Full Stack Development Tools Overview

- **Subtitle:** Postman | Eclipse | Java | HTML | CSS | GitHub | MySQL
  Presented by: SANDHYA C
  Institute: CEC
  Date: 08-10-2025

Agenda

- Introduction
- Postman
- Eclipse
- Java
- HTML
- CSS
- GitHub
- MySQL
- Project Explanation
- Summary
- Q&A

Introduction

Overview of Full-Stack Development

- Full-stack development involves building both frontend (user interface) and backend (server, database, and logic) parts of a web application.
- A full-stack developer handles everything — from designing how a website looks to how it stores and processes data.

Why Each Tool is Essential

Frontend Tools (HTML, CSS, JavaScript, React, etc.) → Create responsive, user-friendly interfaces.

- Backend Tools (Node.js, Express.js, etc.) → Handle logic, authentication, and data flow.
- Database (MongoDB, MySQL, etc.) → Store and retrieve data efficiently.
- Version Control (Git, GitHub) → Manage code and collaborate easily

Tech Stack Covered in This Session

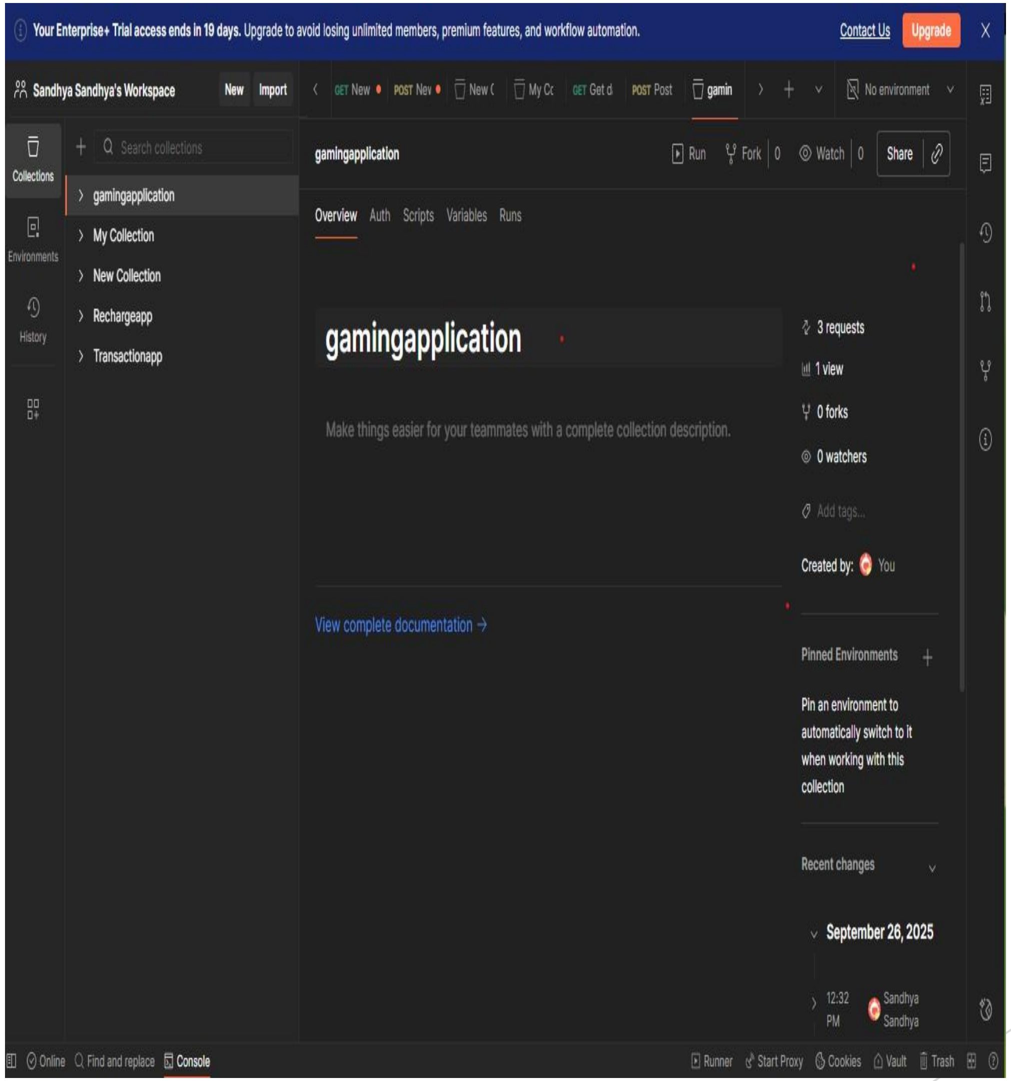
- Frontend: HTML, CSS, JavaScript, React
- Backend: Node.js, Express.js
- Database: MongoDB
- Tools: Git, GitHub, VS Code

Postman

Purpose: API testing and debugging

Key Features:

- Create and test REST API– Send requests (GET, POST, PUT, DELETE) and view responses.
- Automate test cases– Run pre-defined tests to check API performance and correctness.
- Use collections and environments– Organize APIs and manage variables for different setups (dev, test, prod)
- **Example:** Testing login API endpoint before UI integration → Send a POST request with username and password to check if login works properly before connecting it to the UI.



