# **🌟 Phase 8: Data Management & Deployment (Detailed)**

This phase focuses on **managing Salesforce data efficiently, maintaining data quality, and deploying changes safely** across environments.

**1️⃣ Data Import Wizard (Simple Data Upload)**

**Purpose:** Quickly import records into Salesforce without coding.

**Where to Go:**  
Setup → Data Import Wizard

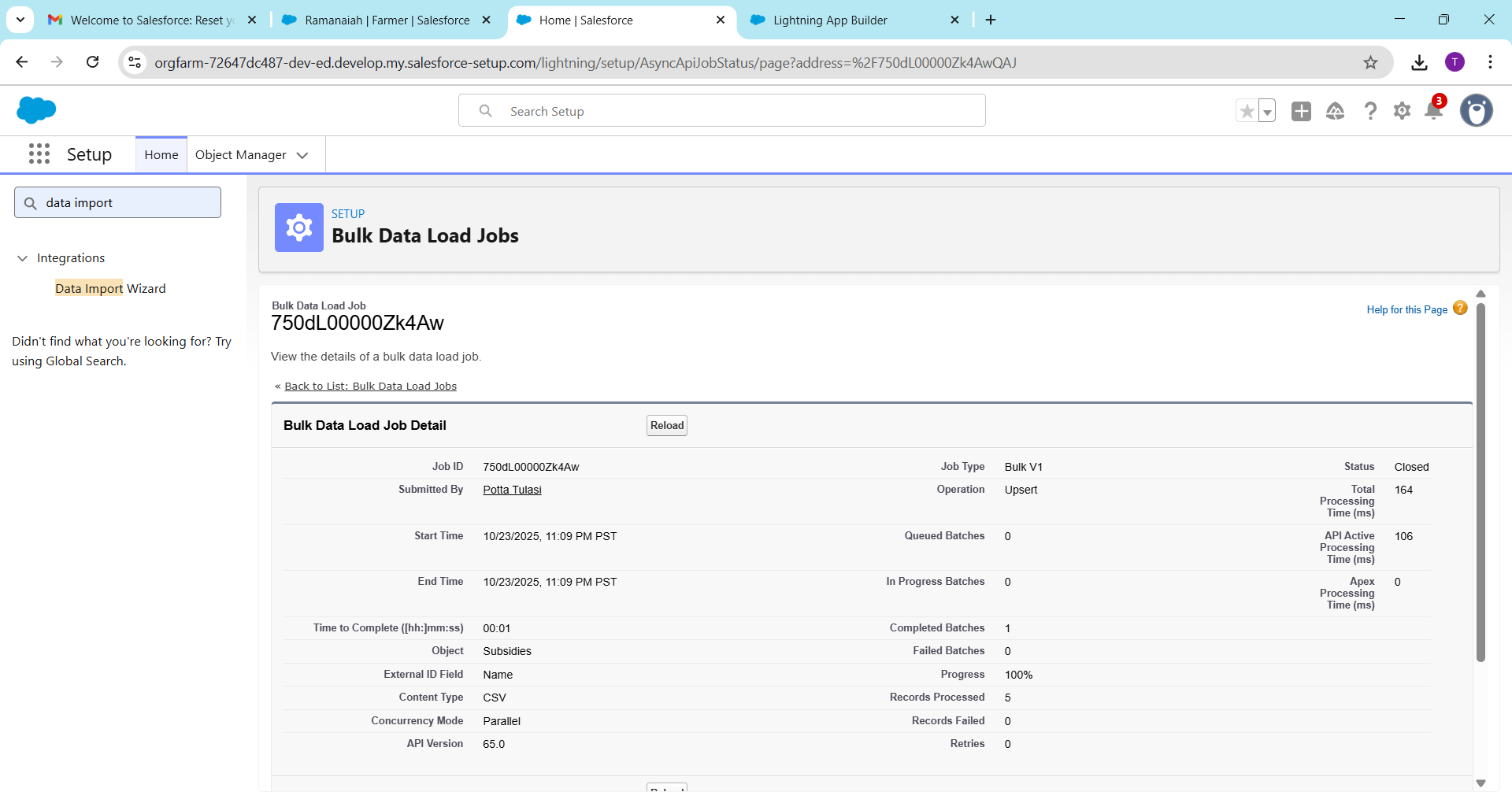
**Step-by-Step Instructions:**

1. Choose the **object** you want to import (e.g., Farmer\_\_c, Buyer\_\_c, Order\_\_c)
2. Upload a **CSV file** with your data
3. **Map CSV columns to Salesforce fields** (e.g., Farmer Name → Name, Phone → Phone)
4. Review errors or duplicates flagged by Salesforce
5. Click **Start Import**

**Best For:** Small to medium data sets (up to 50,000 records).

**Example in Rural Market Linkage:**

* Import **new farmer registrations** collected from offline forms into the Farmer\_\_c object.



