Great — since everything works now, here’s a precise summary of **what caused the issues**:

### ✅ Issue 1: 404 Not Found on /api/todos

**Cause:**  
Your frontend JavaScript used:

const apiUrl = "/api/todos";

But your Spring app was deployed under context /todo, so the correct path was:

/todo/api/todos

**Fix:**  
Set:

const apiUrl = "/todo/api/todos";

### ✅ Issue 2: 406 Not Acceptable or 500 No converter

**Cause:**

Missing Jackson dependency

Or spring-servlet.xml didn’t initially include:

<mvc:annotation-driven/>

**Fix:**

Add Jackson to pom.xml

Add <mvc:annotation-driven/> in Spring config

### ✅ Issue 3: Add button not working

**Cause:**

Outdated todo.js was cached in the browser

Even after code fix, browser used stale file

**Fix:**

Added cache-busting in index.html:

<script src="scripts/todo.js?v=2"></script>

Used Ctrl+Shift+R to hard reload

### ✅ Summary:

| **Problem** | **Root Cause** | **Fix** |
| --- | --- | --- |
| 404 on /api/todos | Wrong JS path; app context was /todo | Changed apiUrl to /todo/api/todos |
| 406/500 errors | Spring missing JSON converter | Added Jackson + mvc:annotation-driven |
| Add button didn’t respond | Cached old JS file | Cache-busting + hard reload |

Now your app is correctly serving and using REST endpoints via Spring MVC and fetch. Let me know if you want to deploy this as a WAR to external Tomcat or host it.

### ✅ Summary:

In IntelliJ Ultimate, a **Data Source** is a visual database connection tool

It helps you inspect and query your DB

You still need a **Spring DataSource bean** in your app to use the DB at runtime

Let me know if you want to sync IntelliJ's Data Source with your Spring config.