Budget Tracker Project 📊 💰

This is a **Budget Tracker** web application built using **Java**, **Servlets**, **JSP**, **MySQL**, and **Tomcat**. It allows users to add, edit, and delete their budget entries. This project is designed to track expenses with simple CRUD operations and a user-friendly interface.

Project Structure 📁

```
BudgetTracker/
│
├── WEB-INF/
│   ├── classes/                 # Compiled Java classes (place all .class files here)
│   ├── lib/                     # External libraries (e.g., MySQL Connector, other JAR files)
│   └── web.xml                  # Web application deployment descriptor
│
├── css/                         # CSS files for styling
│   └── style.css                # Main CSS file
│
├── images/                      # Image assets (if any)
│
├── jsp/                         # JSP files for the front end
│   ├── index.jsp                # Home page for displaying and adding budgets
│   └── edit.jsp                 # Page for editing an existing budget
│
├── src/                         # Source code for servlets and classes
│   ├── com/                     # Package directory for servlets and classes
│   │   └── budget/              # Package for budget-related servlets
│   │       ├── AddBudgetServlet.java     # Servlet for adding new budget
│   │       ├── EditBudgetServlet.java    # Servlet for editing an existing budget
│   │       ├── DeleteBudgetServlet.java  # Servlet for deleting a budget
│   │       └── DBConnection.java         # Database connection utility class
│
├── .gitignore                   # Git ignore file (to exclude unnecessary files)
```

```
└── README.md                    # Project documentation file
```

Features 🚀

- **Add Budget:** Users can add new budget entries with details (e.g., category, amount, date).
- **Edit Budget:** Users can edit existing budget entries to update their information.
- **Delete Budget:** Users can remove budget entries from the tracker.
- **Display Budgets:** All the budgets are displayed in a clear and organized manner on the home page.

Requirements 📦

- **Java JDK 8 or higher** ☕
- **Apache Tomcat 9.0 or higher** 🌐
- **MySQL Database** 🛠️
- **MySQL Connector/J** for connecting the Java application with MySQL (included in `lib/` folder).

Setup Guide 🛠️

Follow these steps to set up and run the project:

- **Step 1: Install Java and Set `JAVA_HOME`** 📍

    - Download and install Java from the official [Oracle website](#).
    - Set the `JAVA_HOME` environment variable on your system:
        - **Windows:** Go to System Properties -> Advanced -> Environment Variables -> New (System Variable) -> JAVA_HOME =
        - **Mac/Linux:** Add `export JAVA_HOME=` to your `~/.bash_profile` or `~/.bashrc`.
- **Step 2: Install Apache Tomcat 🌐**
    - Download and install Apache Tomcat from the official [Tomcat website](#).
    - Set the `CATALINA_HOME` environment variable similarly to `JAVA_HOME`.
- **Step 3: Set up MySQL 🛠️**
    - Install MySQL and create a database for the project (e.g., `budget_tracker`).
    - Make sure to update the database connection details in `DBConnection.java` (URL, username, password).
- **Step 4: Compile the Project 🧑‍💻**
    - Navigate to the project root directory and use the following command to compile all the Java files:

      ```
      javac -cp ".;C:\path\to\tomcat\lib\servlet-api.jar;mysql-connector-java.jar" -d WEB-INF/classes *.java
      ```
- **Step 5: Deploy to Tomcat 🚀**
    - Place the project in the `webapps` directory of Tomcat.
    - Start Tomcat with the command: `bin/startup.bat` (Windows) or `bin/startup.sh` (Mac/Linux).
    - To stop Tomcat: `bin/shutdown.bat` (Windows) or `bin/shutdown.sh` (Mac/Linux).
- **Step 6: Access the Project 🌍**
    - Open your browser and go to `http://localhost:8080/BudgetTracker/` to view and interact with the project.