

WEDNESDAY, JULY 24 – 8:00 AM

# Five Tips Every Admin Should Know about Visual Studio Code

Robert Davis / Cloudy Room / Hike2

JULY 22-24, 2024





**CGI**

**Deloitte.**

**foxit**



**PSQuote™**

Professional Services Quoting by CLD Partners

 **DRIVE**

  
**RaagviTech**

  
**OneMerge**

 **SightLane**



**celigo**



  
**HIKE2**

  
**SUMMIT  
TECHNOLOGIES**



 **DataGroomr**

  
**COPADO**

**THANK YOU to our Sponsors**

# [PRESENTER]



**Robert Davis**  
*Solution Architect @*  
**Hike2**



# AGENDA



01  
02  
03

Moving Data Btwn Salesforce Orgs

How to load records into Custom Metadata Records

Comparing differences in Text files

04  
05

Querying the Salesforce Objects

Building better flows with Lightning Flow Scanner





# What is Visual Studio Code?

- Also known as VS Code
- A free and open source text editor that can be used to edit, analyze and manage text files.
- Metadata, the controlling basis of Salesforce, are text files specifically formatted that the platform can read.
- Salesforces preferred code editor

# Terms

- Repos - Git / Bitbucket
- Command Pallet - Command Shift - P
  - It is the replacement in many cases for the command line.
  - Powerful tool to execute commands. Equivalent of Command Line you see developer uses.
  - Salesforce is similar to setup in VS Code
- Folder Scaffolding - same as Windows Explorer
- Code Editor - Notepad++, Maven, Sublime
- Extension - a bit of code that automates a task - similar to AppExchange



# Why Should I use Visual Studio Code

- When used with the Salesforce CLI, and Extensions written solve specific problems one can do things other than simply write code.
- Some of the things we can do:
  - Move data between Salesforce Orgs
    - Individually or as a group of metadata
  - Load data into custom metadata
  - Query Salesforce
  - Sign into Salesforce instances





# What are Extensions in VS Code?

1

They provide additional functionality.

2

When someone repeatedly runs into repetitive tasks the result seems to be an automation or tool to help solve that repetitive task.

3

Examples:

- Salesforce Diff to show differences in Salesforce Orgs
- Salesforce Profile Modifier to add, edit or remove apex classes, VF pages, Object and Fields to Profiles
- CSV to Table converts text files to a Table
- Excel Viewer allows editing Excel Spreadsheets in VS Code
- CSV to JSON Converter to convert formats
- Salesforce Flow Visualizer provides ability to visualize flow in VS Code



