Phase 8: Data Management & Deployment

CONNECT Student Success Platform - Salesforce CRM Implementation

8.1.Data Import Wizard

Use Case: The Data Import Wizard in Salesforce is used to efficiently import data records into both standard and custom objects without the need for technical expertise. This phase involved importing the provided Indian-origin sample data into custom objects for the Student Success Platform, such as Academic Progress, Student Feedback, Peer Partnership, and Student Intervention. The goal was to populate Salesforce with initial data for operational and reporting use.

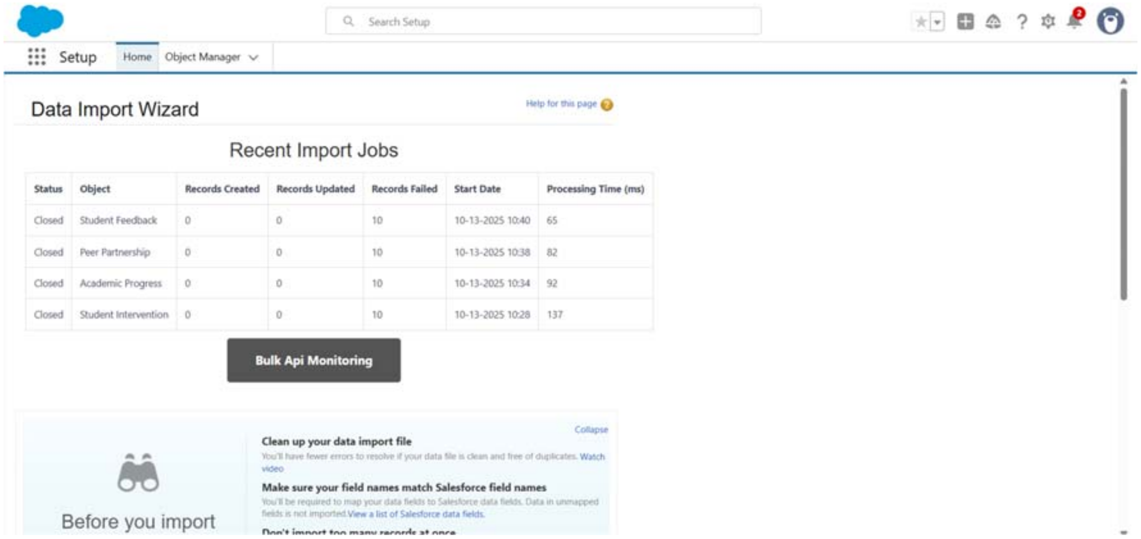
Implementation Steps:

1. **Prepare CSV Files** - Created CSV files matching Salesforce custom objects with Indian-origin data records.
2. **Access the Data Import Wizard**
   → Go to Setup → Quick Find "Data Import Wizard" → Open Data Import Wizard.
3. **Launch Wizard**
   → Click *Launch Wizard* button.
4. **Select Object to Import**
   → Choose *Custom Objects* tab → Select target custom object (e.g., Student Intervention).
5. **Choose Import Action**
   → Select *Add new records* action → Click *Next*.
6. **Upload CSV File**
   → Choose the corresponding CSV file (e.g., student_intervention_import.csv) → Click *Next*.
7. **Map Fields**
   → Verify and edit field mappings from CSV columns to Salesforce fields.
8. **Start Import**
   → Confirm mappings and start import → Wait for confirmation.
9. **Monitor Results**
   → Check status and details in *Bulk Data Load Jobs* → Review success and error logs.
10. **Repeat for Each Object**
    → Perform steps 4-9 for other objects: Academic Progress, Student Feedback, Peer Partnership.

Result:

- Data import successfully completed for all four custom objects.

- Records created in Salesforce with relevant relationships and accurate field mappings.
- Import errors occurred but were reviewed and adjusted as part of the learning process.
- Salesforce now contains populated data ready for platform operations, workflows, and reporting.



Data Import Wizard

Recent Import Jobs

Status	Object	Records Created	Records Updated	Records Failed	Start Date	Processing Time (ms)
Closed	Student Feedback	0	0	10	10-13-2025 10:40	65
Closed	Peer Partnership	0	0	10	10-13-2025 10:38	82
Closed	Academic Progress	0	0	10	10-13-2025 10:34	92
Closed	Student Intervention	0	0	10	10-13-2025 10:28	137

Before you import

Clean up your data import file
You'll have fewer errors to resolve if your data file is clean and free of duplicates. Watch video

Make sure your field names match Salesforce field names
You'll be required to map your data fields to Salesforce data fields. Data in unmapped fields is not imported. View a list of Salesforce data fields.

