
```
let totalCups = 0;

let i = 1;
while (i <= days) {
  totalCups += cupsPerDay;
  i++;
}

alert("You'll need " + totalCups + " cups of coffee to survive finals week!");
```

Screenshot:





2d. The Bot

Since you love talking to AI, let's create a simple bot.

- Create a program that simulates a conversation with your AI bot.
- Use a prompt to get a question from the user.
- Use a `switch` statement to give different responses based on what you ask:
  - If you ask "What's for dinner?", the bot says, "I'm a robot. I don't eat, but how about pizza?"
  - If you ask "What time is it?", the bot says, "It's time to learn JavaScript!".
  - If you ask anything else, the bot responds, "Sorry, I only care about food and coding."
- Use the `includes()` method to check for keywords.

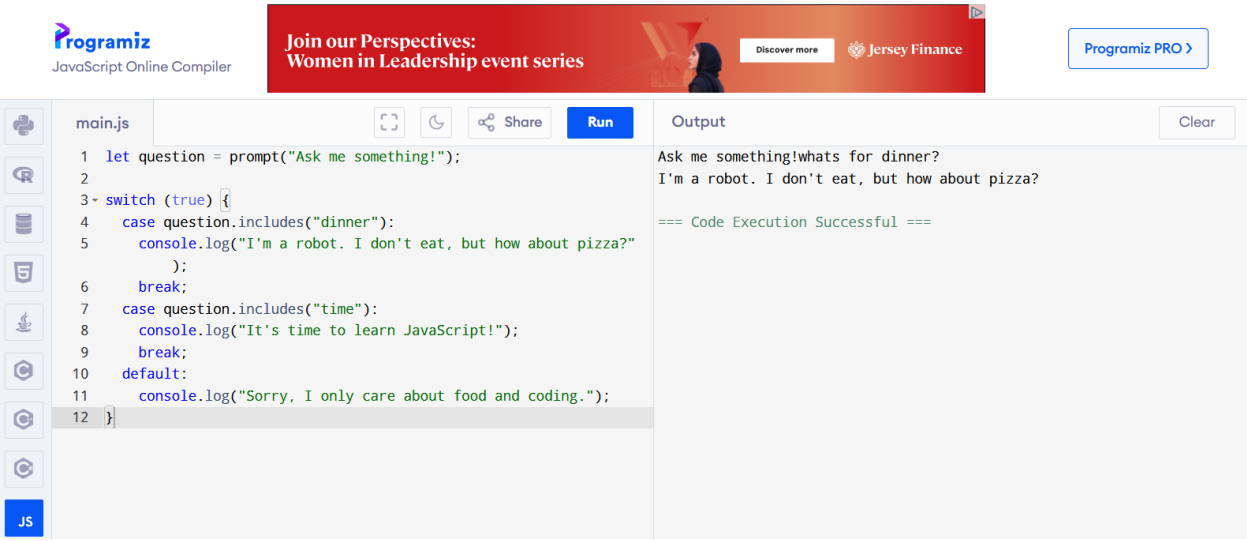
Code:

```
let question = prompt("Ask me something!");


switch (true) {
  case question.includes("dinner"):
    console.log("I'm a robot. I don't eat, but how about pizza?");
    break;
  case question.includes("time"):
```

```
    console.log("It's time to learn JavaScript!");
    break;
  default:
    console.log("Sorry, I only care about food and coding.");
}
```

Screenshot:



2e. Superheroes!

- Create a JavaScript object called superhero with properties name, power, and city.
- Add a method introduce that logs a message introducing the superhero.
- Log the superhero's introduction to the console.