**2️⃣ Data Loader (Bulk Data Management)**

**Purpose:** Bulk insert, update, delete, or export Salesforce records.

**Where to Go:**

* Download **Data Loader**: [Salesforce Data Loader Download](https://help.salesforce.com/s/articleView?id=sf.data_loader_install.htm)

**Step-by-Step Instructions:**

1. Open **Data Loader** → Login with your Salesforce credentials (Sandbox or Production)
2. Select operation:
   * **Insert:** Add new records
   * **Update:** Modify existing records
   * **Upsert:** Insert new records or update existing ones
   * **Delete:** Remove records
   * **Export:** Extract data from Salesforce
3. Upload your **CSV file** and **map fields**
4. Run the operation → check **success and error files**

**Best For:** Large datasets (50,000+ records), complex updates.

**Example:** Bulk update the **Crop\_\_c prices** for multiple farmers after a new government subsidy adjustment.

**3️⃣ Duplicate Rules (Prevent Duplicate Data)**

**Purpose:** Maintain data integrity by **blocking duplicate records**.

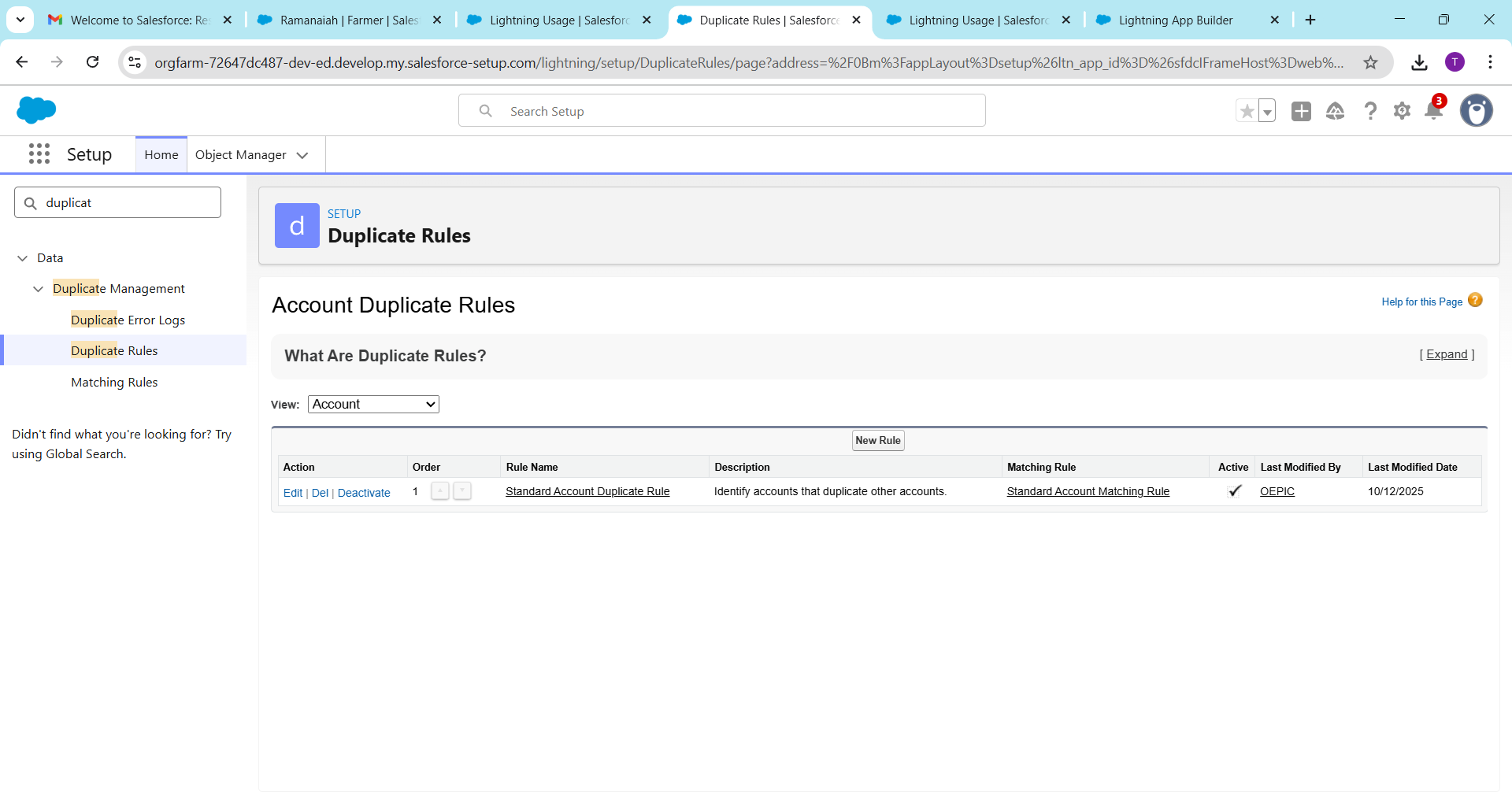
**Where to Go:**  
Setup → Duplicate Management → Duplicate Rules

