Einstein Analytics Part 1 - Enable and Intergration

| Summary	In this codelab you will Set up Einstein Analytics, enable features, and bring data in and integration in Salesforce Einstein Analytics.
URL	https://docs.google.com/document/d/1qOQu7t2lahwiFwGusz_sfQZjXQknwFlHoQyOAzdMmfg/edit
Category	Salesforce
Status	First Review
Author	Prof. Sri Krishnamurthy, Uthsav Shetty

Introduction

Duration: 5:00

Your company data lives in different locations, such as Salesforce, ERPs, data warehouses, and log files. When viewed together, all this data can be a gold mine, changing the way you see your business and improving your customers' entire experience. Analytics solves the challenge of combining all this information to answer key questions of your business users.

Analytics is Salesforce—secure, trusted, scalable, and, of course, mobile! It enables everyone in your org to get immediate access to powerful data insights through its intuitive point-and-click visual interface. Whether you're on a desktop browser or a mobile device, you can get the answers you need from your data in order to run your business

What Can You Do with Einstein Analytics?

Analytics is a self-service application that enables you to make sense of large amounts of data. You can explore your data to discover new, unexpected insights. Get instant visualizations that show how your business is doing, and even gaze into the future to see what the coming months might bring. And create dashboards to continually monitor key business metrics based on the latest data
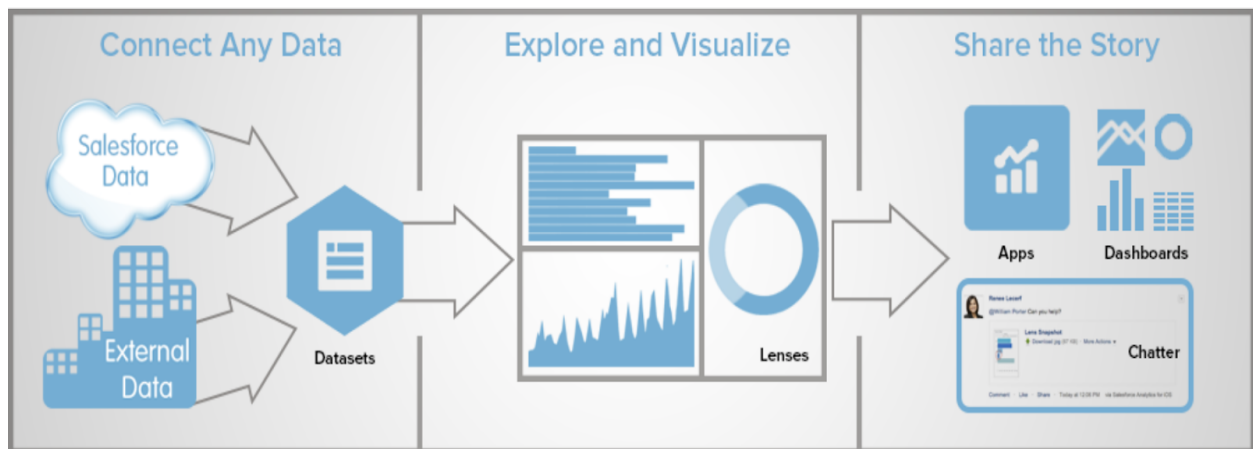
What you'll learn

- How to Set up Security and Access
- Enable Einstein Analytics
- Monitor and Manage your Datasets
- Bringing in your Data
- Prepare your Data

What you'll need

- An Einstein Analytics Developer Account Setup
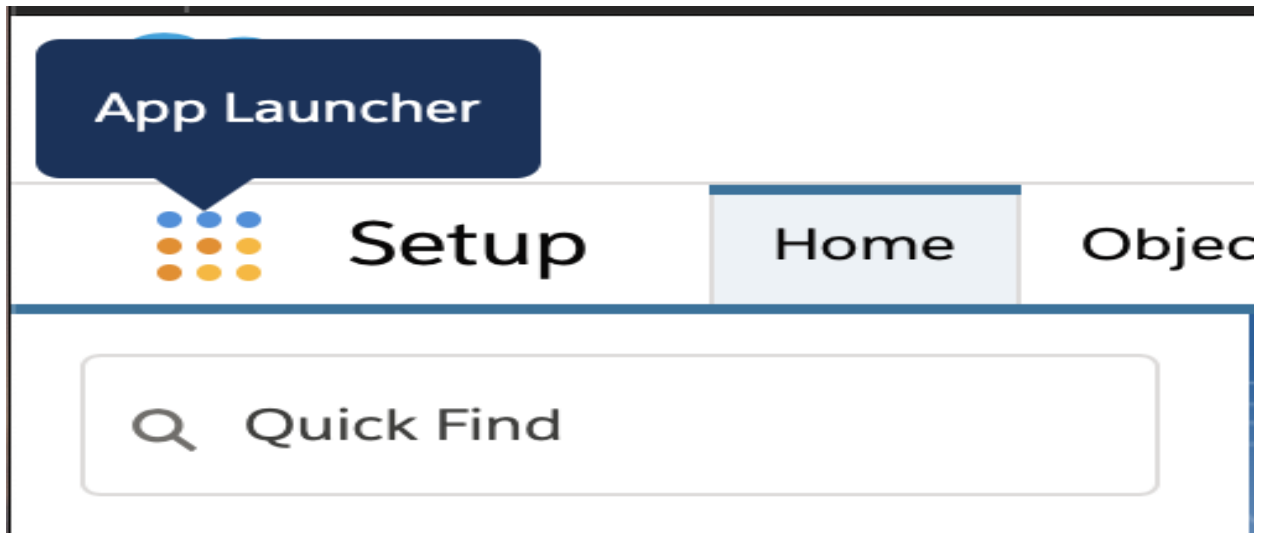

Developer Signup

Duration: 5:00

Please sign up using the link below to sign up for Developer edition

Einstein Analytics Developer
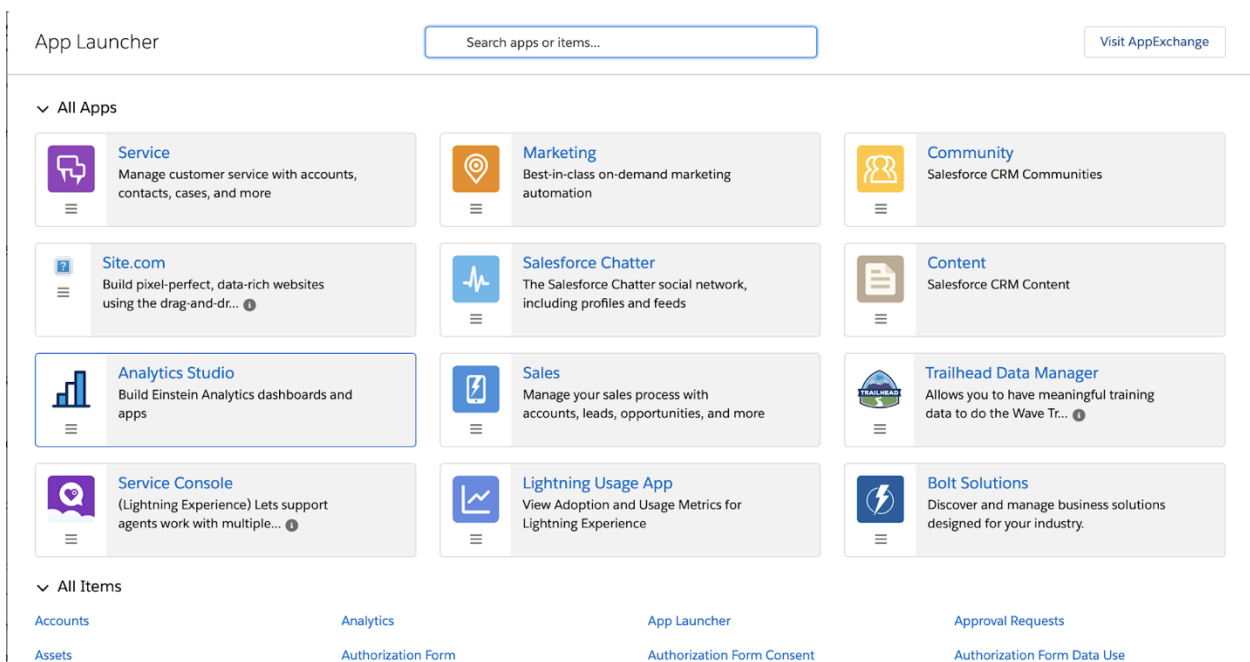
Once you are set up with signup, go to

Salesforce

From Setup click on App Launcher and there should be an App called Analytics Studio

From App Launcher click on Analytics Studio to start Einstein Analytics

Analytics Basics

Duration: 20:00

Security

The Einstein Analytics Platform license includes two prebuilt permission sets:

- **Einstein Analytics Platform Admin** enables all permissions required to administer the Analytics platform, including permissions to enable creating Analytics templated apps and Apps.
- **Einstein Analytics Platform User** enables all permissions required to use the Analytics platform and Analytics templated apps and Apps