Your Enterprise+ Trial access ends in 19 days. Upgrade to avoid losing unlimited members, premium features, and workflow automation.

Make things easier for your teammates with a complete collection description.

Pin an environment to automatically switch to it when working with this collection

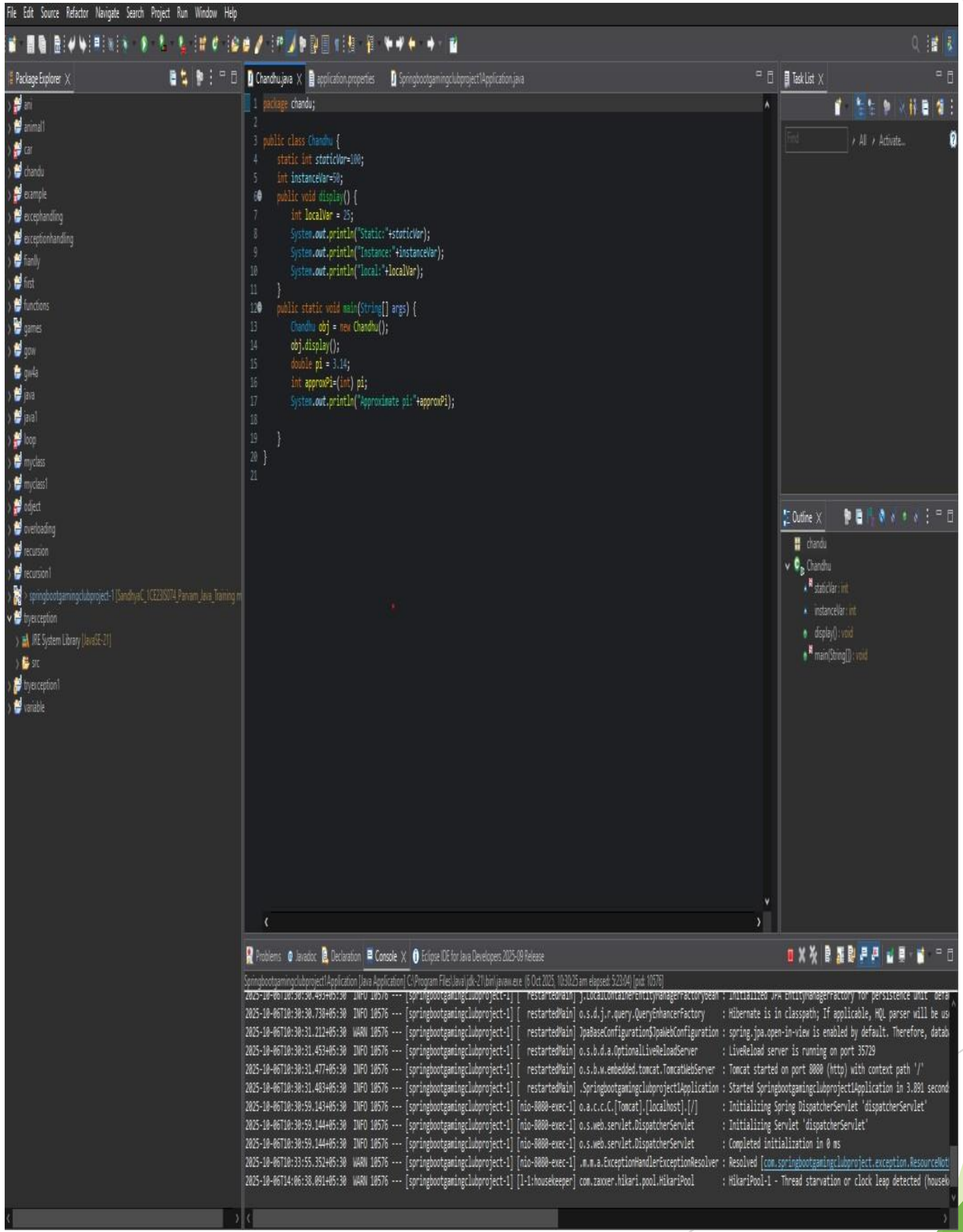
Eclipse IDE

Purpose

- A powerful development platform for Java and other programming languages.
- Commonly used for backend API development in enterprise applications.