**Step-by-Step Instructions:**

1. Create **Matching Rules** (e.g., match farmers by Email or Phone)
2. Create **Duplicate Rule** for the object:
   * **Block:** Prevent duplicates
   * **Alert:** Notify users but allow save
3. Activate the rule

**Example:** Prevent multiple Farmer\_\_c records with the same phone number.

**Tip:** Combine **Duplicate Rules + Matching Rules** for maximum accuracy.



**4️⃣ Data Export & Backup (Secure Your Data)**

**Purpose:** Regular backups prevent data loss.

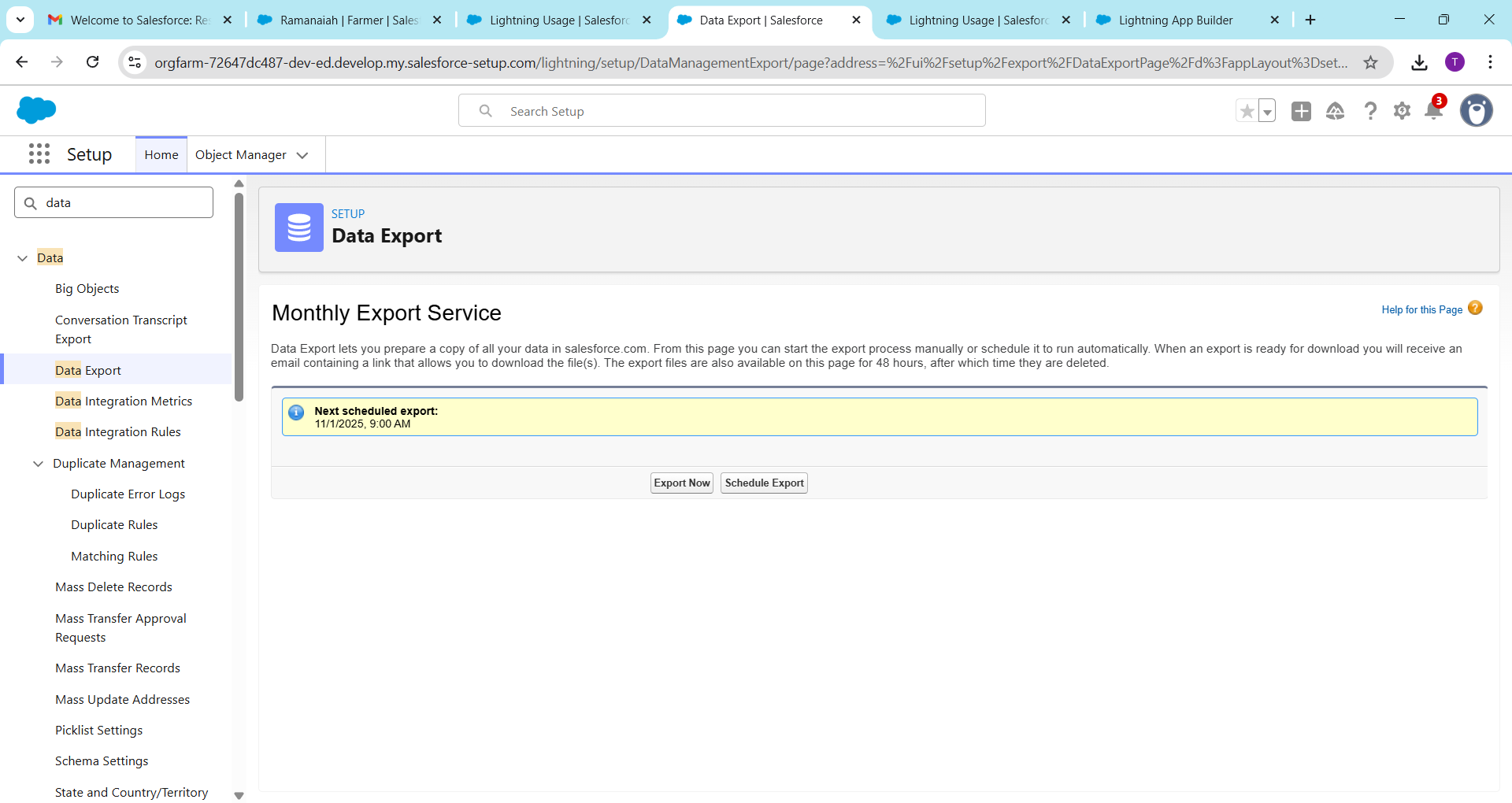
**Options:**

1. **Manual Export:**  
   Setup → Data Export → Choose objects → Export Now
2. **Scheduled Export:** Weekly or monthly automated exports
3. Save CSV / ZIP files securely

**Best Practices:**

* Backup **critical objects** like Farmer\_\_c, Order\_\_c, Buyer\_\_c, Subsidy\_Application\_\_c
* Keep **versioned backups** in case of rollback needs

**Example:** Backup all **pending subsidy records** before making a mass update.



**5️⃣ Change Sets (Deploy Customizations Between Orgs)**

**Purpose:** Move customizations (fields, flows, layouts) from **Sandbox → Production**.

**Step-by-Step Instructions:**

1. In **Sandbox → Setup → Outbound Change Sets → New Change Set**
2. Add components (objects, fields, flows, record pages)
3. Upload to **Production org**
4. In **Production → Inbound Change Sets → Validate & Deploy**

**Best Practices:**

* Test in Sandbox before deploying to Production
* Use descriptive **names and version notes** for each change set

**Example:** Deploy a new **Farmer Record Page with custom LWC** from Sandbox to Production.

**6️⃣ Unmanaged vs Managed Packages**

| **Type** | **Description** | **Use Case** |