The Einstein Analytics Plus license includes two prebuilt permission sets:
- **Einstein Analytics Plus Admin** enables all permissions required to administer the Analytics platform and Einstein Discovery, including permissions to enable creating Analytics templated apps and Apps.
- **Einstein Analytics Plus User** enables all permissions required to use the Analytics platform, Einstein Discovery, and Analytics templated apps and Apps

Clone	Einstein Analytics Platform Admin	Create and customize Einstein Analytics apps, dashboards, datasets and data flows.	Analytics Platform
Clone	Einstein Analytics Platform User	View Einstein Analytics apps and dashboards.	Analytics Platform
Clone	Einstein Analytics Plus Admin	Access to all features enabled by Einstein Analytics Plus license	Einstein Analytics Plus
Clone	Einstein Analytics Plus User	Access to read only features enabled by Einstein Analytics Plus license	Einstein Analytics Plus

Steps to edit permissions

1. In Salesforce Setup, select **Users**.
2. Select **Permission Sets**.
3. Select one of the following prebuilt permission sets.
   - If you and users of your Salesforce org have Einstein Analytics Platform Licenses:
     - **Einstein Analytics Platform Admin** enables all permissions required to administer the Analytics platform, including permissions to enable creating Analytics templated apps and Apps.
     - **Einstein Analytics Platform User** enables all permissions required to use the Analytics platform and Analytics templated apps and Apps
   - If you and users of your Salesforce org have Einstein Analytics Plus Licenses:
     - **Einstein Analytics Plus Admin** enables all permissions required to administer the Analytics platform and Einstein Discovery, including permissions to enable creating Analytics templated apps and Apps.
     - **Einstein Analytics Plus User** enables all permissions required to use the Analytics platform, Einstein Discovery, and Analytics templated apps and Apps
4. That opens the page for the selected permission set.
5. Click **Manage Assignments**. The next page shows the users already assigned the selected permission set.
6. Click **Add Assignments** to see all the users in your org.

7. Select one or more users by checking the box next to their name in the left-hand column.
8. Click **Assign**.

Select and Enable Analytics Features

Show or Hide Data in Thumbnails
1. From Setup, enter Analytics in the Quick Find box, then select Settings.
2. Select Show preview thumbnails for lenses and dashboards with row level-security enabled.
3. Click Save.

Enable or Disable Downloading from Analytics

1. From Setup, enter Permission Sets in the Quick Find box, then select Permission Sets.
2. Select the Analytics permission set where you want to add the permission, and then select System Permissions.
3. Click Edit, then select Download Analytics Data, if it's not already selected.
4. Click Save.

Enable Annotations on Dashboard Widgets
1. From Setup, enter Feed Tracking in the Quick Find box, then select Feed
2. Tracking.
3. Select Analytics Asset.
4. Select Enable Feed Tracking.
5. Click Save.

Enable the Analytics REST API
1. From Setup, enter Analytics in the Quick Find box, then select Settings.
2. Select Grant all users access to Wave API.
3. Click Save.

Monitor and Manage your Dataset
Duration: 10:00

Monitor Dataflows

The Dataflows subtab on the Monitor tab shows the status, start time, and duration of the last 10 dataflow jobs and retains the last 7 days of history. To help you troubleshoot a failed job, you can view error messages about the job, view the run-time details about every transformation that is processed, and download error logs where available.
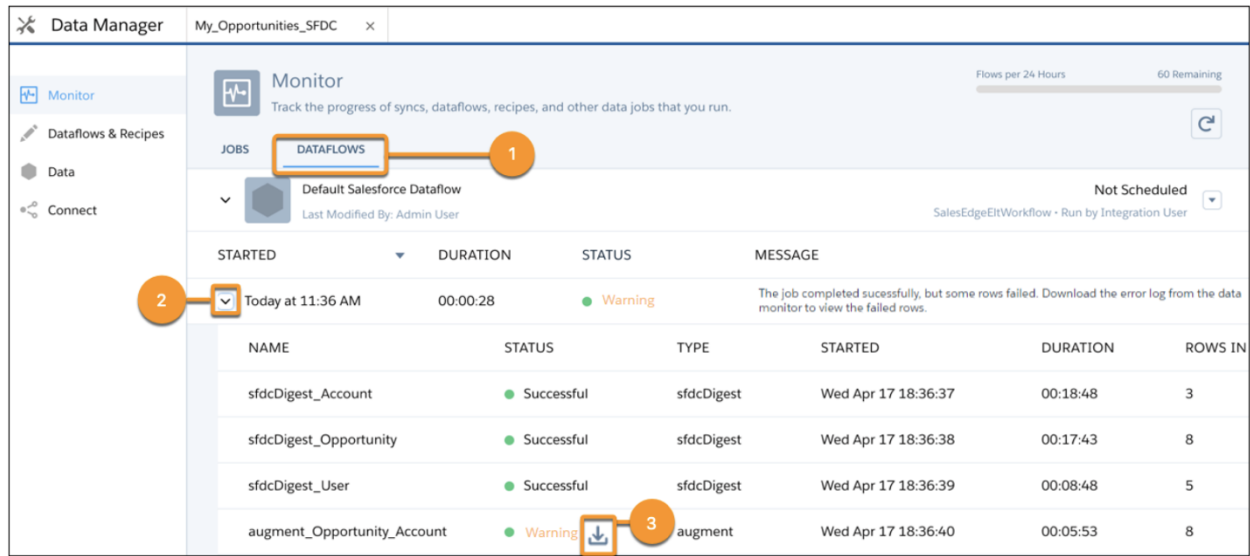
1. In Analytics, click the gear icon ( ) and then click **Data Manager**.
2. The data manager opens on the Monitor tab.
3. Click the **Dataflows** subtab (1).

4. Click to see the latest status of a job.

STATUS	DESCRIPTION
Running	The job is running.
Failed	The job failed.
Successful	The job completed successfully.
Warning	The job completed successfully, but some rows failed.

1. Each job can have one of these statuses.If the dataflow job fails, expand the job node (2) and review the run-time details for every transformation that was processed.
1. If an error log is available for a node, click the download log button (3) to download a CSV file containing the failed rows.
1. If there's a problem with the dataflow logic, edit the dataflow and then run it again.

NOTE
Error logs display the data from rows that have failed to load. To maintain data security and prevent unauthorized access to this data, only the dataflow owner or users with the View All Data permission can download an error log.

You can have up to 60 dataflow runs in a rolling 24-hour period. Dataflow and recipe runs that take less than 2 minutes to run, and data sync, don't count toward this limit. To track your current usage, use the flow indicator at the top of the Monitor tab.

Schedule Dataflow

1. In Analytics, click the gear icon ( ) and then click **Data Manager**.
2. The data manager opens on the Monitor tab, with the Jobs tab open by default.
3. Click the Dataflows & Recipes tab.
4. On the right of the dataflow, click and select **Schedule**.
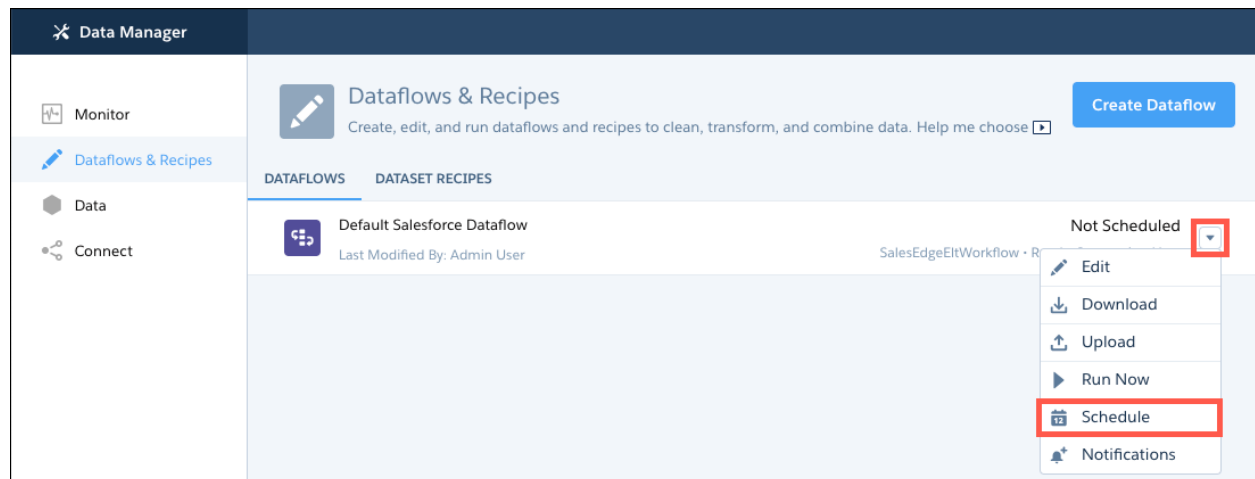
1. The scheduler appears.
1. Under Schedule Mode, select the **Time-based** or **Event-based**.
1. Choose the time-based mode to schedule the dataflow to run at a specified time. If you select this mode, continue with these steps to set the time information.
1. Choose the event-based mode to run the dataflow after the data sync finishes for all Salesforce objects in the dataflow. For example, if the dataflow pulls data from Accounts and Opportunities, both objects must sync before the dataflow can run. If you select this mode, click **Save** to complete the setup.
1. If you selected the time-based mode, from the **Schedule by** picklist, select the time frame that you want to schedule by and complete the other settings.
1. **By Hour**

1.   Run the dataflow at specified hourly intervals, starting at the time you select on the selected days.

1.