# Prerequisites



Visual Studio Code



Salesforce Command Line



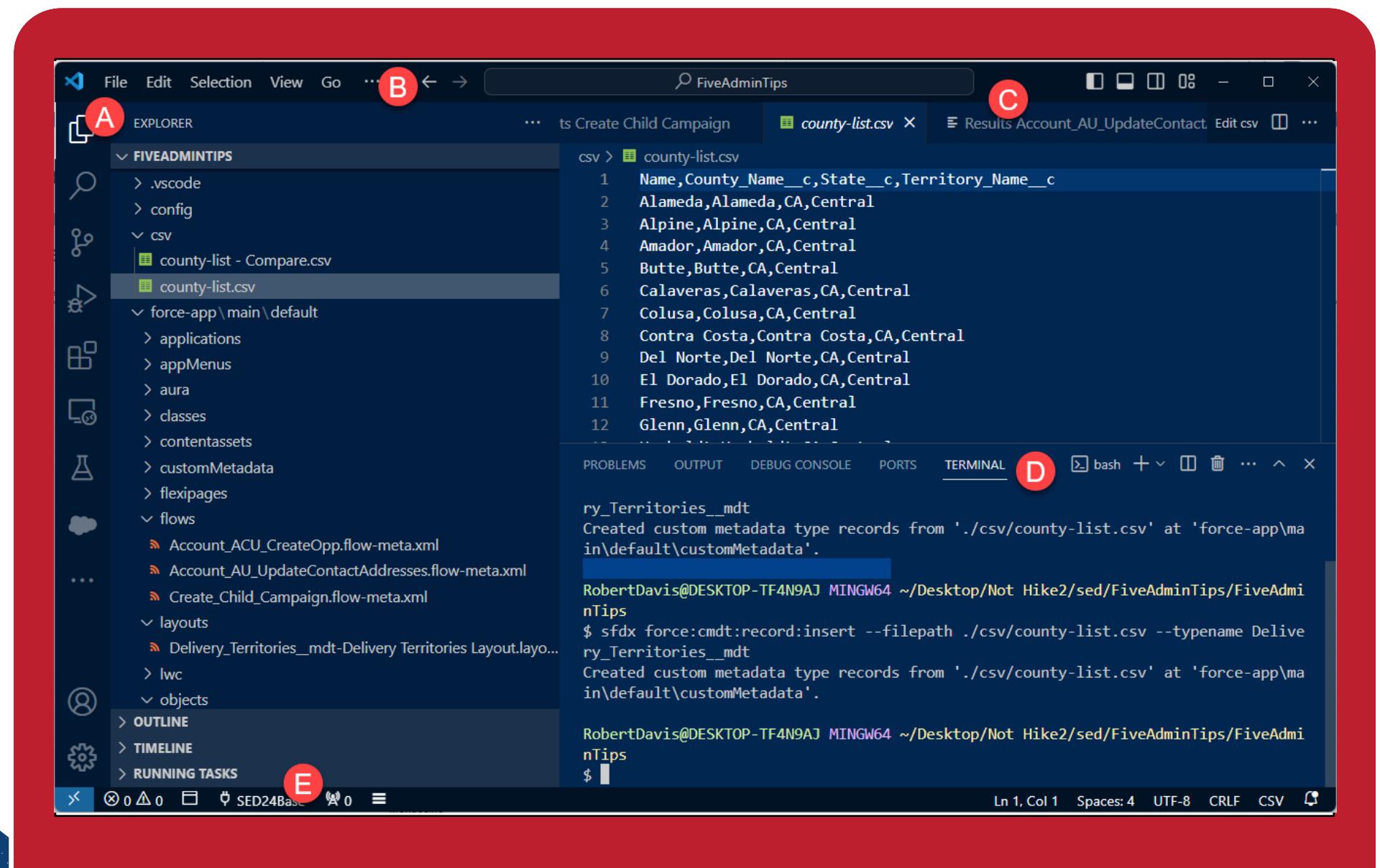
Salesforce Extension Pack



JULY 22-24, 2024



# Navigating Visual Studio Code



A – Activity Bar

B – Menu

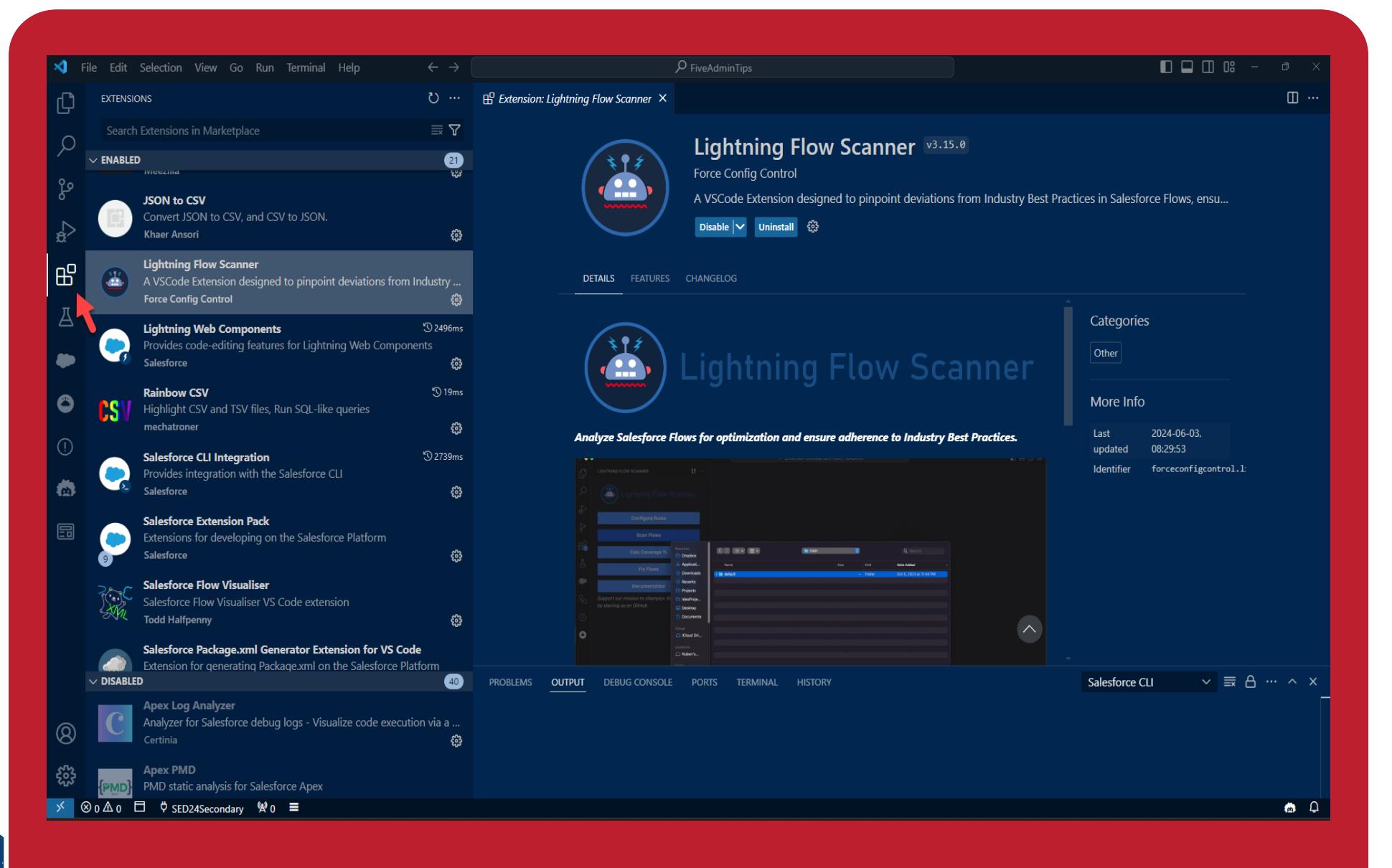
C – Editor Groups

D – Command Line / Terminal