Don't import too many records at once

8.2 Data Loader

Use Case: Salesforce Data Loader is a desktop application used to import, update, export, and delete large amounts of data in Salesforce records efficiently. It is mainly used for bulk data management, such as migrating thousands or millions of records, cleaning up data, or extracting data for reporting purposes.

When It Is Used:

- When working with very large data sets (50,000+ records), or up to millions of records at once.
- When importing into objects or fields not supported by the Data Import Wizard.
- For scheduling regular data loads (like nightly or weekly imports and exports).
- When complex field mappings or frequent data operations are needed.
- For exporting data for backups or migrating data between Salesforce orgs.

Advantages:

- Can handle very large volumes of data quickly and efficiently.
- Supports all objects—standard and custom.
- Allows field mapping and upsert operations (insert/update at once).

- Can schedule and automate data management tasks.
- Offers detailed success/error logs for troubleshooting.
- More flexibility and control than the Data Import Wizard.

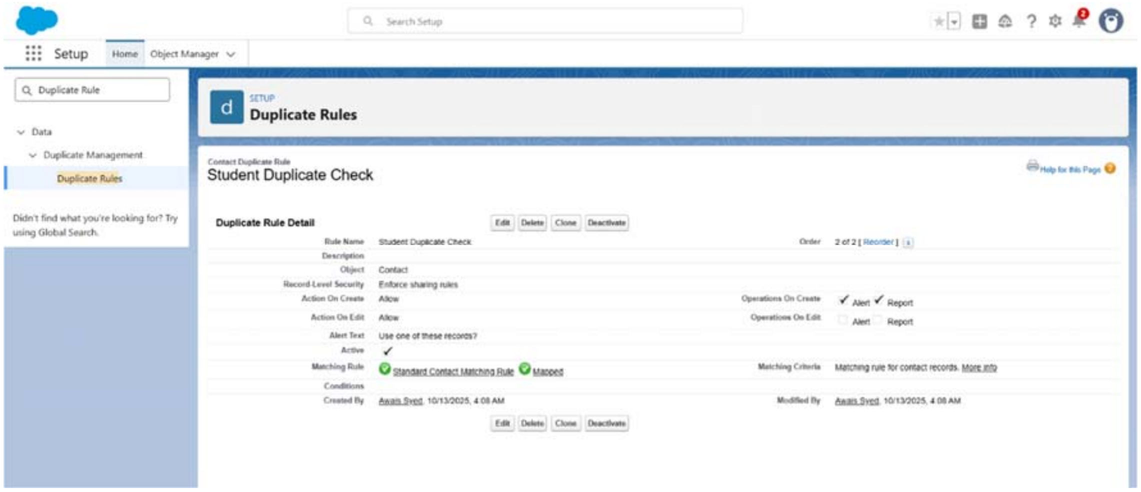
8.3 Duplicate Rules

Use Case: Duplicate Rules are used in Salesforce to help keep data clean by identifying and managing duplicate records when users create or edit Contacts, Leads, Accounts, or other objects. This prevents confusion, wasted effort, and bad reporting by ensuring each person or account only exists once in the system.

Implementation Steps:

- A rule named **Student Duplicate Check** was created for the **Contact** object.
- **Enforce sharing rules** was selected for security.
- On create, Salesforce allows saving duplicates, but shows an alert and includes cases in duplicate reports.
- Standard Contact Matching Rule was selected (checks fields like name, email, and phone).
- The rule was saved and activated, now working for the whole org.

Result: Now, when a user tries to create a new Contact that looks similar to an existing one, Salesforce will display a warning asking: "Use one of these records?" and the system will log duplicate cases for reporting. This improves data quality and prevents unnecessary duplicate records.