1.   NOTE To stop the dataflow from running after a certain time each day, select **Stop queuing at a specific time**. For example, set a job to start at 8:00 am, run every hour, and stop after 6:00 pm to restrict runs to just office hours.

1.   **By Week**

1.   Run the dataflow at the time you select on the selected days.

1. Click **Save**.

To remove a dataflow schedule, select Unschedule from the dataflow's menu.

Bring in Your Data

Duration: 20:00

Analytics Data Integration Basics

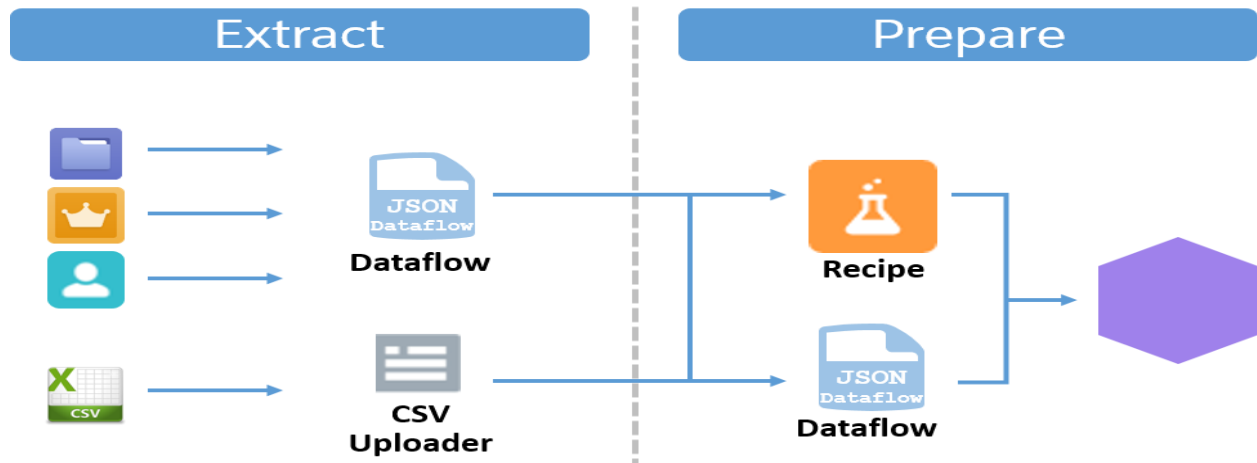## Entities in Analytics

What Is a Dataset?

Think of a dataset as a box of data. A dataset can contain data from a single Salesforce object, such as opportunities. Or it can contain data combined from different objects, such as opportunities, accounts, and users, and data from external sources, such as financial data. You can also create a dataset by combining data from other datasets, each in turn containing data from multiple sources and dataset
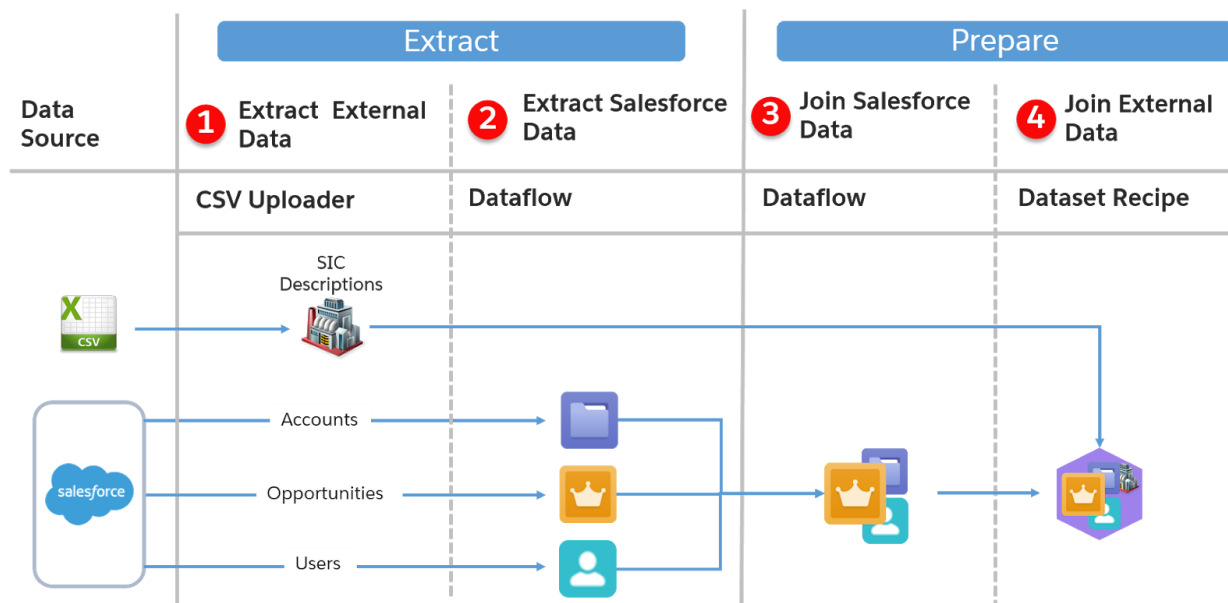
How Do You Create a Dataset?

There are two main stages in creating a dataset: extraction and preparation. Extraction, is simply the process of bringing data into Analytics. Preparation involves getting that data into a form that's meaningful to the people exploring it. To compare the process to cooking, extraction is taking the ingredients from your cupboards, and preparation is putting them together to make, say, a stew.
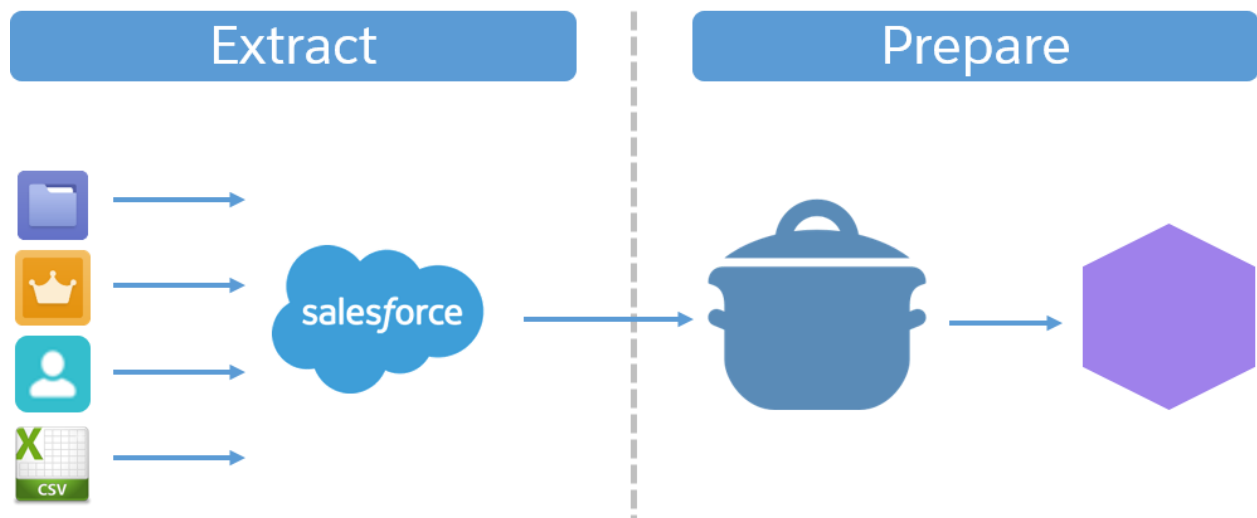
There are various ways you can extract data into Analytics. You can bring in external data through a CSV file using the CSV uploader, or bring it in using connectors or the Analytics API. For Salesforce data, you can use a powerful tool called the dataflow. For preparation, you can also use the dataflow—or you can get cooking with the dataset recipe tool.

Complete Data Integration Overview

Data Source	Extract		Prepare	
	1 Extract External Data	**2** Extract Salesforce Data	**3** Join Salesforce Data	**4** Join External Data
	CSV Uploader	Dataflow	Dataflow	Dataset Recipe

Upload Your CSV Data

1. If you're not already logged in, log in to the Developer Edition org that you signed up for in the previous unit.

2. Go to Analytics by first clicking the app launcher (⠿).
3. Then click the **Analytics Studio** tile.
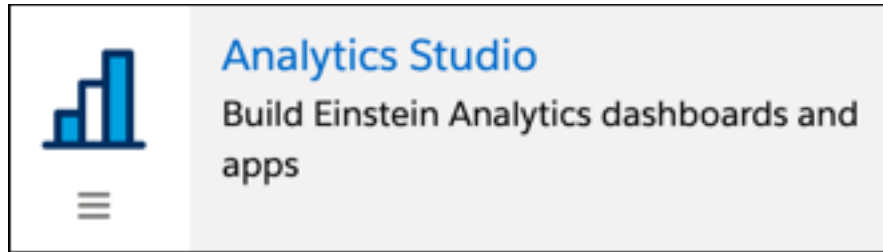
4.