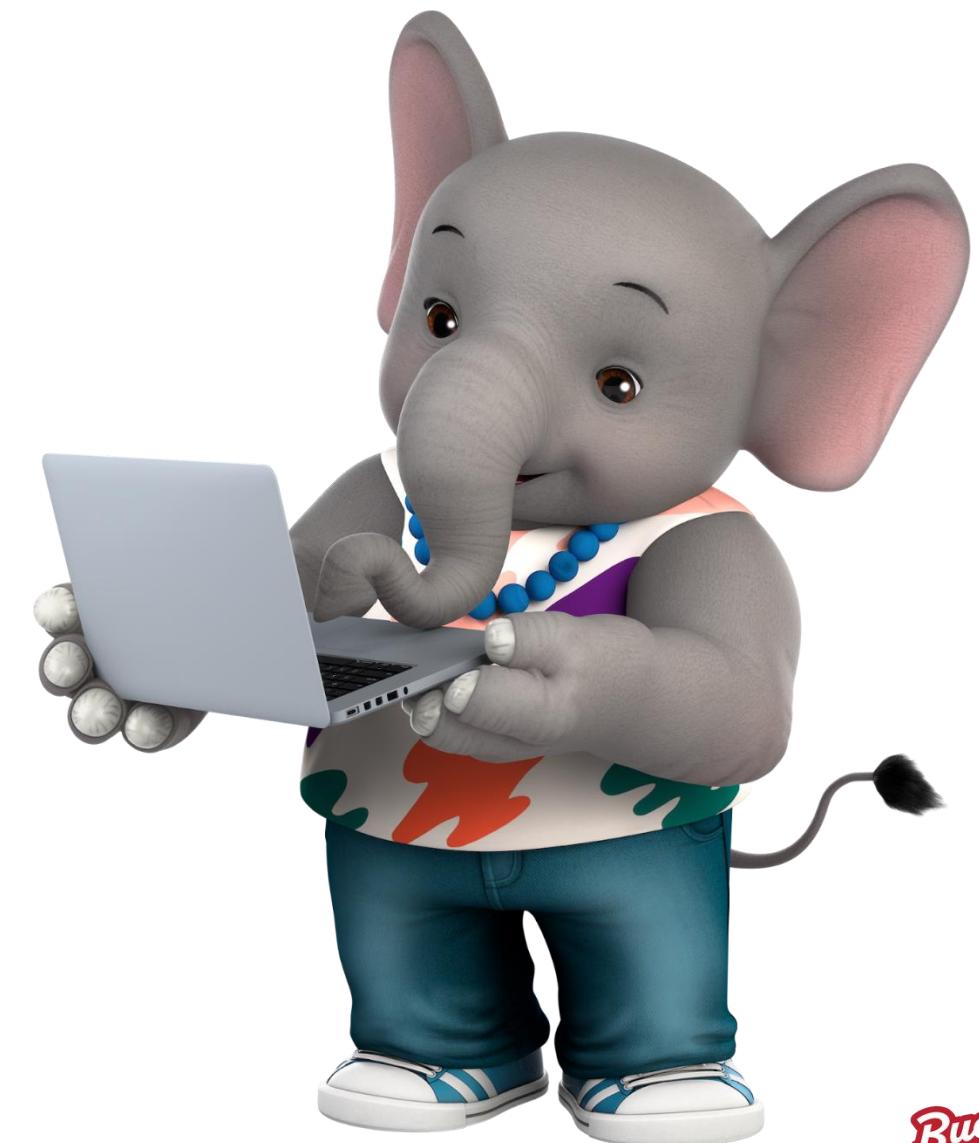
E - SideBar



## Find Extensions



Can find and select extensions for installation





## RETRIEVE

- Via Project Manifest
  - Right click the package.xml file and select SFDX:Retrieve Source in Manifest from Org
- Individual Metadata
  - Find in Org Browser in Visual Studio Code
  - Select the object and select the little cloud icon beside its name.

## DEPLOY

- Via Project Manifest
  - Right Click the package.xml file and select SFDX: Deploy Source in Manifest to Org
- Individual Metadata
  - Right click the component and select SFDX: Deploy Source to Org