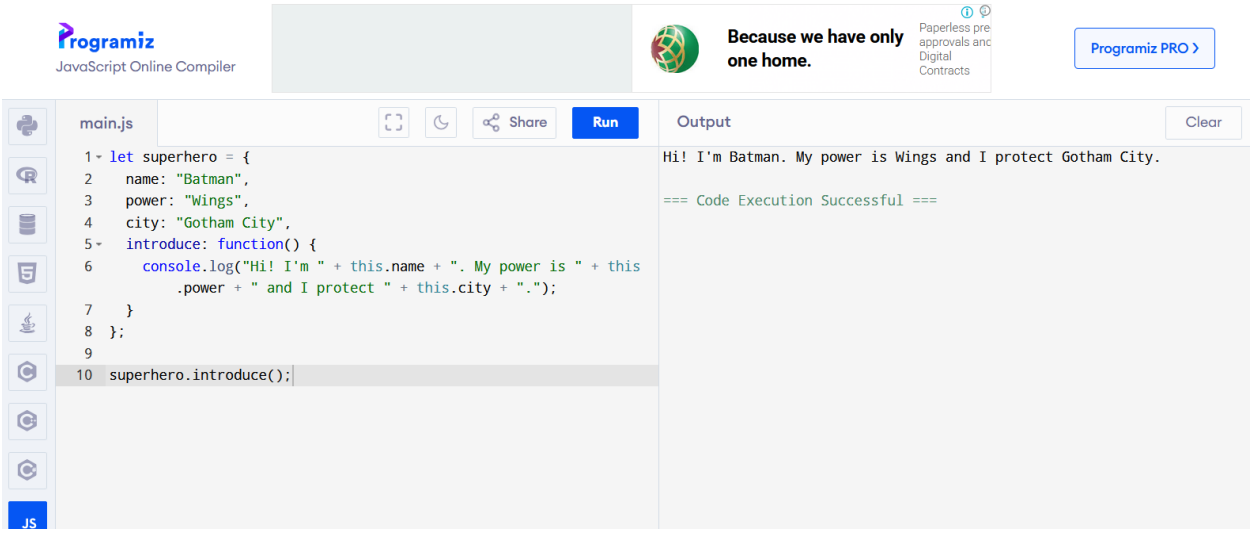
Code:

```
let superhero = {
  name: "Batman",
  power: "Wings",
  city: "Gotham City",
```

```
  introduce: function() {
    console.log("Hi! I'm " + this.name + ". My power is " + this.power + " and I protect " +
this.city + ".");
  }
};

superhero.introduce();
```

screenshot:



2f. Guess the number

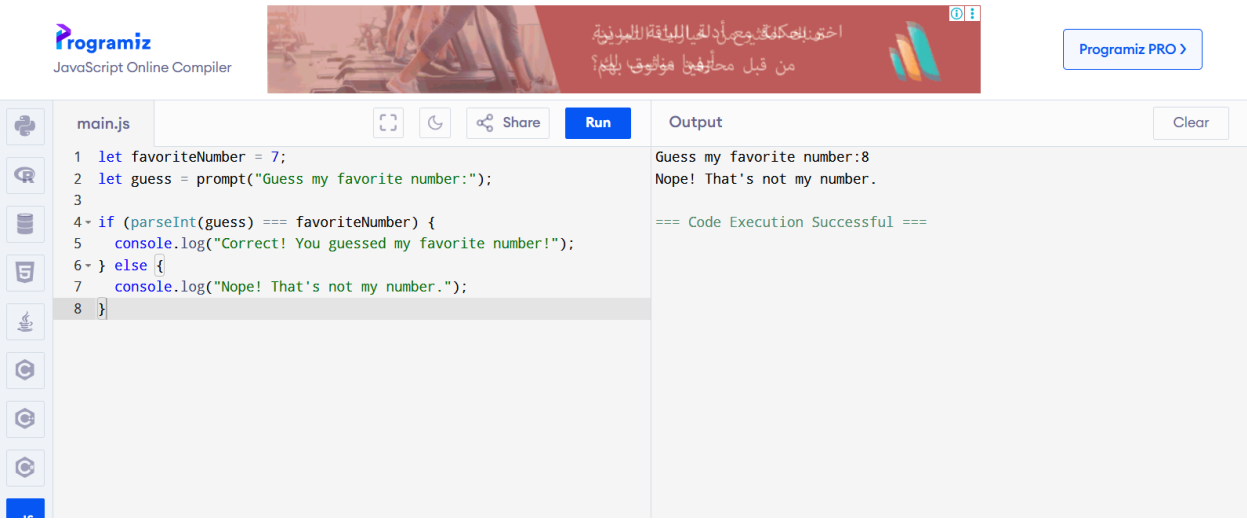
- Prompt the user to guess your favorite number.
- Use a strict equality (===) check to compare their guess with the correct number.
- Log a different message based on whether they guessed correctly or not.

Code:

```
let favoriteNumber = 7;
let guess = prompt("Guess my favorite number:");

if (parseInt(guess) === favoriteNumber) {
  console.log("Correct! You guessed my favorite number!");
} else {
  console.log("Nope! That's not my number.");
}
```

Screenshot:



Submission:

Upload the following to iLearn:

- A word document containing screenshots from exercise 1 and code and screenshots of all problems in exercise 2.