5. In the Analytics Studio, click **Create** and select **App**.
6. You're creating an empty app, so click **Create Blank App**.
7. In the **Name your app** field, enter Sales Performance Datasets.
8. Click **Create**.
9. At the top of the app page, click **Create** and select **Dataset**.
10. Click **CSV File**.
11. Click **Select a file or drag it here**.
12. Navigate to the CSV file that you downloaded earlier, select it, and then click **Open**.
13. Click **Next**.
14. Analytics uses the file name as the name for the new dataset, and selects the app.
15. In the **Dataset Name** field, change the name to SIC Descriptions.
16. Click **Next**.
17. The Edit Field Attributes screen appears, giving you a preview of the data you're about to extract to Analytics.

Dataset View

The icon in each header tells you the field type:

- Measure

- Dimension ![A icon]

- Date ![calendar icon]

Extract Salesforce Data into Analytics

Dataflow

The dataflow is a set of instructions in JavaScript Object Notation (JSON) that runs to extract data and create datasets. These instructions specify which objects and fields you want to extract data from and the names of the datasets you want to create. A dataflow is not single use. You can use it to create lots of datasets from lots of different objects at the same time. You can also schedule it to run regularly to keep the datasets up to date. Since there's a chance the dataflow is already in use, it's a good idea to make a backup before you add new instructions

To Download JSON File of Dataflow

1. In Analytics, click the gear icon (![gear icon]) and then click Data Manager.
2. The data manager opens in a new browser tab.
3. In the data manager, click the Dataflows & Recipes tab.

4. On the right of the Default Salesforce Dataflow, click ![dropdown] and select Downlo
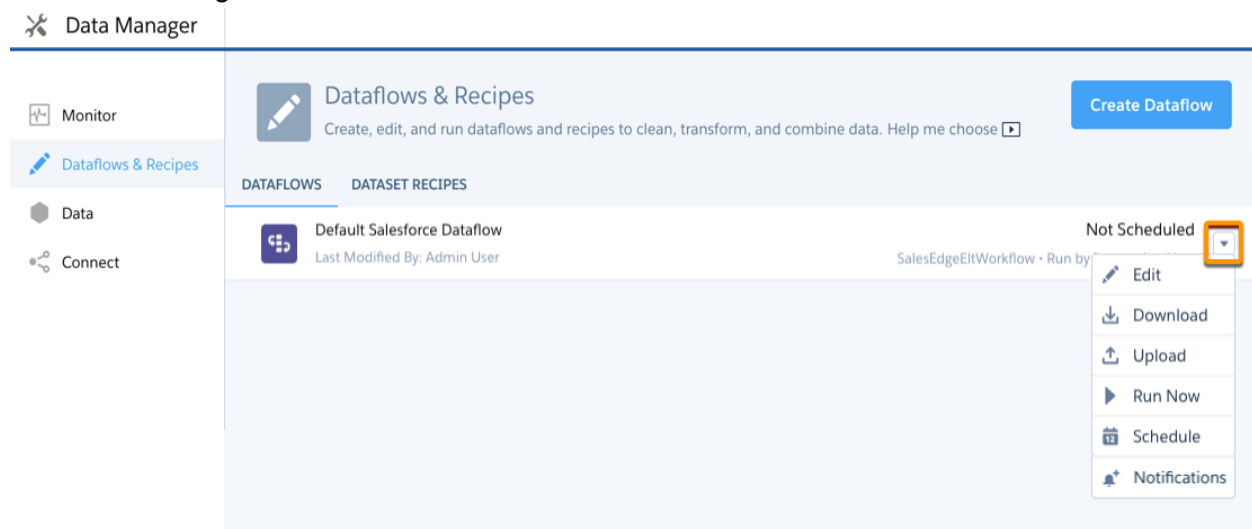
1. Save the JSON file locally and keep it as a backup of your existing dataflow. To go back to this version of the dataflow later, repeat these steps and choose Upload in step 3.

1. Click the gear icon (![gear icon]) and then click Analytics Studio.

Get to Know Datasets

Duration: 3:00

Date: A date can be represented as a day, month, year, and, optionally, time. You can group, filter, and perform math on dates.

Dimension: A dimension is a qualitative value that usually contains categorical data, such as Product Category, Lead Status, and Case Subject.
Dimensions are handy for grouping and filtering your data. Unlike measures, you can't perform math on dimensions. To increase query performance, Analytics indexes all dimension fields in datasets.

Measure: A measure is a quantitative value that contains numerical data like revenue and exchange rate. You can do the math on measures, such as calculating the total revenue and minimum exchange rate.

Sync Local and External Data with Analytics Connectors
Duration: 20:00
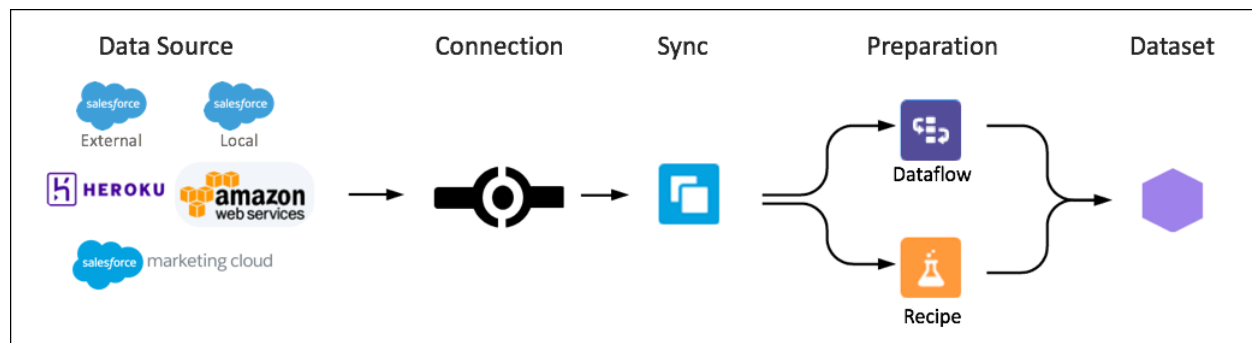
Analytics provides an internal Salesforce connector for data in your local org, and a range of remote connectors for data in apps, data warehouses, and database services. Let's look at these connectors and how an example company, Blue Sky Solar, uses some of them. Blue Sky Solar sells and installs solar panels in California and Arizona.

Salesforce Local Connector
- The Salesforce Local Connector is the default connector.
- If you're currently syncing objects in your local Salesforce org, you're already using this connector.

Steps to Connect
1. Create a new Dataset

New dataset

Choose a source for your data.

CSV File
Upload data directly into a dataset.

Salesforce Data
Create a dataset from related Salesforce objects using the dataset builder and dataflow.

External Data
Connect to data in supported enterprise applications, data warehouses, and database services.

Your Datasets
Prepare and combine data in datasets using a recipe and create a dataset with the results.

1.   Select Salesforce Data

New Dataset

Name the dataset and select a dataflow to add it to. The dataflow creates the dataset and keeps it updated. Learn More

Dataset Name

Add to existing dataflow

Select dataflow...

Add to new dataflow

Enter name for new dataflow...

Next

1.   We can select the object to be added to existing Dataflow or new Dataflow and select Next

1.   We can see below that we can integrate salesforce remote data with Analytics Studio using Dataset Builder
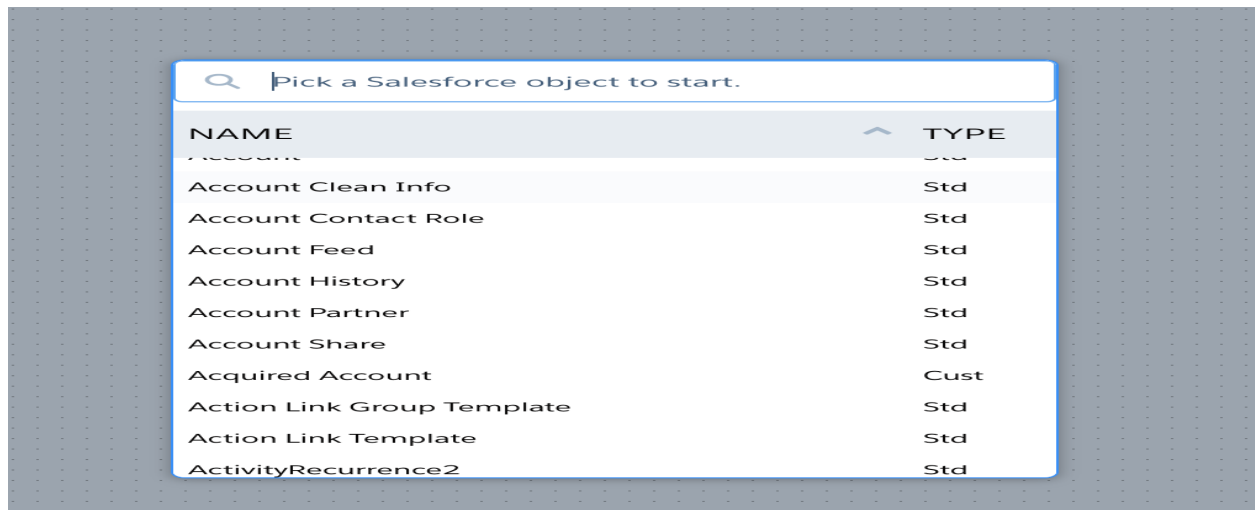
Data Warehouse Connectors
- Amazon Redshift
- Google BigQuery
- Microsoft Azure SQL Data Warehouse
- Snowflake Computing

Database Services Connectors
- AWS RDS Aurora MySQL
- AWS RDS Aurora PostgresSQL
- AWS RDS MariaDB
- AWS RDS MySQL
- AWS RDS PostgreSQL
- AWS RDS SQL Server
- Google Cloud Spanner
- Heroku Postgres
- Microsoft Azure SQL Database
- SAP HANA

File-Based Data Source Connectors
- Amazon S3

Website Analytics Connectors
- Google Analytics

Link for Video Tutorial to connect to external data Connecting to External Data

Transformation in Data flow

Duration: 30:00

- **append Transformation**
  The append transformation combines rows from multiple datasets into a single dataset.

