**Setting Up Your Sales Data Analytics Project: A Student's Guide**

Let me walk you through this process in simple terms. Think of this like building with LEGOs - we need to get all our pieces (tools) first, then learn how to connect them properly to create our project.

**Step 1: Installing the Tools (Getting Your LEGO Pieces)**

**1. NetBeans IDE 8.0.2 (Your Workspace)**

**What it is:** NetBeans is like a digital workbench where you'll write and organize your code. It makes programming much easier by highlighting errors and helping you manage your project files.

**How to install:**

1. Go to the NetBeans website: <https://netbeans.org/downloads/8.0.2/>
2. Download the version for your computer (Windows, Mac, or Linux)
3. Run the installer - it's like installing any other program
4. Just click "Next" through all the options
5. When it's done, open NetBeans - you'll see a welcome screen

**2. Database (MySQL or PostgreSQL) (Your Digital Filing Cabinet)**

**What it is:** A database is like a super-organized filing cabinet where your sales data will be stored. It keeps everything neat and easy to find.

**How to install MySQL:**

1. Go to: <https://dev.mysql.com/downloads/installer/>
2. Download the MySQL Installer for your computer
3. Run the installer
4. When it asks for a password, create one and WRITE IT DOWN (you'll need it later)
5. Choose "Developer Default" setup
6. Finish the installation

**3. Java Development Kit (JDK) (Your Tools)**

**What it is:** The JDK is like the toolbox that contains all the tools needed to build Java programs.

**How to install:**

1. Go to: <https://www.oracle.com/java/technologies/javase/javase8-archive-downloads.html>
2. Download "JDK 8" for your computer
3. Run the installer
4. Follow the instructions (mostly just clicking "Next")

**4. Apache POI (For Excel Files) (Your Special Attachment)**

**What it is:** This is an add-on that lets your program work with Excel files.

**How to get it:**

1. Go to: <https://poi.apache.org/download.html>
2. Download the latest binary release (it will be a ZIP file)
3. Extract it to a folder you can remember (like "Downloads")

**Step 2: Setting Up Your Project (Organizing Your Workbench)**

**Creating Your Project in NetBeans**

1. Open NetBeans
2. Click "File" > "New Project"
3. In the Categories, select "Java"
4. In Projects, select "Java Application" and click "Next"
5. Name your project "SalesDataAnalytics"
6. Uncheck "Create Main Class" (we'll add our own later)
7. Click "Finish"

**Creating the Project Structure**

1. In the Projects panel (left side), right-click "Source Packages"
2. Select "New" > "Java Package"
3. Name it "models" and click "Finish"
4. Repeat this process to create these packages:
   * "dao"
   * "services"
   * "utils"
   * "ui"

**Adding Libraries (Connecting Special Tools)**

1. In the Projects panel, right-click "Libraries"
2. Select "Add JAR/Folder"
3. Find and add these files:
   * Your database JDBC driver (usually in your database installation folder)
   * All the JAR files from the Apache POI folder you extracted earlier

**Step 3: Adding the Source Code (Following the Instructions)**

**Understanding What You're Doing**

Think of the source code as the instruction manual for your project. Each file has a specific job:

* **models**: These are like templates for your data (what an inventory item looks like)
* **dao**: These are like librarians that know how to store and retrieve data
* **services**: These are like managers that handle the business logic
* **utils**: These are like helper tools for special tasks (like reading Excel files)
* **ui**: This is what the user sees and interacts with

**How to Add the Code**

1. For each file mentioned in the project structure:
   * Right-click the appropriate package (like "models")
   * Select "New" > "Java Class"
   * Name the file exactly as shown (like "InventoryItem")
   * Copy and paste the code into the file
   * Save the file
2. **Important:** In the **DatabaseUtil.java** file, you'll need to update these lines with your database information:

java

Line Wrapping

Collapse

Copy

1

2

3

private static final String URL = "jdbc:mysql://localhost:3306/sales\_analytics";

private static final String USER = "your\_username"; // Usually "root"

private static final String PASS = "your\_password"; // The password you created

**Step 4: Setting Up Your Database (Building Your Filing Cabinet)**

**Creating the Database**

1. Open MySQL (or your database tool)
2. Create a new database named "sales\_analytics"
3. Run these SQL commands to create your tables (copy and paste them):

sql

Line Wrapping

Collapse

Copy

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

⌄

⌄

⌄

CREATE TABLE inventory (

item\_id VARCHAR(20) PRIMARY KEY,

description VARCHAR(100) NOT NULL,

quantity INT NOT NULL,

supplier VARCHAR(50) NOT NULL

);

CREATE TABLE transactions (

transaction\_id VARCHAR(20) PRIMARY KEY,

item\_id VARCHAR(20) NOT NULL,

transaction\_date DATE NOT NULL,

quantity\_sold INT NOT NULL,

unit\_price DECIMAL(10,2) NOT NULL,

FOREIGN KEY (item\_id) REFERENCES inventory(item\_id)

);

CREATE TABLE users (

user\_id VARCHAR(20) PRIMARY KEY,

username VARCHAR(50) NOT NULL UNIQUE,

password VARCHAR(100) NOT NULL,

role VARCHAR(20) NOT NULL

);

**Step 5: Running Your Project (Bringing It to Life)**

**Testing Your Project**

1. Create a simple CSV file named "inventory.csv" with this content:

Line Wrapping

Collapse

Copy

1

2

3

ITM001,Apple Juice,50,SupplierA

ITM002,Bread,30,SupplierB

ITM003,Cereal,20,SupplierC

1. In NetBeans, find the "Main.java" file in the "ui" package
2. Right-click it and select "Run File"
3. Your application should open!

**Using Your Application**

1. **Import Data**: Go to "File" > "Import CSV" and select your inventory.csv file
2. **View Reports**: Try clicking the buttons to see different reports
3. **Export Data**: Try exporting to Excel using "File" > "Export to Excel"

**Step 6: Troubleshooting (When Things Don't Work)**

**Common Problems and Solutions**

1. **Database Connection Error**
   * Make sure your database is running
   * Check that your username and password in DatabaseUtil.java are correct
   * Verify the database name is exactly "sales\_analytics"
2. **Library Errors**
   * Make sure you added all the JAR files to your project
   * Check that you're using the right versions
3. **Code Errors**
   * Look for red underlines in the code - these indicate errors
   * Hover over the error to see what's wrong
   * Check that you copied the code exactly as provided

**Final Tips for Success**

1. **Take it step by step** - Don't try to do everything at once
2. **Save often** - Click the save icon frequently
3. **Test each part** - After adding a new file, try running the project to see if it still works
4. **Ask for help** - If you're stuck, ask your instructor or classmates
5. **Experiment** - Once it's working, try making small changes to see what happens

Remember, learning to program is like learning a new language or instrument - it takes practice and patience. You'll make mistakes, but that's how you learn! Good luck with your project!