Key Features

- Code Editing, Debugging & Build Automation –Write, run, and debug Java projects efficiently.
- Supports Maven and Git Plugins – Manage dependencies and integrate version control seamlessly.
- Suitable for Backend API Development – Ideal for creating and testing server-side logic and RESTful services.



```java
package chandu;

public class Chandhu {
    static int staticVar=100;
    int instanceVar=50;
    public void display() {
        int localVar = 25;
        System.out.println("Static:"+staticVar);
        System.out.println("Instance:"+instanceVar);
        System.out.println("local:"+localVar);
    }
    public static void main(String[] args) {
        Chandhu obj = new Chandhu();
        obj.display();
        double pi = 3.14;
        int approxPi=(int) pi;
        System.out.println("Approximate pi:"+approxPi);

    }
}
```

```
Springbootgamingclubproject1Application [Java Application] C:\Program Files\Java\jdk-21\bin\javaw.exe (6 Oct 2025, 10:30:25 am elapsed: 5:23:04) [pid: 10576]
2025-10-06T10:30:30.493+05:30  INFO 10576 --- [springbootgamingclubproject-1] [  restartedMain] j.LocalContainerEntityManagerFactoryBean : Initialized JPA EntityManagerFactory for persistence unit 'defa
2025-10-06T10:30:30.738+05:30  INFO 10576 --- [springbootgamingclubproject-1] [  restartedMain] o.s.d.j.r.query.QueryEnhancerFactory    : Hibernate is in classpath; If applicable, HQL parser will be us
2025-10-06T10:30:31.212+05:30  WARN 10576 --- [springbootgamingclubproject-1] [  restartedMain] JpaBaseConfiguration$JpaWebConfiguration : spring.jpa.open-in-view is enabled by default. Therefore, datab
2025-10-06T10:30:31.453+05:30  INFO 10576 --- [springbootgamingclubproject-1] [  restartedMain] o.s.b.d.a.OptionalLiveReloadServer       : LiveReload server is running on port 35729
2025-10-06T10:30:31.477+05:30  INFO 10576 --- [springbootgamingclubproject-1] [  restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat started on port 8080 (http) with context path '/'
2025-10-06T10:30:31.483+05:30  INFO 10576 --- [springbootgamingclubproject-1] [  restartedMain] .Springbootgamingclubproject1Application : Started Springbootgamingclubproject1Application in 3.891 second
2025-10-06T10:30:59.143+05:30  INFO 10576 --- [springbootgamingclubproject-1] [nio-8080-exec-1] o.a.c.c.C.[Tomcat].[localhost].[/]       : Initializing Spring DispatcherServlet 'dispatcherServlet'
2025-10-06T10:30:59.144+05:30  INFO 10576 --- [springbootgamingclubproject-1] [nio-8080-exec-1] o.s.web.servlet.DispatcherServlet        : Initializing Servlet 'dispatcherServlet'
2025-10-06T10:30:59.144+05:30  INFO 10576 --- [springbootgamingclubproject-1] [nio-8080-exec-1] o.s.web.servlet.DispatcherServlet        : Completed initialization in 0 ms
2025-10-06T10:33:55.352+05:30  WARN 10576 --- [springbootgamingclubproject-1] [nio-8080-exec-1] .m.m.a.ExceptionHandlerExceptionResolver : Resolved [com.springbootgamingclubproject.exception.ResourceNot
2025-10-06T14:06:38.091+05:30  WARN 10576 --- [springbootgamingclubproject-1] [l-1:housekeeper] com.zaxxer.hikari.pool.HikariPool        : HikariPool-1 - Thread starvation or clock leap detected (housek
```