- **augment Transformation**
  The augment transformation augments an input dataset by combining columns from another related dataset. The resulting, augmented dataset enables queries across both related input datasets. For example, you can augment the Account dataset with the User dataset to enable a query to return account records and the full names of the account owners.

- **computeExpression Transformation**
  The computeExpression transformation enables you to add derived fields to a dataset. The values for derived fields aren't extracted from the input data source. Instead, Analytics generates the values using a SAQL expression, which can be based on one or more fields from the input data or other derived fields. For example, you can use an expression to assign a value to a field, concatenate text fields, or perform mathematical calculations on numeric fields.

- **computeRelative Transformation**
  You can use the computeRelative transformation to analyze trends in your data by adding derived fields to a dataset based on values in other rows. For example, to analyze sales pipeline trends, create derived fields that calculate the number of days an opportunity remains in each stage. You can also calculate the changes to the opportunity amount throughout the stages of the opportunity.

- **delta Transformation**
  The delta transformation calculates changes in the value of a measure column in a dataset over a period of time. The delta transformation generates an output column in the dataset to store the delta for each record. Create deltas to make it easier for business analysts to include them in queries.

- **digest Transformation**
  The digest transformation extracts synced connected data in a dataflow. Use it to extract data synced from your local Salesforce org, or data synced through an external connection.

- **dim2mea Transformation**
  The dim2mea transformation creates a new measure based on a dimension. The transformation adds the new measure column to the dataset. The transformation also

preserves the dimension to ensure that existing lenses and dashboards don't break if they use the dimension.

- **edgemart Transformation**
  The edgemart transformation gives the dataflow access to an existing, registered dataset, which can contain Salesforce data, external data, or a combination of the two. Use this transformation to reference a dataset so that its data can be used in subsequent transformations in the dataflow. You can use this transformation and the augment transformation together to join an existing dataset with a new dataset.

- **export Transformation**
  The export transformation creates a data file and a schema file from data in a specified source node in your dataflow. After the dataflow runs, Einstein Discovery users can access these files through the public API.

- **filter Transformation**
  The filter transformation removes records from an existing dataset. You define a filter condition that specifies which records to retain in the dataset.

- **flatten Transformation**
  The flatten transformation flattens hierarchical data. For example, you can flatten the Salesforce role hierarchy to implement row-level security on a dataset based on the role hierarchy.

- **sfdcDigest Transformation**
  The sfdcDigest transformation generates a dataset based on data that it extracts from a Salesforce object. You specify the Salesforce object and fields from which to extract data. You might choose to exclude particular fields that contain sensitive information or that aren't relevant for analysis.

- **sfdcRegister Transformation**
  The sfdcRegister transformation registers a dataset to make it available for queries. Users cannot view or run queries against unregistered datasets.

- **sliceDataset Transformation**
  The sliceDataset transformation removes fields from a dataset in your dataflow, leaving you with a subset of fields for use in a new dataset or in other transformations. This allows you to create multiple datasets, each with different sets of fields from a single dataset.

- **update Transformation**
  The update transformation updates the specified field values in an existing dataset

based on data from another dataset, which we'll call the lookup dataset. The transformation looks up the new values from corresponding fields in the lookup dataset. The transformation stores the results in a new dataset.

- **Overriding Metadata Generated by a Transformation**
  Optionally, you can override the metadata that is generated by a transformation. You can override object and field attributes. For example, you can change a field name that is extracted from a Salesforce object so that it appears differently in the dataset. To override the metadata, add the overrides to the Schema section of the transformation in the dataflow definition file.

Optimize Your Dataflows with Data Sync

Duration: 10:00

data sync decouples the extract of Salesforce data from your dataflows, and sync this data in Analytics on a separate schedule. By scheduling sync ahead of time, your dataflow has less to do and runs faster. To lighten the load even more, Analytics syncs Salesforce data incrementally by default, meaning that only data that's changed gets synced.

Without data sync, a dataflow performs a separate extract each time it needs data from a Salesforce object. Let's look at an example. Imagine your organization has three dataflows, extracting data from Salesforce objects as follows:

	ACCOUNTS	CONTACTS	OPPORTUNITIES	CAMPAIGNS	LEADS	CASES	USERS
Default Dataflow	■	■	■			■	■
Sales Analytics App Dataflow	■	■	■	■	■	■	■
Service Analytics App Dataflow	■	■					■

Every time these dataflows run, they have to extract all this Salesforce data. And the more data there is, the longer the dataflow takes to run. What's more, the dataflows are duplicating effort by performing separate extracts from the same object. For example, Accounts data is extracted by all three dataflows.

With data sync, all of these extracts are performed as a separate process, which you can schedule to take place before your dataflows run.

	ACCOUNTS	CONTACTS	OPPORTUNITIES	CAMPAIGNS	LEADS	CASES	USERS
Data Sync							

This synced data is then available to all your dataflows, which run faster because they no longer have to extract any data—just load and transform.

Understanding How Data Sync works

Enable Data Sync and Connections

To enable data sync and connections, perform the following steps.

1. From Setup, enter Analytics in the Quick Find box, then select Settings.
2. Select Enable Data Sync and Connections and Enable Analytics Templates.
3. Save your changes.

Disconnect an Object from Sync

To disconnect an object from sync, perform the following steps.

1. In Analytics, click the gear icon (   ) and then click **Data Manager** to open the data manager.
2. In the data manager, click the **Connect** tab.
3. The Connect tab displays a list of objects enabled for sync. Salesforce objects from your local org are listed first.
4. Click the actions button (   ) to the right of the object and then select **Disconnect Object**.
5. A warning message appears.
6. Read the warning message carefully and then click **Disconnect**.

Recipes

Duration: 15:00

A dataset recipe is simply a saved set of transformations, or steps, that you want to perform on a specific source dataset or connected data.

Use a recipe to perform transformations like combining data from multiple datasets or connected objects, bucketing the data, adding formula fields, and cleansing the data by changing field values. You can remove fields and filter rows that you don't need.

When you run a recipe, it applies the transformations and outputs the results to a new target dataset. You can schedule a recipe to run on a recurring basis to keep your target dataset up to date.
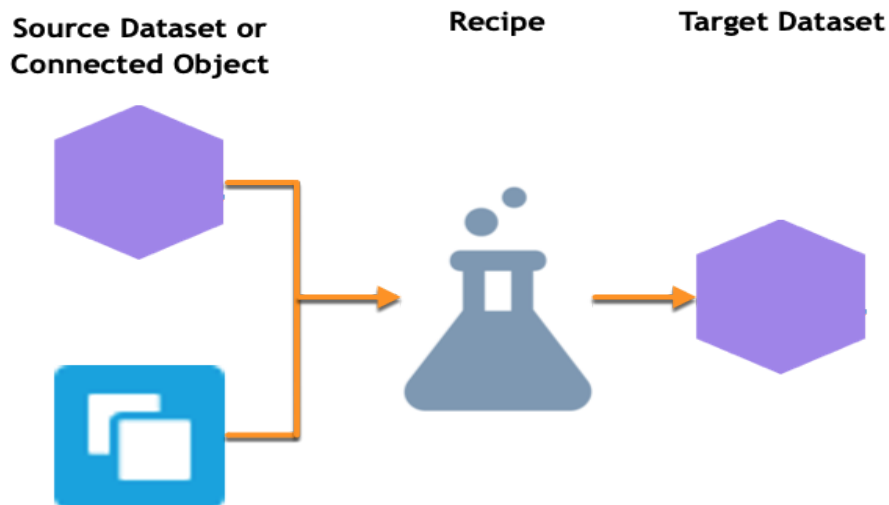
The target dataset can then be used as a standalone dataset for exploration or dashboard creation, or used as input to your dataflows or other recipes.

Here's what you can do with your source data in a recipe:
- Add fields from another dataset or connected object.
- Remove fields.
- Filter rows.
- Add bucket fields.
- Add formula fields.
- Transform field values.

Choosing between Dataflow and Recipe

Start a Dataset Recipe

1. In Analytics, click the gear icon ( ) and then click **Data Manager** to open the data manager.
2. The data manager opens in a new browser tab.
3. In the data manager, click the Dataflows & Recipes tab.
4. In the Dataflows & Recipes tab, click the Dataset Recipes subtab.
5. The Dataset Recipes subtab displays a list of your existing recipes.

