Demo : Generate Git commit messges using GitHub Copilot

In this demo, we will explore how to use GitHub Copilot to generate meaningful and efficient Git commit messages for your projects. This can help streamline your development workflow and ensure that your commit history is clear and informative.

Prerequisites

- Visual Studio Code installed
- GitHub Copilot extension enabled
- GitHub Copilot subscription
- NodeJS 18 or later
- BookAPI project

Steps to Commit the BookAPI Project to Git

1. Open `book-api` project in VS Code. Open Copilot edit and run the following prompt to add a `.gitignore` file.

```
Generate a .gitignore file for the javascript project
```

2. Initialize Git Repository

```
git init
```

3. Stage All Files for Initial Commit

```
git add .
```

4. Make the Initial Commit
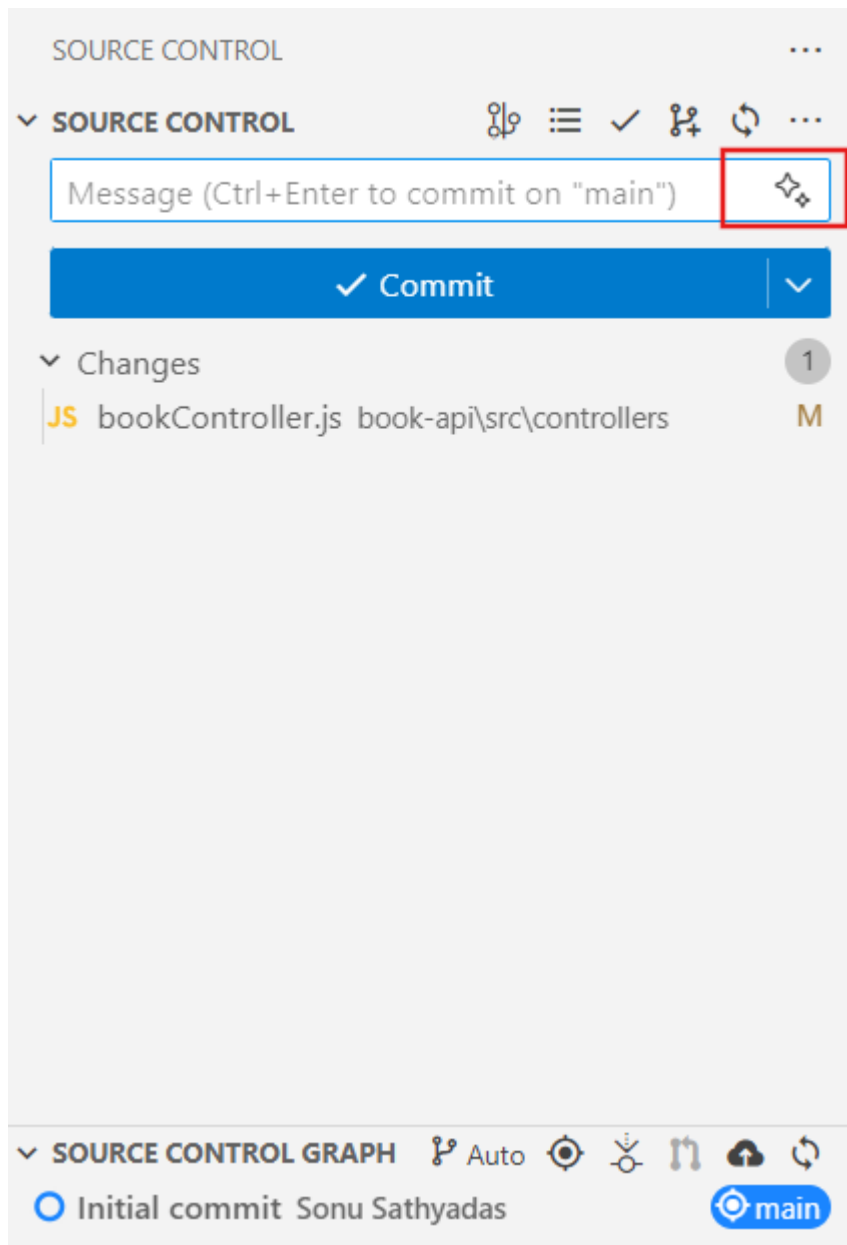
```
git commit -m "Initial commit"
```

5. Open `bookController.js` and add a new method or modify an existing one. Save the changes. You can add the following method

```
async getPaginatedBooks(req, res) {
    try {
```

```js
        const { page = 1, limit = 10 } = req.query;
        const offset = (page - 1) * limit;

        this.bookModel.getPaginatedBooks(offset, limit, (err, books) => {
            if (err) {
                res.status(500).json({ message: 'Error retrieving books',
  error: err });
            } else {
                res.status(200).json(books);
            }
        });
    } catch (error) {
        res.status(500).json({ message: 'Error retrieving books', error });
    }
}
```

6. Open the source control in VS Code and click on the `Generate commit message with Copilot` icon inside the textbox.

7. It will generate a commit message. Click on the commit button to commit the changes to local machine