Contact Duplicate Rule
Student Duplicate Check

Duplicate Rule Detail

Rule Name	Student Duplicate Check	Order	2 of 2 [Reorder]
Description			
Object	Contact		
Record-Level Security	Enforce sharing rules		
Action On Create	Allow	Operations On Create	✓ Alert ✓ Report
Action On Edit	Allow	Operations On Edit	□ Alert □ Report
Alert Text	Use one of these records?		
Active	✓		
Matching Rule	Standard Contact Matching Rule Mapped	Matching Criteria	Matching rule for contact records. More info
Conditions			
Created By	Awais Syed, 10/13/2025, 4:08 AM	Modified By	Awais Syed, 10/13/2025, 4:08 AM

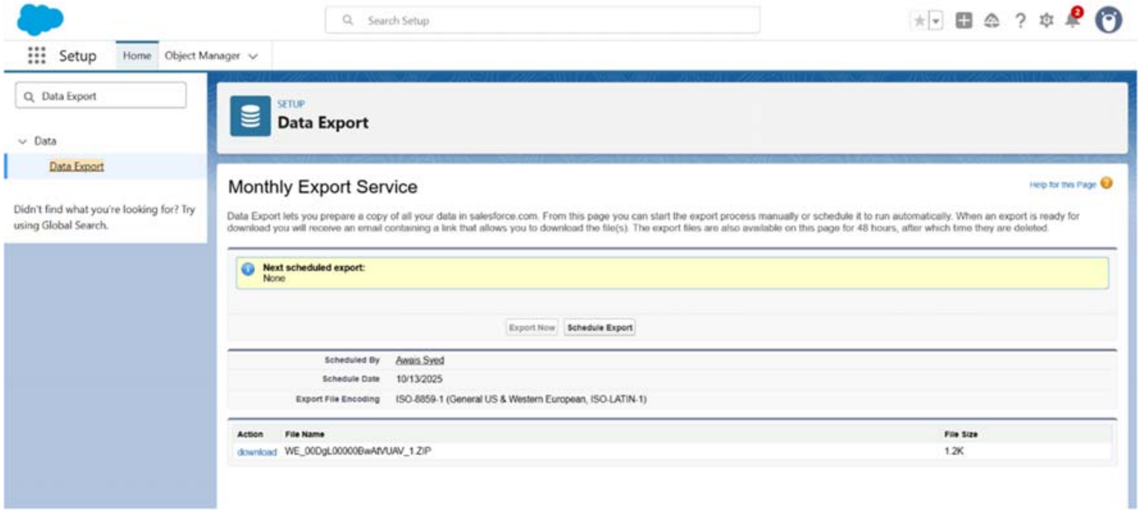
8.4 Data Export & Backup

Use Case: The Data Export & Backup feature in Salesforce is used to keep regular copies of all important data. If data is lost, damaged, or needs to be moved to another system, these backup files help you recover everything easily. This protects the organization from losing student, academic, or feedback information.

Implementation Steps:

1. Go to **Setup** and search for **Data Export** under Data Management.
2. Click **Export Now** for a manual backup, or **Schedule Export** to set up automatic backups.
3. Select the custom objects you want to back up, such as **Academic Progress, Student Feedback, Peer Partnership, Student Intervention**.
4. Click **Start Export**. Wait for Salesforce to prepare the files.
5. Download the ZIP file when ready. Store it in a safe folder named with today's date (like "SalesforceBackup-20251013.zip").

Result: Your Salesforce data from selected objects is now saved in a backup ZIP file.
If something ever happens to your data, you can restore it using these CSV files, keeping your records safe and secure.



Didn't find what you're looking for? Try using Global Search.

Monthly Export Service

Data Export lets you prepare a copy of all your data in salesforce.com. From this page you can start the export process manually or schedule it to run automatically. When an export is ready for download you will receive an email containing a link that allows you to download the file(s). The export files are also available on this page for 48 hours, after which time they are deleted.

Next scheduled export:
None

Scheduled By	Awais Syed
Schedule Date	10/13/2025
Export File Encoding	ISO-8859-1 (General US & Western European, ISO-LATIN-1)

Action	File Name	File Size
download	WE_00DgL00000BwAfVUAV_1.ZIP	1.2K

8.5 Change Sets

Use Case: Change Sets in Salesforce are used to move customizations and development work between connected orgs (e.g., from sandbox to production). They do not move data, only metadata (like custom fields, objects, automations, page layouts, etc.), supporting team-based and multi-org project management.