6. Click **Create Recipe**.
7. From the list of datasets, click the one that contains the data you want to prepare. Alternatively, if you have enabled data sync, you can click the **CONNECTED DATA** tab to select a connected object instead.

Clean and Prepare Data Intelligently with Column Profiles and Smart Suggestions

The column profile gives you key insights into the quality of your data and suggests additional transformations to help you clean and prepare it. This profile is especially useful when you are combining data from different sources, where inconsistencies are often introduced.

Let's look at an example of how you can use the profile to review the quality of data in a set of leads. Select the column you want to analyze to see its profile on the right of the recipe editor.

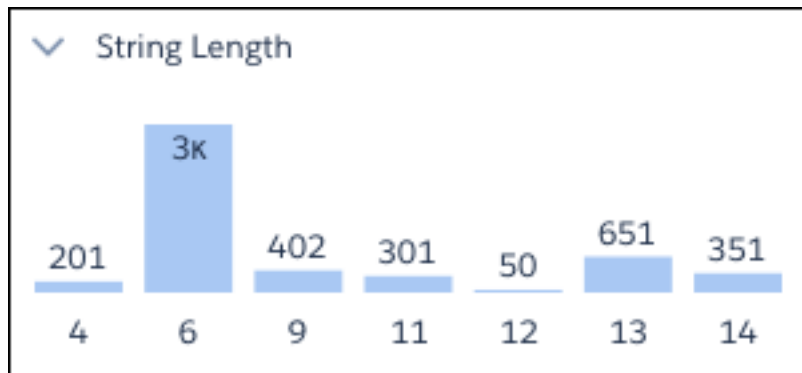To open the column profile if it isn't visible, click or select Show Profile from the column's menu.

Sample Size

This section shows the number of rows used to generate the column profile data. To use a larger sample, click .
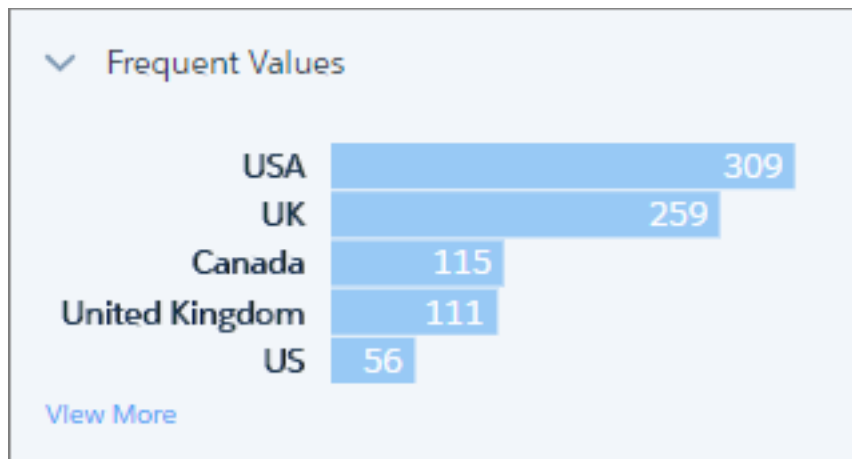
Valid Values

This section gives you an idea of how the column is populated. A high proportion of null values can indicate lower quality data of limited usefulness in lenses and dashboards.

Frequent Values

This section shows the count for each of the top five values in the column. To see the count for all values in the sample, click **View More**. This information gives you a quick insight into the variation in your data. This example shows different values for the same country, which can impact grouping and filtering in a lens or dashboard.

For dimension and date columns, click one or more bars to see suggested transformations for those values in the Suggestions bar. In this example, if you select the US value, Analytics suggests replacing it or filtering it out.
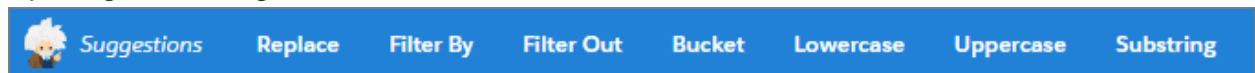
String Length (Dimension Columns Only)
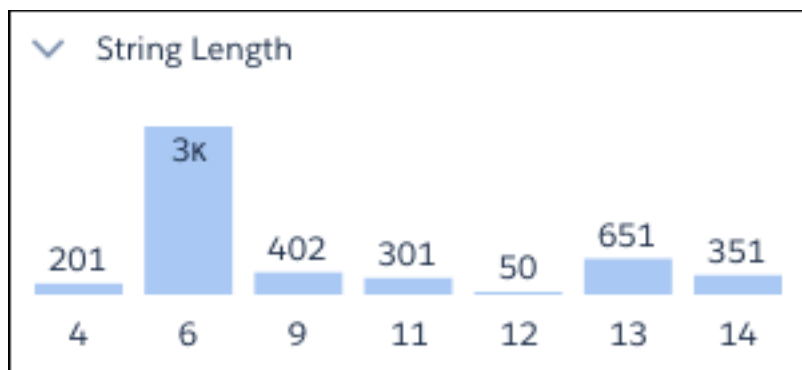
This section gives you a breakdown of the number of characters in dimension values. A wider variation than expected can indicate inconsistency issues with data from mixed sources.

Add a Filter in a Dataset Recipe

To add a filter to a dataset recipe, follow these steps.

1. On the column header of the field you want to filter by, click ⏷ , and then select **Filter**.
2. Filter settings appear in the Add Transformation panel below the preview data.

1. In the first field, confirm the field to filter by, or click to select a different field from the list. You can select from any field in the recipe, even if it has been hidden.
1. In the second field, confirm the operator, or click to select a different one from the list.
1. In the third field, enter the filter value.

1. To enter another condition, click ⊞ and repeat steps 2–4.
1. When you have added all your conditions, click **Add**.
1. The filter appears as a step in the left pane of the recipe.

Add Rows in a Dataset Recipe with Append

1. On the dataset recipe page, click the Append Data button (⊞).
2. Select either the DATASETS or CONNECTED DATA tab, and click the dataset or object that you want to add. In our example, you would select the Opportunity dataset from the CanadaOrg connection.

Select the dataset to append

REPLICATED

NAME	CONNECTION NAME	LAST RUN
Opportunity	SFDC_LOCAL	Apr 11, 2018 at 11:48 AM
User	SFDC_LOCAL	Apr 11, 2018 at 11:48 AM
Account	SFDC_LOCAL	Apr 11, 2018 at 11:48 AM
Opportunity	CanadaOrg	Apr 11, 2018 at 11:48 AM

1.   On the map fields dialog, review how Analytics has mapped the recipe fields in the first column to fields from the data to append in the second column. Remember, fields with the same name and type are mapped automatically.

Map fields to append from CA Sales

Å FIELD IN RECIPE	⬡ FIELD TO APPEND (138 Rows)	SAMPLE APPEND VALUE
A Billing Country	A Billing Country 🔍	Canada
A Currency	A Currency 🔍	CAD
A Lead Source	⊖ None 🔍	
A Account Name	A Account Name 🔍	Butler1...
📅 Close Date	📅 Close Date 🔍	7/7/17
⊘ Amount	⊖ None 🔍	

Show 3 hidden fields

Back   Continue

1.   NOTE By default, hidden fields in your recipe aren't mapped. To map hidden fields, click the **Show hidden fields** link and accept or change the mappings.

1.   To change a field mapping, click the field in the FIELD TO APPEND column, and select a field. You can only select from fields of the same type as the recipe field that you're mapping to.

1. When you finish mapping fields, click **Continue**.
1. On the confirmation message, click **Append**.

NOTE Rows that you append in a recipe do not appear in the recipe preview, but are added to the target dataset when you run the recipe.

When you append data, a new step is created in your recipe.

Demo AdventureWorks Data

Duration: 60:00

Dataset Link

Data Model

Create an APP named AdventureWorks and Load all the CSV's under this app

NOTE
1. Join operation can be performed only between dimensions. So make sure to convert your key fields to dimension while uploading data to create a dataset
2. ProductID, CategoryID, SubCategoryID, ModelID, SalesorderID, SalesorderDetailID, SpecialOfferID as Dimensions
3. Go to Setup > Enter Analytics in Search box> Setting and Select Enable Sync and Connection for you to establish remote connections and configure Dataflow

Part 1: Basic Join Between Two Dataset

1. First we **augment Product Sub Category and Product Category.**
2. Select **EdgeMart as Transformation objec**t to create a Dataset Object.
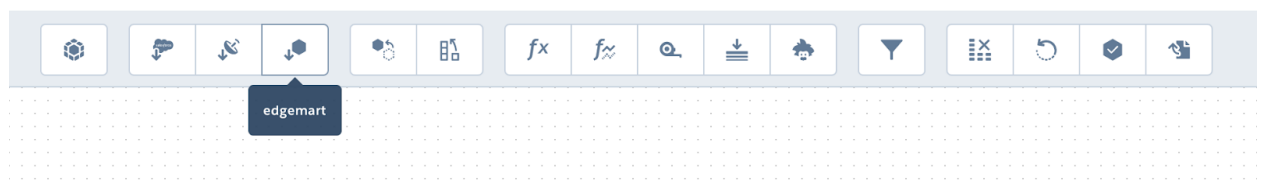
Category: edgemart ✕