| --- | --- | --- |
| **Unmanaged Package** | Editable source code | Internal projects or learning |
| **Managed Package** | Protected code, includes license management | AppExchange apps or ISV products |

**Example:** Use a **managed package** for a **crop subsidy calculator**, while using **unmanaged packages** for your own custom flows and LWCs.

**7️⃣ ANT Migration Tool (Command-line Deployment)**

**Purpose:** Automate deployment of metadata between orgs using **XML + CLI**.

**Step-by-Step Instructions:**

1. Install **Java + ANT + Salesforce Migration Tool**
2. Configure:
   * build.properties → Org credentials
   * build.xml → Metadata components and deployment targets
3. Run commands:
   * ant retrieve → Get metadata from org
   * ant deploy → Deploy to target org

**Best For:**

* Large deployments
* CI/CD pipelines
* Repeated or automated updates

**8️⃣ VS Code & Salesforce DX (SFDX)**

**Purpose:** Modern development and deployment for Salesforce developers.

**Step-by-Step Instructions:**

1. Install **VS Code + Salesforce Extensions**
2. Authenticate your org:
3. sfdx force:auth:web:login -a SandboxOrg
4. Pull metadata from org:
5. sfdx force:source:retrieve -m ApexClass, LightningComponentBundle
6. Make changes → Push to org:
7. sfdx force:source:push

**Best Practices:**

* Use **scratch orgs** for testing
* Maintain **version control (Git)**
* Integrate with **CI/CD pipelines**

**Example:** Developer builds **LWC + Apex** in VS Code → Pushes to Sandbox → Tested → Deployed to Production.