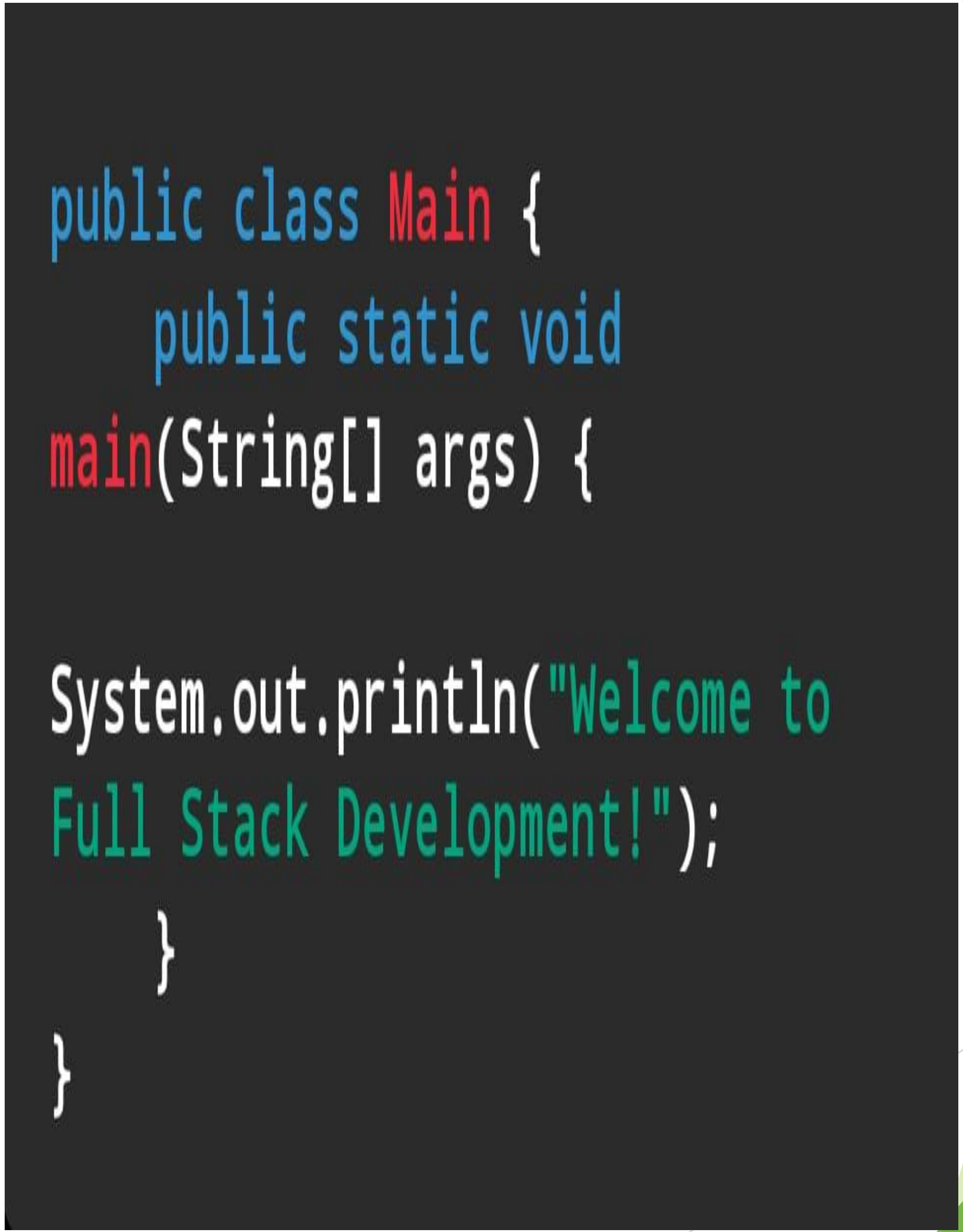
JAVA

Purpose:

- Core backend programming language

Highlights:

- Object-Oriented (follows class and object concepts)
- Platform Independent (runs on any OS via JVM)
- Rich Library Support (extensive APIs for networking, I/O, etc.)
- Secure and Robust (automatic memory management)
- Multi-threaded (supports concurrent programming)
- Widely Used for Enterprise Applications



```
public class Main {
    public static void
main(String[] args) {

System.out.println("Welcome to
Full Stack Development!");
    }
}
```