ATTRIBUTES   OUTPUT FIELDS

Node Name

Category

Dataset

["ProductCategory"]

Cancel   Save

1. Enter **Category** as Node Name and select **ProductCategory** as Dataset
1. Similar Select Another Edgemart for **ProductSubCategory** and create one
1. Now use an **Augment Transformation** to join the two datasets. (Left Join by Default)

augment ✕

ATTRIBUTES   OUTPUT FIELDS

Node Name

Category_SubCategory

Left Source

["getSubCategory"]

Left Key

["ProductCategoryID"]

Relationship

Category

Right Source

["getCategory"]

Cancel   **Create**

1.   Here we need to enter a few attribute details,
1.   NodeName is the **connection** name

1. Left source specifies left table of the join
1. **Left key is the on which key the join should take place on.** Here ProductCategory and ProductSubCategory have relationship key as ProductCategoryID
1. **Relationship name is appended** to the fields selected from right table

SubCategory_Category: augment ✕

ATTRIBUTES   OUTPUT FIELDS

Relationship

Category

Right Source

["Category"]

Right Key

["ProductCategoryID"]

Right Fields

["Name"]

Operation

Look Up Multiple Values ⌄

Cancel   **Save**

1. We select Right table as Category and Right Key as ProductCategoryID
1. Right fields specify which field we want to select from right table. Here we select just name as other field is already available.

1. **Operation is multiple look up as its one to many relationship**
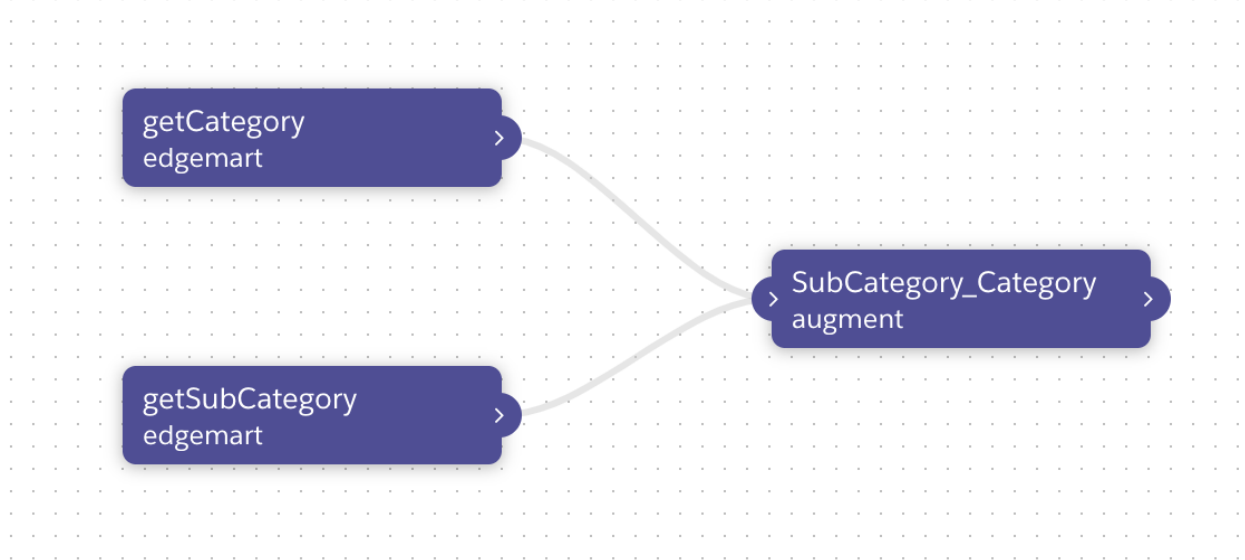
1. Next, we need to use a **SDFC Register to create a dataset** of this integration

sfdcRegister

×

ATTRIBUTES   **OUTPUT FIELDS**

Node Name

ProductSubCategory_Category

Source Node

["SubCategory_Category"]

Alias

ProductSubCategory_Category

Name

ProductSubCategory_Category

Sharing Source

[]

Cancel   **Create**

1.   Here we give node name that would be created as our dataset.

1.   Source is taken as our Augment Transformation that is SubCategory_Category

1.   Below is a working model for a simple join using augment transformation and creating a dataset
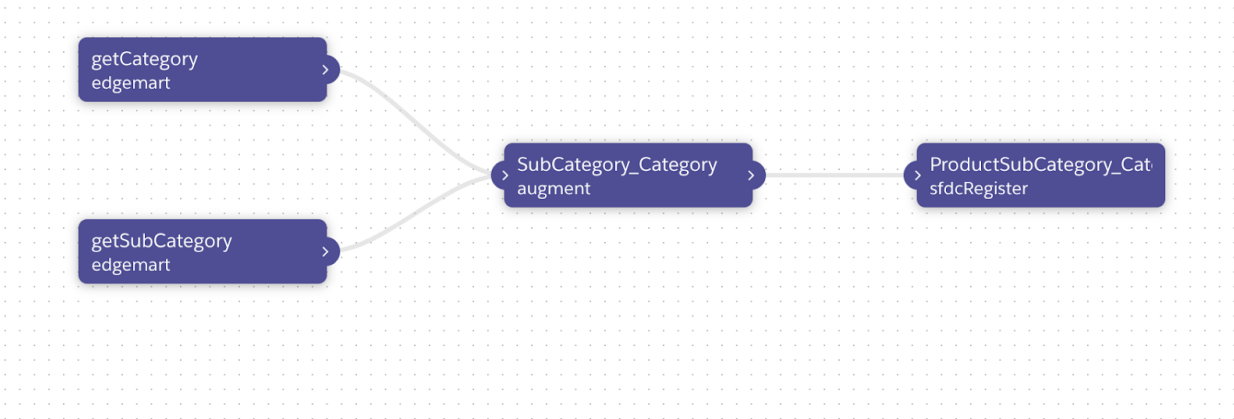
1.   Update Workflow builds the task and checks for errors. Once its built you can run the workflow to integrate data
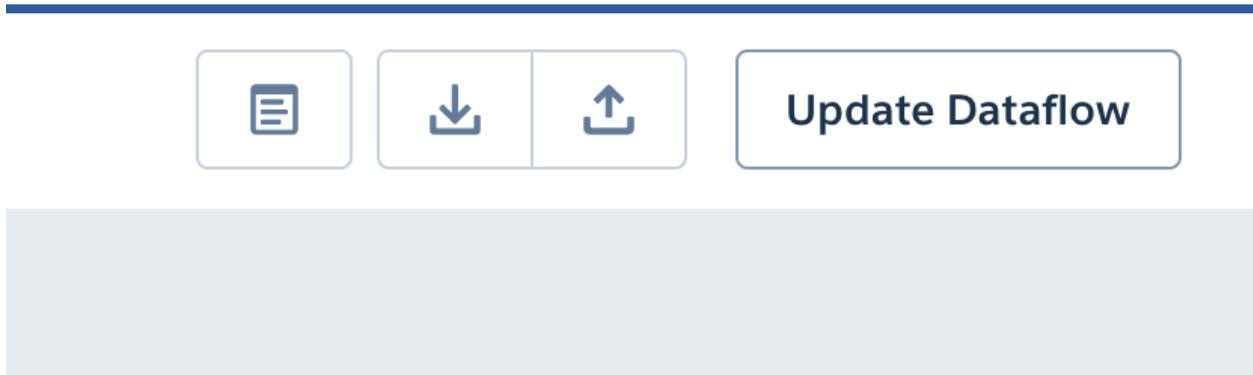
1.   Run DataFlow to begin the workflow

1.   After running you can monitor your task and see if its succesful from **Monitor**
1.   Once its successful you can see the dataset has been created in your data repository.

NAME	STATUS	TYPE	STARTED	DURATION	ROWS IN	ROWS OU
getCategory	● Successful	edgemart	Today at 10:07 PM	00:00:00		
getSubCategory	● Successful	edgemart	Today at 10:07 PM	00:00:00		
DropSharingRulesFrom-getCategory	● Successful	sliceDataset	Today at 10:07 PM	00:00:00	4	4
DropSharingRulesFrom-getSubCate...	● Successful	sliceDataset	Today at 10:07 PM	00:00:00	37	37
SubCategory_Category	● Successful	augment	Today at 10:07 PM	00:00:00	37	37
optimize-ProductSubCategory_Cate...	● Successful	optimizer	Today at 10:07 PM	00:00:00		
ProductSubCategory_Category	● Successful	sfdcRegist...	Today at 10:07 PM	00:00:01		

(tooltip: DropSharingRulesFrom-getSubCategory)

1.   This completes a basic join operation between two dataset Objects.

Part 2: Create DimProduct

1.   Now if we look at our data model. We need to merge these data sets to create the DimProduct Dataset
a.   Product
b.   Product Category
c.   Product SubCategory
d.   Product Model