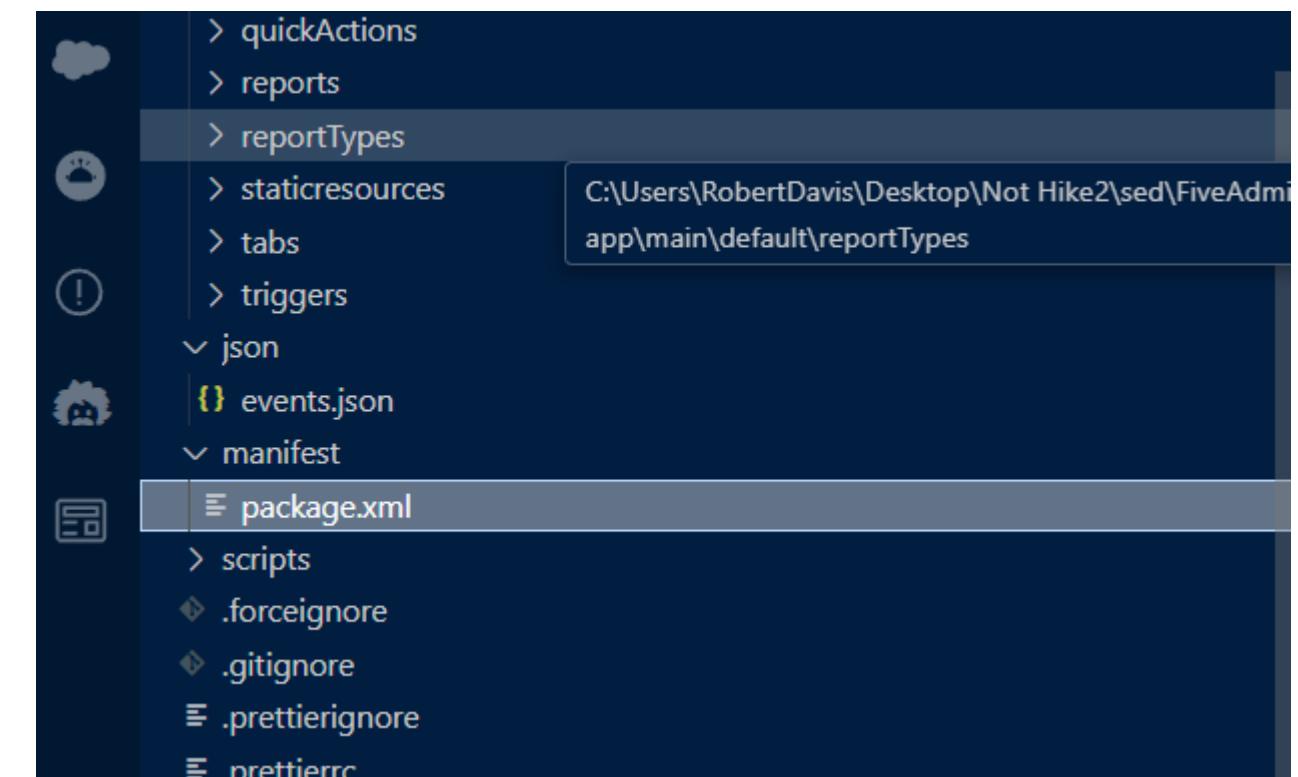


# Getting Salesforce Metadata into Visual Studio Code

A screenshot of the Visual Studio Code interface. The Explorer sidebar on the left shows a project structure for 'FIVEADMIN TIPS' with various metadata types like flows, layouts, objects, and reports. A red arrow points to the 'manifest' folder in the sidebar. The main editor area displays the contents of a 'package.xml' file, which is an XML manifest for a Salesforce package. The file includes definitions for Apex classes, components, pages, test suites, triggers, and app menus.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Package xmlns="http://soap.sforce.com/2006/04/metadata">
<types>
<members*></members>
<name>ApexClass</name>
</types>
<types>
<members*></members>
<name>ApexComponent</name>
</types>
<types>
<members*></members>
<name>ApexPage</name>
</types>
<types>
<members*></members>
<name>ApexTestSuite</name>
</types>
<types>
<members*></members>
<name>ApexTrigger</name>
</types>
<types>
<members*></members>
<name>AppMenu</name>
</types>
<types>
<members*></members>
<name>AuraDefinitionBundle</name>
</types>
<types>
```

- Can use project manifest
- Can pull down individually Org Browser



# Where Can VS Code Be Used?

- Sandboxes
- Production
  - Will run all the test just like deploying with a Changeset
  - All same requirements as any other push into Production