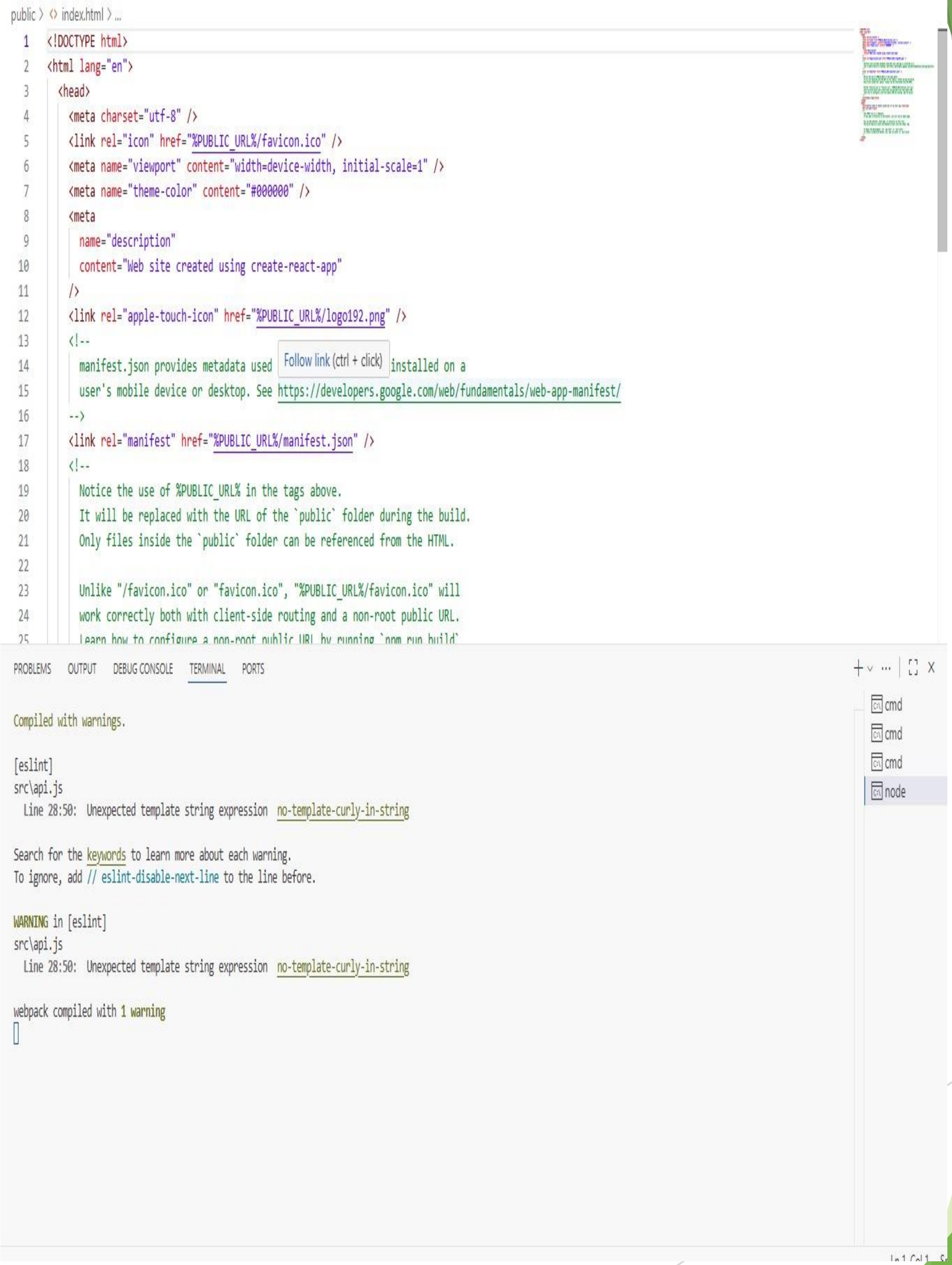
HTML

Purpose:

- ❖ Provides the structure and layout of web pages

Highlights:

- ❖ Stands for Hyper Text Mark up Language
- ❖ Defines headings, paragraphs, links, forms, and media
- ❖ Forms the skeleton of every website
- ❖ Works together with CSS (for design) and JS (for behavior)
- ❖ Uses tags and attributes to format content



```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <link rel="icon" href="%PUBLIC_URL%/favicon.ico" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <meta name="theme-color" content="#000000" />
    <meta
      name="description"
      content="Web site created using create-react-app"
    />
    <link rel="apple-touch-icon" href="%PUBLIC_URL%/logo192.png" />
    <!--
      manifest.json provides metadata used                            installed on a
      user's mobile device or desktop. See https://developers.google.com/web/fundamentals/web-app-manifest/
    -->
    <link rel="manifest" href="%PUBLIC_URL%/manifest.json" />
    <!--
      Notice the use of %PUBLIC_URL% in the tags above.
      It will be replaced with the URL of the `public` folder during the build.
      Only files inside the `public` folder can be referenced from the HTML.

      Unlike "/favicon.ico" or "favicon.ico", "%PUBLIC_URL%/favicon.ico" will
      work correctly both with client-side routing and a non-root public URL.
      Learn how to configure a non-root public URL by running `npm run build`
```

```
Compiled with warnings.

[eslint]
src\api.js
  Line 28:50:  Unexpected template string expression  no-template-curly-in-string

Search for the keywords to learn more about each warning.
To ignore, add // eslint-disable-next-line to the line before.

WARNING in [eslint]
src\api.js
  Line 28:50:  Unexpected template string expression  no-template-curly-in-string

webpack compiled with 1 warning
```

