



Assignment: Build a Full Todo App with FastAPI + Templates

Project Overview

You've been given a starter FastAPI project that includes:

- A FastAPI setup
- Templating integrated using Jinja2
- Endpoints for creating todos and to view individual and all todos
- Static files (CSS) already connected
- A PostgreSQL database model using SQLModel
- Check the `main.py` for the FastAPI routes and `templates/` for Jinja2 files.

Your Task

Extend the application by implementing full **Update** and **Delete** functionality for the Todo items using both backend logic and HTML templates.

What You Need to Do

1. **Unzip the `todo_app.zip` file**
2. **Create a virtual environment in the project directory (`todo_app`)**

```
python -m venv .venv
```

3. Upgrade pip:

```
python -m pip install --upgrade pip
```

4. **Install the project dependencies from the `requirements.txt` file**

```
pip install -r requirements.txt
```

5. Run the project:

```
fastapi dev main.py
```

Then open your browser and go to <http://127.0.0.1:8000> to access the app.

6. Update a Todo

- Add a route and form to update an existing todo.

7. Delete a Todo

- Add a route and button to delete a todo item.

Requirements

- All pages must be rendered using **Jinja2 templates**
- Use **HTML and CSS** for styling (static files are already configured)
- Display proper feedback to the user after each action (e.g., "Todo updated successfully")

Resources

You'll find the starter project attached:

 [todo_app.zip](#)

Deliverables

- A complete and working FastAPI app
- All four CRUD operations implemented (users should be able to Create, Read, Update, and Delete todos)
- Clean and readable HTML templates
- **⚠ Before uploading your project as a `.zip` file, make sure to remove the `.venv` folder.**

