**Session 2: Practical**

**1. Setting Up Development Environments**

* Install Python, Django, Node.js, React, and MySQL.
* Configure Git and create a new repository for the project.

**2. Git Basics & Collaboration**

* Initialize a Git repository and push an initial commit.
* Work with branches: create, switch, and merge.
* Practice pull requests and code reviews.

**3. Project Setup**

* Django group: Create a new Django project and run a basic server.
* Node.js group: Set up an Express.js project and create an API route.
* React: Initialize a React project using Create React App.

**4. Database Connection**

* Set up a MySQL database and connect it to the backend.
* Run initial migrations in Django or Sequelize in Node.js.

**5. First Application Commit**

* Push changes to GitHub.
* Write a README file explaining the project setup.

**6. Debugging and Troubleshooting**

* Basic debugging techniques in Django and Node.js.
* Using browser dev tools to inspect API responses in React.
* Checking MySQL database connections and resolving common setup issues.

**7. Code Linting and Formatting**

* Install ESLint and Prettier for consistent JavaScript code formatting.
* Use Black for Python code formatting in Django.
* Configure linting tools in VS Code for real-time feedback.

**8. Environment Variables and Configuration Management**

* Set up .env files to store secret keys and database credentials securely.
* Use dotenv for Node.js and django-environ for Django to manage configurations.

**9. Basic API Testing**

* Use Postman or Thunder Client to test API endpoints.
* Verify JSON responses from Django REST Framework or Express.js APIs.

**10. Git Best Practices for Team Collaboration**

* Enforce commit message guidelines (feat: add login functionality).
* Introduce Git hooks for pre-commit linting and formatting.
* Demonstrate rebasing and interactive commits for clean commit history.

**11. Deployment Preparation**

* Brief overview of hosting options for React (Netlify, Vercel).
* Brief overview of backend deployment options (Heroku, AWS, DigitalOcean).
* Explain the importance of .gitignore to exclude unnecessary files from version control.