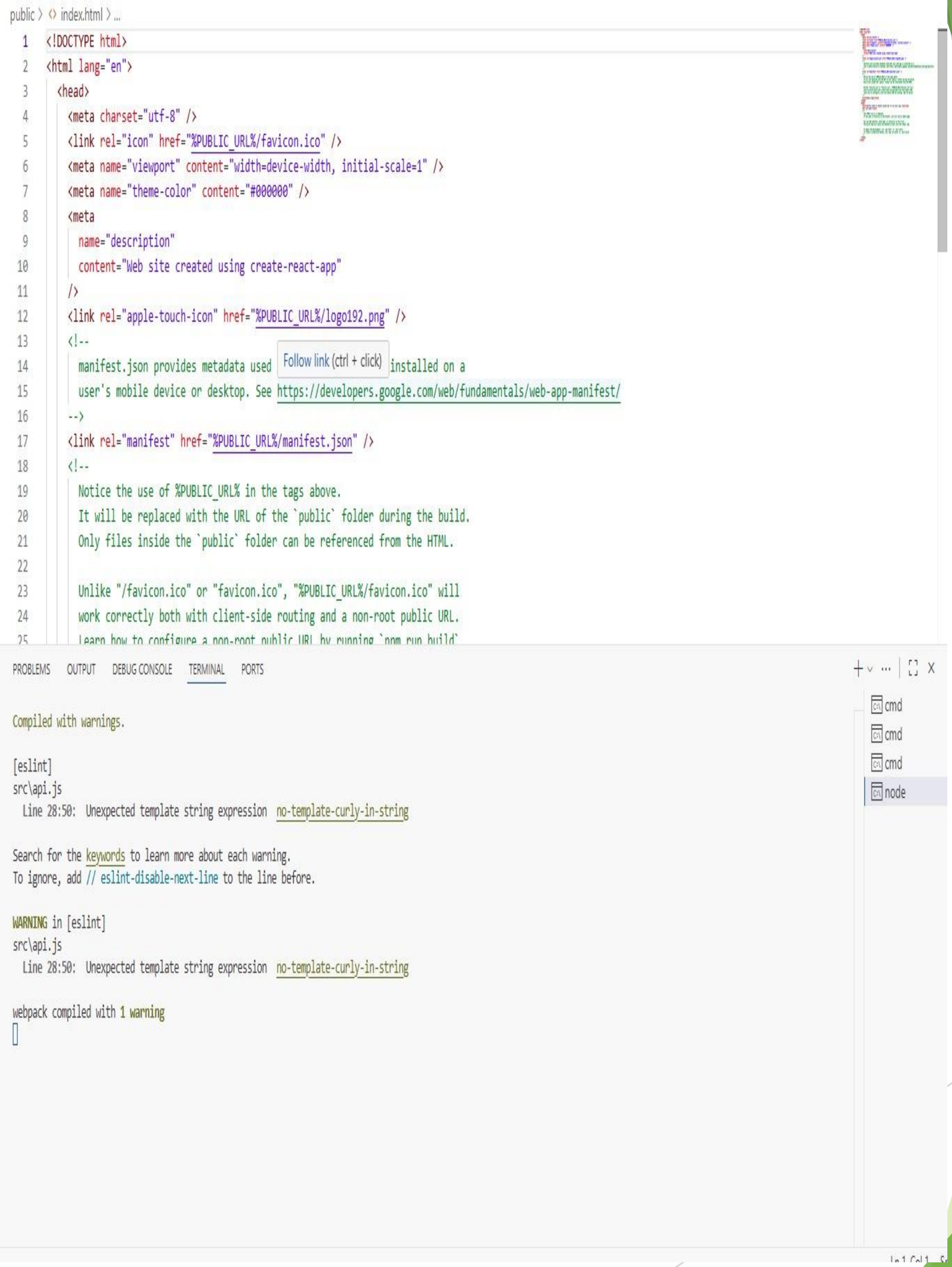
CSS

Purpose:

- Adds styling and design to HTML pages

Highlights:

- Stands for Cascading Style Sheets
- Controls layout, colors, fonts, and spacing
- Separates content (HTML) from design (CSS)
- Supports responsive design for all screen sizes
- Enhances user interface and overall experience (UI/UX)



```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <link rel="icon" href="%PUBLIC_URL%/favicon.ico" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <meta name="theme-color" content="#000000" />
    <meta
      name="description"
      content="Web site created using create-react-app"
    />
    <link rel="apple-touch-icon" href="%PUBLIC_URL%/logo192.png" />
    <!--
      manifest.json provides metadata used                            installed on a
      user's mobile device or desktop. See https://developers.google.com/web/fundamentals/web-app-manifest/
    -->
    <link rel="manifest" href="%PUBLIC_URL%/manifest.json" />
    <!--
      Notice the use of %PUBLIC_URL% in the tags above.
      It will be replaced with the URL of the `public` folder during the build.
      Only files inside the `public` folder can be referenced from the HTML.

      Unlike "/favicon.ico" or "favicon.ico", "%PUBLIC_URL%/favicon.ico" will
      work correctly both with client-side routing and a non-root public URL.
      Learn how to configure a non-root public URL by running `npm run build`.
```

```
Compiled with warnings.

[eslint]
src\api.js
  Line 28:50:  Unexpected template string expression  no-template-curly-in-string

Search for the keywords to learn more about each warning.
To ignore, add // eslint-disable-next-line to the line before.

WARNING in [eslint]
src\api.js
  Line 28:50:  Unexpected template string expression  no-template-curly-in-string

webpack compiled with 1 warning
```