# Loading Data into Custom Metadata Objects

- Mass loading is not supported by Data Loader
- Loading in a few hundred records can be time consuming
- Usually to load territories or zips for some sort of Flow or Apex lookup
  - Country Mappings
  - Courthouse Mappings
- Command Line:
  - Sfdx force:cmdt:record:insert –filepath FIVEADMINTIPS/csv/county-list.csv –typename Delivery\_Territories\_\_mdt
- Deploy to the Salesforce Org



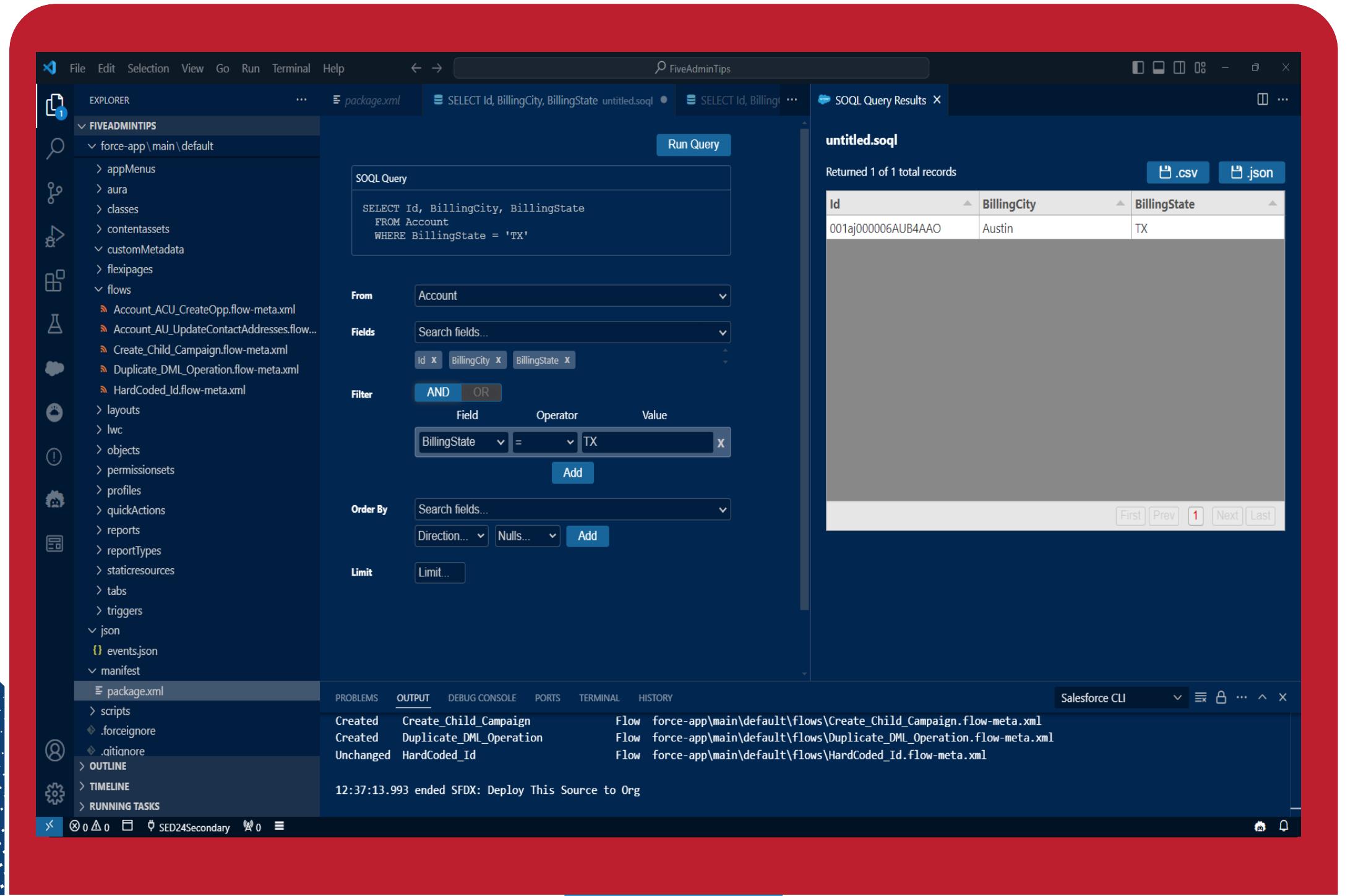
# Differences in **TEXT FILES**

Lorem

- Large text file and you need to know if they are the same.
  - Example - troubleshooting long token
  - Example - comparing new values before uploading
- Using Visual Studio Code Extension named [Diff Tool](#) by jinsihou
  - Lots of Diff checker extensions
  - Like this one because I do not have to search for the other file like some ask
- Open the two files in VS Code
  - Control + Shift + P DiffTool Mark file 1
  - Control + Shift + P DiffTool Mark file 2
- Will get a file with differences highlighted



# SOQL



The screenshot shows the Visual Studio Code interface with a red overlay. On the left is the Explorer sidebar with a tree view of Salesforce metadata, including Flow definitions like 'Account\_ACU\_CreateOpp.flow-meta.xml' and 'Create\_Child\_Campaign.flow-meta.xml'. The main area contains two tabs: 'SOQL Query Results' and 'untitled.sql'. The 'untitled.sql' tab displays a SOQL query:

```
SELECT Id, BillingCity, BillingState  
FROM Account  
WHERE BillingState = 'TX'
```

The results table shows one record:

ID	BillingCity	BillingState
001aj000006AUB4AAO	Austin	TX

Below the results are buttons for CSV and JSON export.

- Visual Studio Code SOQL by Salesforce
- Easily Answer: "How many records have..."
- Easily Export to CSV or JSON



# SOQL More Complicated

The terminal window shows the following content:

- Result: SFDX: Execute SOQL Query with Currently Selected Text
- recently used: mail, Ho Untitled-1
- 1 Scan Flows
- CSV: Edit csv
- SFDX: Create Query in SOQL Builder
- DiffTool: Mark 2nd file
- DiffTool: Mark 1st file
- SFDX: Authorize an Org
- SFDX: Create Project with Manifest
- Convert JSON to CSV
- SFDX Package.xml Generator: Choose Metadata Components
- Org Browser: Focus on Metadata View
- Apex Static Analysis: On Workspace
- Apex Static Analysis: Clear Problems
- View Default Flow Rules
- Extensions Manager: Authorized Extension URLs

PROBLEMS    OUTPUT    DEBUG CONSOLE    PORTS    TERMINAL

```
Querying Data... done
001ai000006AkMeAAK Take Llorrac null
```

- Open Text File
- Create a SOQL :
  - SELECT Id, Name, (SELECT AccountId, Name, HomePhone FROM Contacts) FROM Account
- Select Query
- SFDX: Execute SOQL Query with Currently Selected Text



## Flow Scanner

The screenshot shows the Lightning Flow Scanner extension running in a Visual Studio Code window. The sidebar on the left contains icons for file operations, search, navigation, and configuration. The main area is titled "Scan Results" and features a table with columns for "# Results", "Label", "Flow Type", "% Test ...", and "Details". The table lists five findings. At the bottom, there's a terminal window showing a Salesforce CLI command and its output.

# Results	Label	Flow Type	% Test ...	Details
1	Hardcoded Id	AutoLaunchedFlow		Details
1	Duplicate DML Operation	Flow		Details
5	Create Child Campaign	AutoLaunchedFlow		Details
1	Account AU UpdateContactAddresses	AutoLaunchedFlow		Details
4	Account ACU CreateOpp	AutoLaunchedFlow		Details
5				

```
14:00:36.9 sf data:query --query SELECT Id, Name, (SELECT AccountId, Name, HomePhone FROM Contacts) FROM Account
Querying Data... done
ID NAME CONTACTS.ACCOUNTID CONTACTS.NAME CONTACTS.HOMEPHONE
001aj000006AUB4AAO Edge Communications 001aj000006AUB4AAO Sean Forbes null
001aj000006AUB5AAO Burlington Textiles Corp of America 001aj000006AUB5AAO Rose Gonzalez null
001aj000006AUB6AAO Pyramid Construction Inc. 001aj000006AUB6AAO Jack Rogers null
001aj000006AUB7AAO Dickenson plc 001aj000006AUB7AAO Andy Young null
```

- Visual Studio Code Extension Lightning Flow Scanner by Force Fig Control
- Applies Standard Best practices by evaluating and surfacing
- Helps us build better flows



# Flow Visualizer

The screenshot shows the Lightning Flow Scanner application running in a terminal-based IDE. The main window displays a table of scan results:

#	Results	Label	Flow Type	% Test ...	Details
1	1	<a href="#">Hardcoded Id</a>	AutoLaunchedFlow		<a href="#">Details</a>
1	1	<a href="#">Duplicate DML Operation</a>	Flow		<a href="#">Details</a>
5	5	<a href="#">Create Child Campaign</a>	AutoLaunchedFlow		<a href="#">Details</a>
1	1	<a href="#">Account AU UpdateContactAddresses</a>	AutoLaunchedFlow		<a href="#">Details</a>
4	4	<a href="#">Account ACU CreateOpp</a>	AutoLaunchedFlow		<a href="#">Details</a>
5	5				

Below the results table is a terminal window showing a Salesforce CLI command and its output:

```
14:00:36.9 sf data:query --query SELECT Id, Name, (SELECT AccountId, Name, HomePhone FROM Contacts) FROM Account
Querying Data... done
ID NAME CONTACTS.ACCOUNTID CONTACTS.NAME CONTACTS.HOMEPHONE
001aj000006AUB4AAO Edge Communications 001aj000006AUB4AAO Sean Forbes null
001aj000006AUB5AAO Burlington Textiles Corp of America 001aj000006AUB5AAO Rose Gonzalez null
001aj000006AUB6AAO Pyramid Construction Inc. 001aj000006AUB6AAO Jack Rogers null
001aj000006AUB7AAO Dickenson plc 001aj000006AUB7AAO Andy Young null
```



- Flow Visualizer Render Command
- Shows image of process
- Shows variables and formulas
- Flow Visualizer by Todd Halfpenny



JULY 22-24, 2024



# THANK YOU

# RESOURCES

## References

- [How to Setup Visual Studio Code for Salesforce](#) by Apex Hours
  - [Quick Start: Visual Studio Code for Salesforce Development](#) by Trailhead
    - [Visual Studio Code User Interface](#)
    - [Mass Insert Custom Metadata Type Records](#) by SFDC Lessons
    - Write [SOQL Queries](#) by Trailhead

## Extensions

- [Salesforce Package.xml Generator Extension](#) for VS Code by Vignaesh Ram A
  - [Diff Tool](#) by jinsihou
  - [SOQL](#) by Salesforce
- [Lightning Flow Scanner](#) by Force Fig Control
  - [Flow Visualizer](#) by Todd Halfpenny

JULY 22-24, 2024



# Slides

<https://bit.ly/4cNBxqq>

<https://github.com/robtdavis/FiveTipsEveryAdminShouldKnowAboutVSCode/blob/main/Buckeye%20Dreamin%202024%20-%202024070802%20-%20PDF.pdf>

JULY 22-24, 2024