Implementation Steps:

Not needed for this single-org data management and configuration project. If your team works in a sandbox and needs to transfer customizations to production, then follow these steps:

- In Setup, search "Change Sets" and select "Outbound Change Sets".
- Create a new outbound change set; add components (fields, layouts, etc.) to the change set.
- Upload to the target org; in the target org, navigate to "Inbound Change Sets" and deploy.

Result:

No Change Sets are created or deployed for this project, because all work is completed inside one org.

8.6 Unmanaged vs Managed Packages

Use Case: Salesforce packages help developers bundle custom objects, fields, apps, and Apex code for sharing or selling. **Managed Packages** are locked for editing and used for commercial, upgradeable app delivery. **Unmanaged Packages** allow full editing and are mainly for internal, free, or one-off distribution.

Implementation Steps:

No steps required for this project, because everything is managed in a single org and there is no need to distribute or install packages.

- If building components to share, you would:
  - Go to Setup → **Create Package**
  - Choose **Unmanaged** or **Managed**
  - Add components like custom objects, fields, Visualforce pages, etc.
  - Upload/share or install in another org

Result:

No packages (managed or unmanaged) are included or installed for this work, as the project does not require distributing code or components to other Salesforce orgs.

8.7 ANT Migration Tool

Use Case: The ANT Migration Tool is used for moving metadata (such as custom objects, fields, layouts, Apex code, etc.) between Salesforce orgs or between a local directory and a Salesforce org using the command line. It is helpful for automating large or repeatable deployments, managing metadata source control, or when certain components are not supported by Change Sets.

How to Use:

1. **Prerequisites:**
   - Install Java (at least version 1.7 or above).
   - Download and install Apache Ant on your computer.

- Download the ANT Migration Tool (zip) from Salesforce.

2. **Set Up Environment Variables:**
   - Set ANT_HOME (your Ant installation path).
   - Set JAVA_HOME (your JDK installation path).
3. **Configure the Tool:**
   - Unzip both Apache Ant and the ANT Migration Tool.
   - Copy the ant-salesforce.jar file from the tool's zip into Ant’s lib directory.
4. **Prepare for Deployment:**
   - Set up a build.properties file with your Salesforce credentials and org details.
   - Create a build.xml file, which contains the Ant scripts for retrieve/deploy tasks.
   - Create a package.xml file listing the metadata components to retrieve or deploy.
5. **Typical Deployment Steps:**
   - To retrieve metadata from a source org:
     Run ant retrieveCode
   - To deploy metadata to a target org:
     Run ant deployCode
   - To delete components, add a destructiveChanges.xml and run deploy as above.
6. **Download and Logs:**
   - Download logs after each deployment to verify which components
     were successfully moved and fix any deployment errors.

Important Note

- **The ANT Migration Tool is retired as of Spring '24. It still works for now, but Salesforce recommends switching to Salesforce CLI (SFDX) for modern and supported deployments.**

Advantages

- Automates deployments using scripts.
- Supports components not available with Change Sets.
- Good for continuous integration and large/developer teams.
- Allows deleting metadata, not just deploying.
- Works for unrelated Salesforce orgs, not just sandboxes.

This tool is mainly useful for advanced or team-based Salesforce development, but for most admin-level or single-org work, ANT Migration Tool is not required.