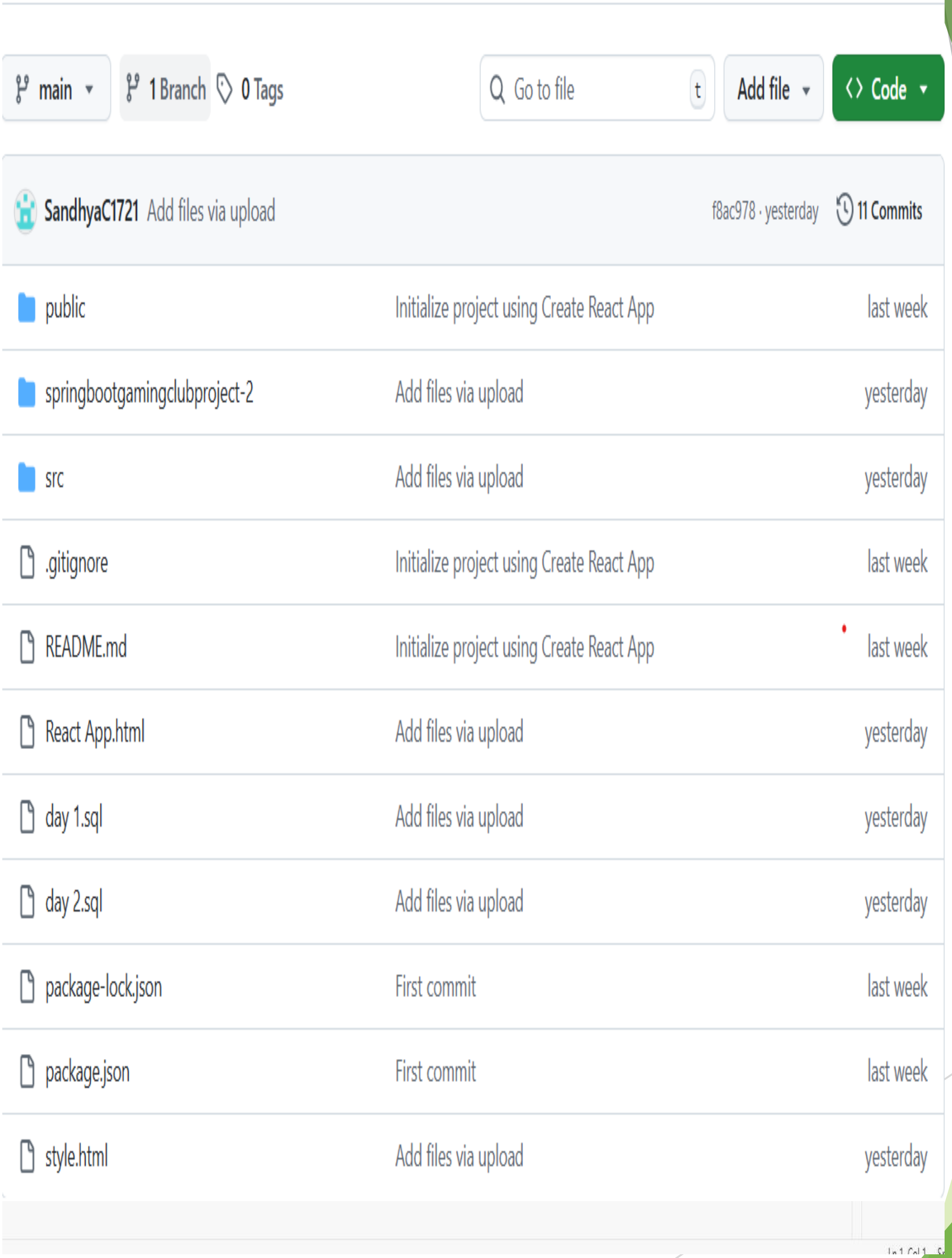
GITHUB

Purpose:
Version control and team collaboration platform

Features:

- Manage and track source code versions
- Supports branching and merging for team work
- Integrates with CI/CD pipelines for automation
- Enables pull requests and code reviews
- Provides GitHub Pages for hosting projects
- Cloud-based and accessible anywhere

Example: Store and share code projects Collaborate with developers through commits and branches



SandhyaC1721 Add files via upload f8ac978 · yesterday 11 Commits

public	Initialize project using Create React App	last week
springbootgamingclubproject-2	Add files via upload	yesterday
src	Add files via upload	yesterday
.gitignore	Initialize project using Create React App	last week
README.md	Initialize project using Create React App	last week
React App.html	Add files via upload	yesterday
day 1.sql	Add files via upload	yesterday
day 2.sql	Add files via upload	yesterday
package-lock.json	First commit	last week
package.json	First commit	last week
style.html	Add files via upload	yesterday

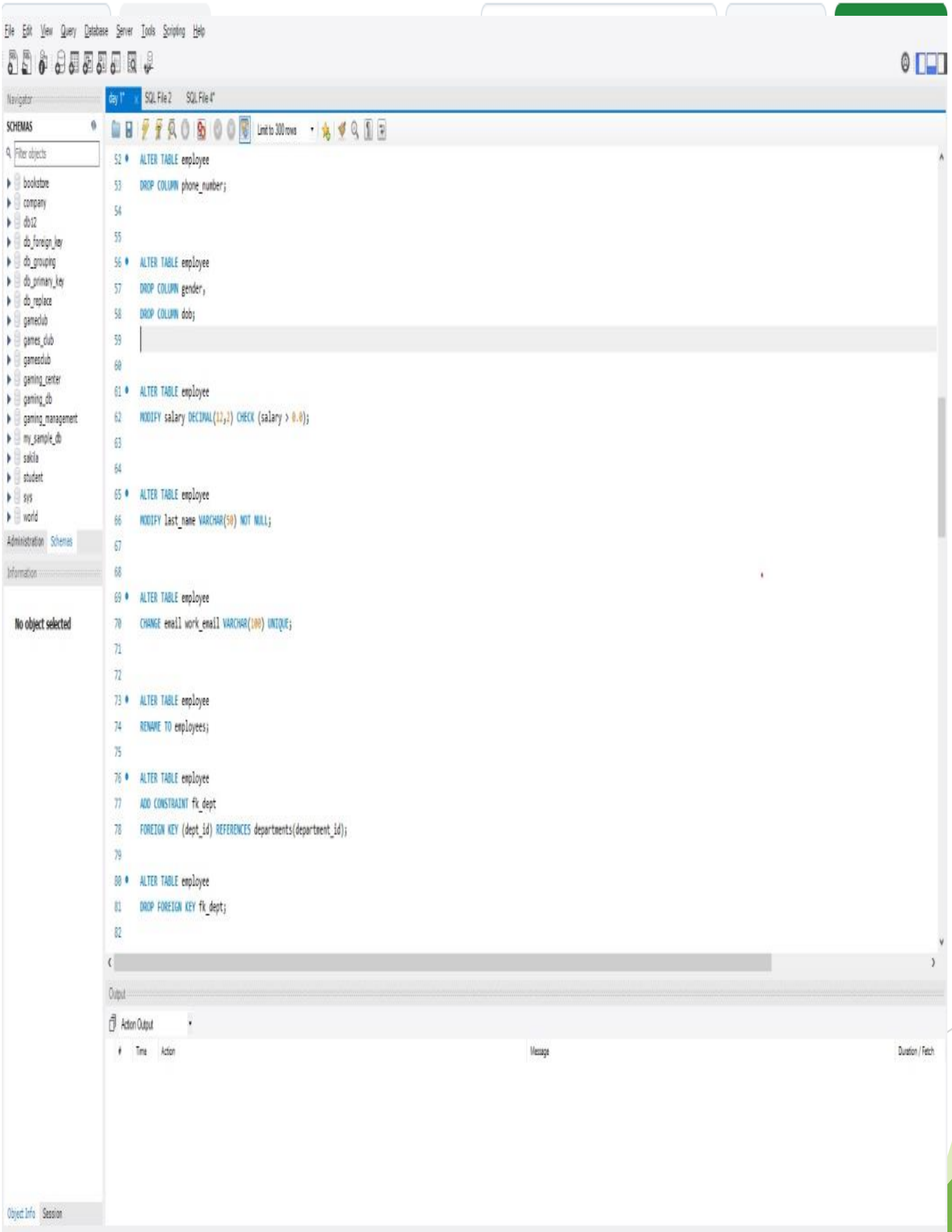
MYSQL

Purpose:

- Backend database used for storing and managing application data

Highlights:

- Open-source Relational Database Management System (RDBMS)
- Uses SQL (Structured Query Language) for data operations
- Supports tables, relationships, and constraints
- Ensures data integrity and security
- Widely used with backend technologies like Java, PHP, and Python



```sql
ALTER TABLE employee
DROP COLUMN phone_number;



ALTER TABLE employee
DROP COLUMN gender,
DROP COLUMN dob;



ALTER TABLE employee
MODIFY salary DECIMAL(12,2) CHECK (salary > 0.0);



ALTER TABLE employee
MODIFY last_name VARCHAR(50) NOT NULL;



ALTER TABLE employee
CHANGE email work_email VARCHAR(100) UNIQUE;



ALTER TABLE employee
RENAME TO employees;

ALTER TABLE employee
ADD CONSTRAINT fk_dept
FOREIGN KEY (dept_id) REFERENCES departments(department_id);

ALTER TABLE employee
DROP FOREIGN KEY fk_dept;

```

Slide 11: Project Explanation

Name: Employee Management System

Overview:

A web-based system to manage employee details, allowing CRUD operations through a Java backend and MySQL database

.**Tools Used and its Purpose:**

HTML/CSS
Frontend UI
Java (Eclipse)
Backend logic
MySQL
Data storage
Postman
API testing
GitHub
Version control
Eclipse IDE
Code development

Workflow:

Frontend (HTML/CSS) → Sends data to backend via form submission .**Backend (Java)** → Processes data and interacts with My SQL. **Database (MySQL)** → Stores and retrieves employee information .**Postman** → Used for testing backend .**APIs GitHub** → Used for source code management. **Outcome:** ✓ Complete CRUD-based application ✓ Integrated testing using Postman ✓ Database connectivity with MySQL ✓ Hosted and managed via GitHub

SUMMARY

Tool
Role
Postman
API Testing
Eclipse
IDE for Development
Java
Backend Logic
HTML
Page Structure
CSS
Page Styling
GitHub
Version Control
MySQL
Database

Q&A

QUESTONS AND DISCUSSION

CONTACT:[ssandhyac747@gmail.com]