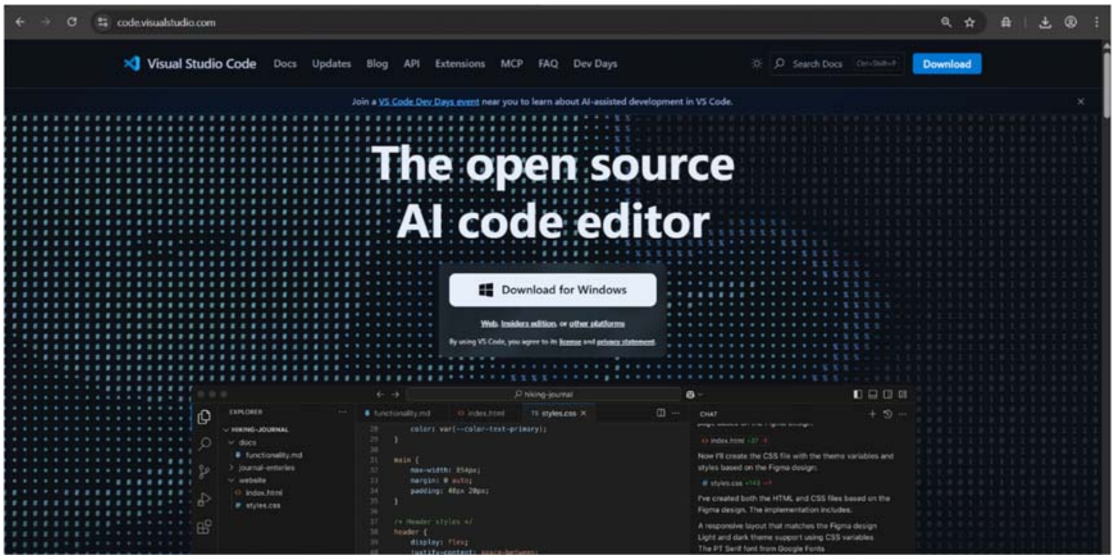
8.8 VS Code & SFDX

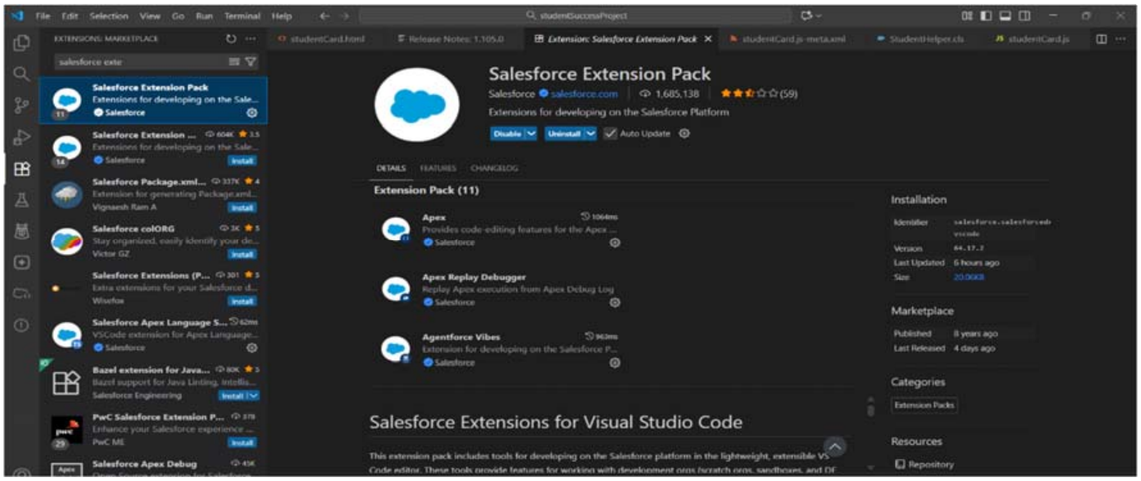
Use Case: Visual Studio Code (VS Code) with Salesforce SFDX (Salesforce Developer Experience/CLI) is used by Salesforce developers to write code, manage metadata, automate deployments, version control, and debug complex logic in a modern, powerful environment. It enables fast coding, easy management of big projects, and integration with source control like Git. It's the main tool for creating Lightning Web Components, Apex classes, triggers, and bulk configuration changes.

How to Use:

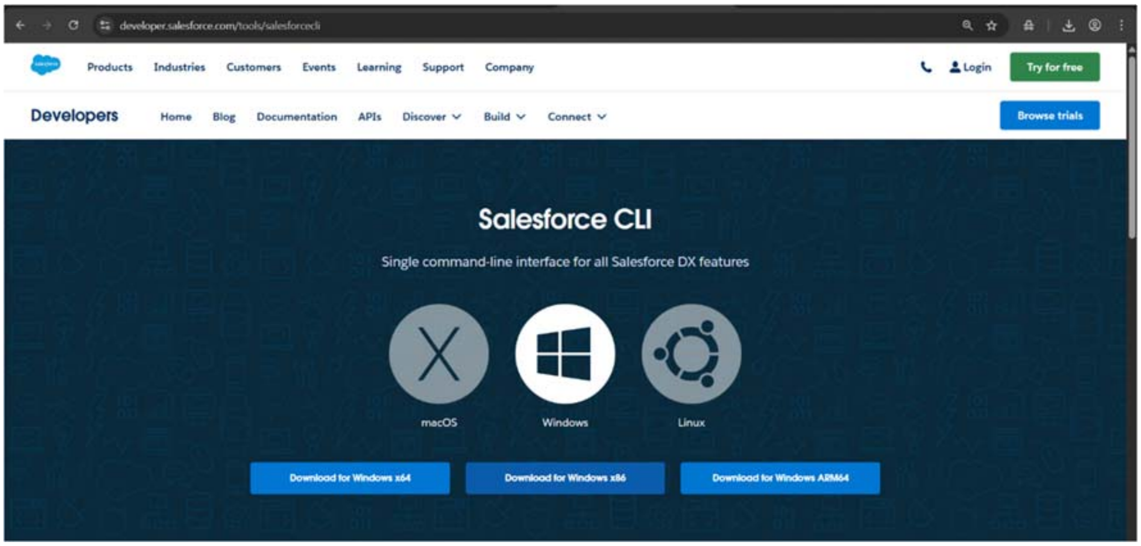
1. **Set Up VS Code and SFDX:**

- Download and install VS Code for your operating system.
- Install the Salesforce Extension Pack from the VS Code marketplace (search and click Install).


code.visualstudio.com
Visual Studio Code
Docs
Updates
Blog
API
Extensions
MCP
FAQ
Dev Days
Search Docs
Download
Join a VS Code Dev Days event near you to learn about AI-assisted development in VS Code.
The open source AI code editor
Download for Windows
Web, Insiders edition, or other platforms
By using VS Code, you agree to its license and privacy statement.


EXTENSIONS: MARKETPLACE
salesforce exte
Salesforce Extension Pack
Salesforce Extension ...
Salesforce Package.xml...
Salesforce colORG
Salesforce Extensions (P...
Salesforce Apex Language S...
Bazel extension for Java...
PwC Salesforce Extension P...
Salesforce Apex Debug
Install
Extension: Salesforce Extension Pack
Salesforce Extension Pack
Salesforce
salesforce.com
1,685,138
(59)
Extensions for developing on the Salesforce Platform
Disable
Uninstall
Auto Update
DETAILS
FEATURES
CHANGELOG
Extension Pack (11)
Apex
Apex Replay Debugger
Agentforce Vibes
Installation
Marketplace
Categories
Extension Packs
Resources
Repository
Salesforce Extensions for Visual Studio Code
This extension pack includes tools for developing on the Salesforce platform in the lightweight, extensible VS Code editor.

- Download and install Salesforce CLI (SFDX) for interacting with Salesforce from the command line.



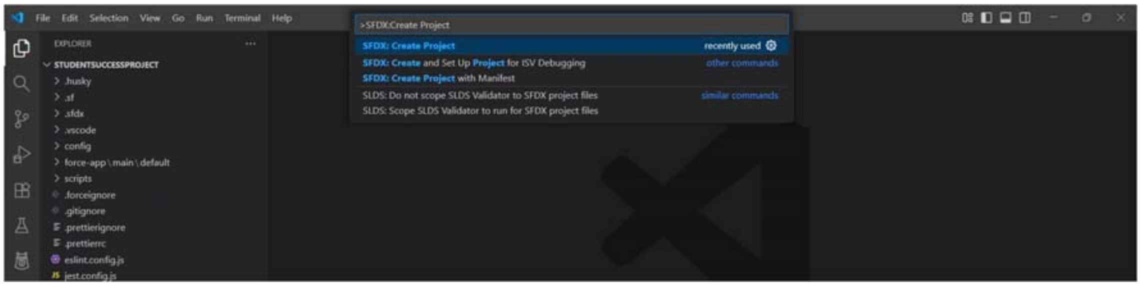
developer.salesforce.com/tools/salesforcecli

Salesforce CLI

Single command-line interface for all Salesforce DX features

2. **Create a Salesforce DX Project:**

- Open VS Code.
- Press Ctrl+Shift+P to open the Command Palette.
- Type and select **SFDX: Create Project with Manifest** or sf project generate --name MyProject.
- Enter a project name and location. VS Code will create standard folders for code, config, and manifest files.



```
>SFDX:Create Project
SFDX: Create Project                                          recently used
SFDX: Create and Set Up Project for ISV Debugging             other commands
SFDX: Create Project with Manifest
SLDS: Do not scope SLDS Validator to SFDX project files       similar commands
SLDS: Scope SLDS Validator to run for SFDX project files
```

```
EXPLORER
STUDENTSUCCESSPROJECT
  > .husky
  > .sf
  > .sfdx
  > .vscode
  > config
  > force-app\main\default
  > scripts
    .forceignore
    .gitignore
    .prettierignore
    .prettierrc
    eslint.config.js
    jest.config.js
```

3. **Authorize and Connect to Salesforce Org:**

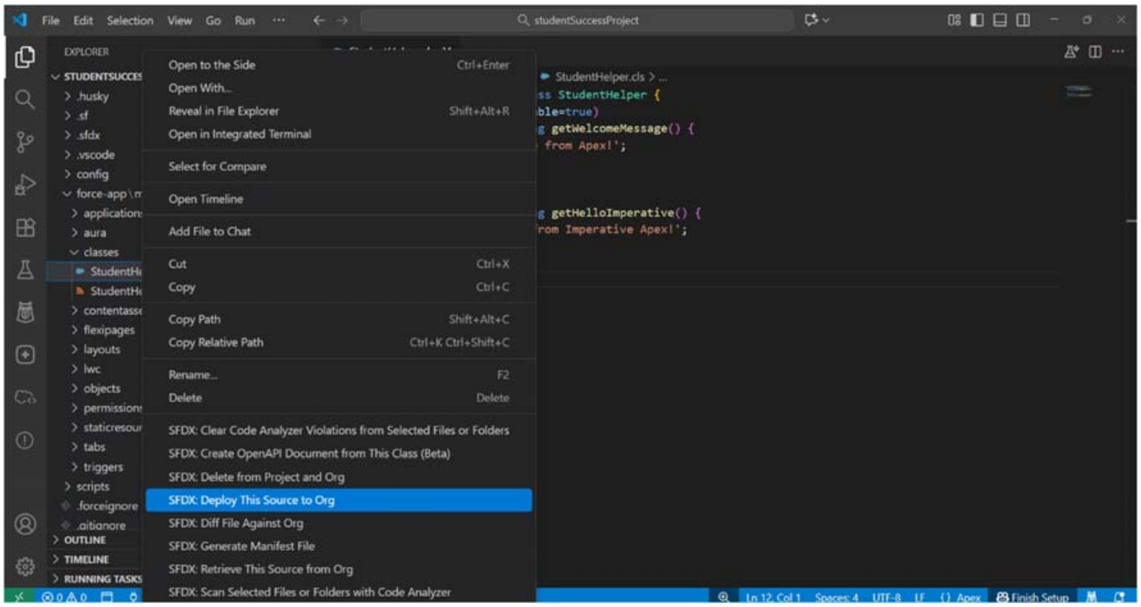
- Press Ctrl+Shift+P, type **SFDX: Authorize an Org** and follow the browser prompts to log in.

4. **Retrieve Metadata from Org:**

- Edit the package.xml file in the manifest folder to specify which components to pull.
- Right-click package.xml → select **SFDX: Retrieve This Source from Org**.

5. **Develop and Deploy:**

- Use VS Code to write code (Apex, LWC, Aura, triggers).
- Right-click files or folders and choose **SFDX: Deploy This Source to Org** to send your code and metadata to Salesforce.



```
ss StudentHelper {
ble=true)
g getWelcomeMessage() {
 from Apex!';


g getHelloImperative() {
rom Imperative Apex!';
```

6. **Run SOQL/SOSL Queries:**

- Highlight a query in your code, open the Command Palette, select **SFDX: Execute SOQL Query**, and view results in Output tab.

7. **Version Control and Automation:**

- Initialize a Git repository for your project using the built-in VS Code tools.
- Commit and push changes to keep project versions safe and easy to collaborate.

Extra Features

- Built-in interactive debugger for Apex code, making error-finding easy.
- Automatic syntax highlighting and suggestions.
- Supports scratch org and sandbox development for testing changes safely.
- Automation: Write deployment scripts and CI/CD workflows using SFDX commands.

VS Code & SFDX are the most modern tools for Salesforce development and admin automation, making the platform easy to customize, scale, and manage for all types of users.