1.   By using the same Augment Transformation in step 1, **we Augment, Product and Product Mode**l to the dataflow in step 1 to create **dimProduct**
1.   The keys to join on would be Model ID
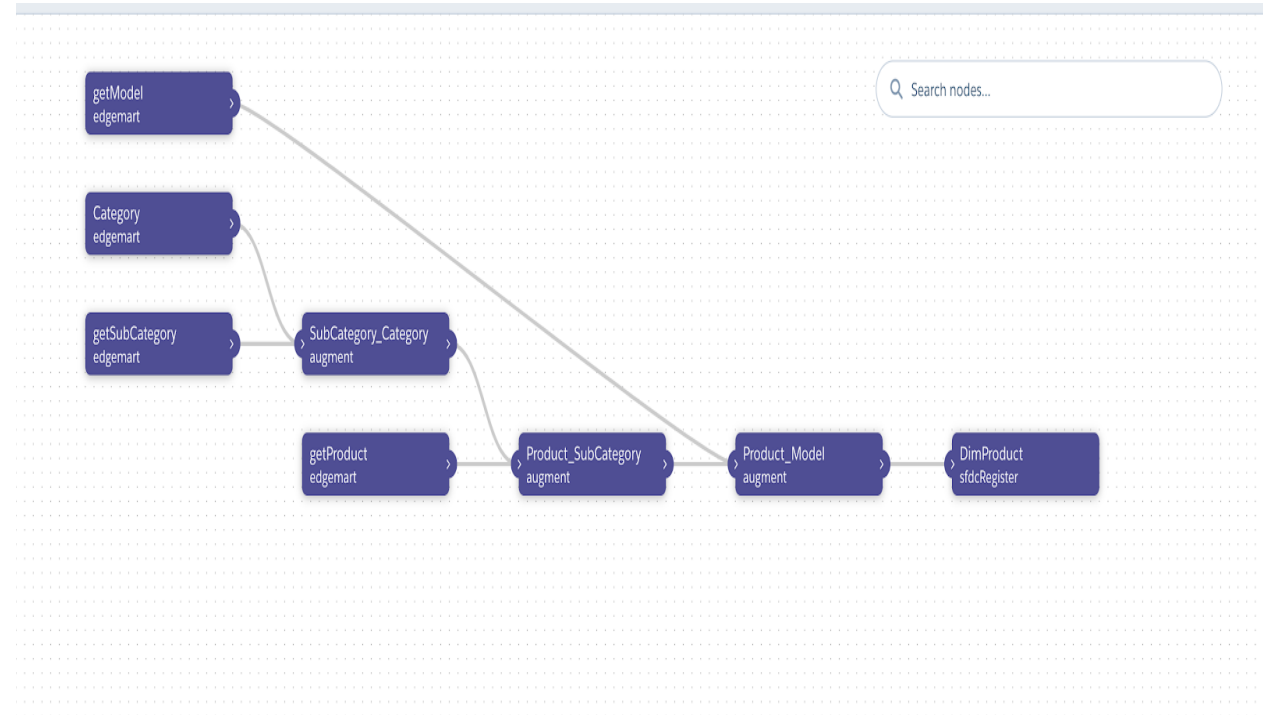
1.   Once we have created this dataflow, we can run the job to create a dataset containing dimProduct

NAME	STATUS	TYPE	STARTED	DURATION	ROWS IN	ROWS OU
Run AW_Product	● Successful	Dataflow	Today at 10:19 PM	00:00:32		
Category	● Successful	edgemart	Today at 10:19 PM	00:00:00		
getModel	● Successful	edgemart	Today at 10:19 PM	00:00:00		
getProduct	● Successful	edgemart	Today at 10:19 PM	00:00:00		
getSubCategory	● Successful	edgemart	Today at 10:19 PM	00:00:00		
DropSharingRulesFrom-Category	● Successful	sliceDataset	Today at 10:19 PM	00:00:00	4	4
DropSharingRulesFrom-getModel	● Successful	sliceDataset	Today at 10:19 PM	00:00:00	128	128
DropSharingRulesFrom-getProduct	● Successful	sliceDataset	Today at 10:19 PM	00:00:00	504	504
DropSharingRulesFrom-getSubCate...	● Successful	sliceDataset	Today at 10:19 PM	00:00:00	37	37
SubCategory_Category	● Successful	augment	Today at 10:19 PM	00:00:00	37	37

1.   The task ran successfully and **DimProduct Dataset has been created**

Part 3: Create DimSales

1. Now if we look at our data model. We need to merge these data sets to create the DimSales Dataset

a.   SalesOrderDetails
b.   SalesOrderHeader
c.   Customer
d.   Promotions (Special Offer)

   2. First create an **Augment Transformation between SalesOrderDetails and SalesOrderHeader as left and right sources respectively**
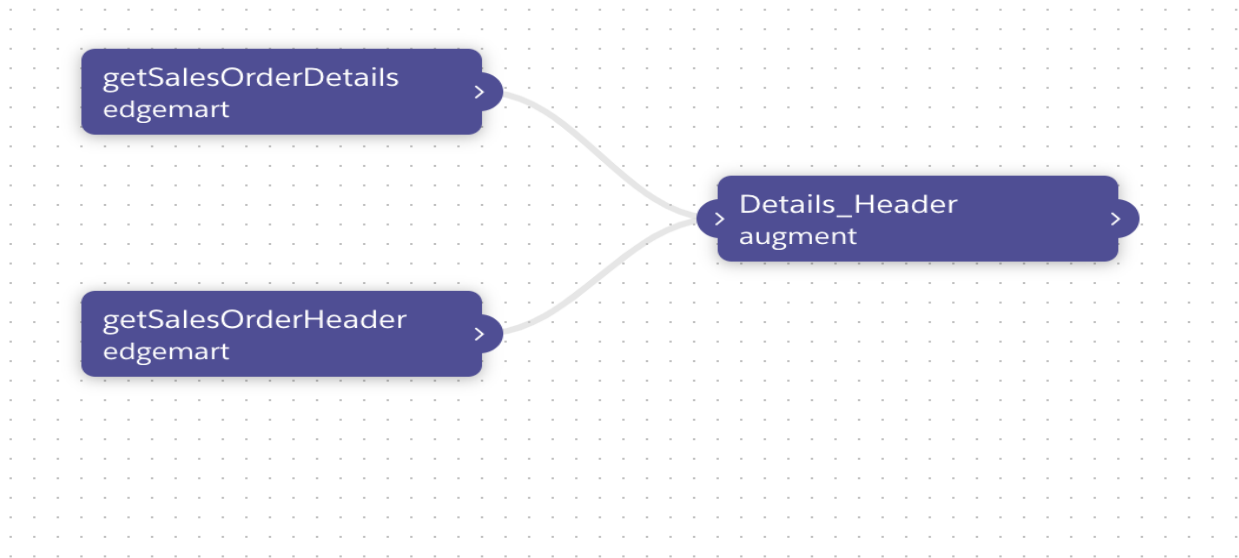
1.   Now create an **Augment with both Customer and Promotion** individually to create the whole task
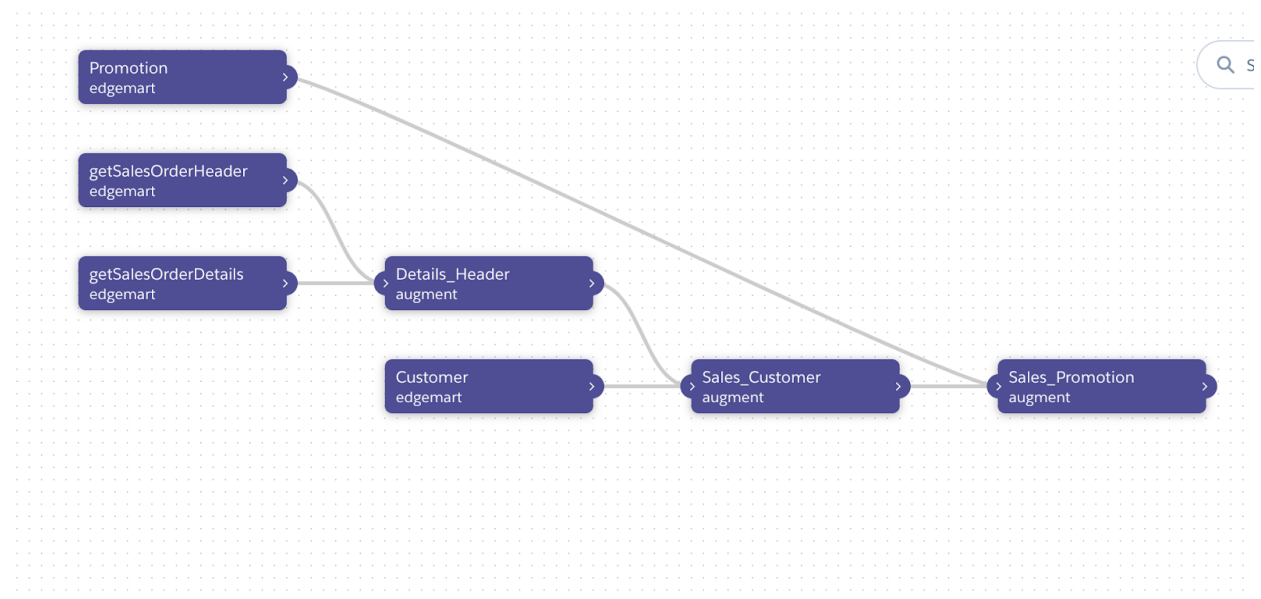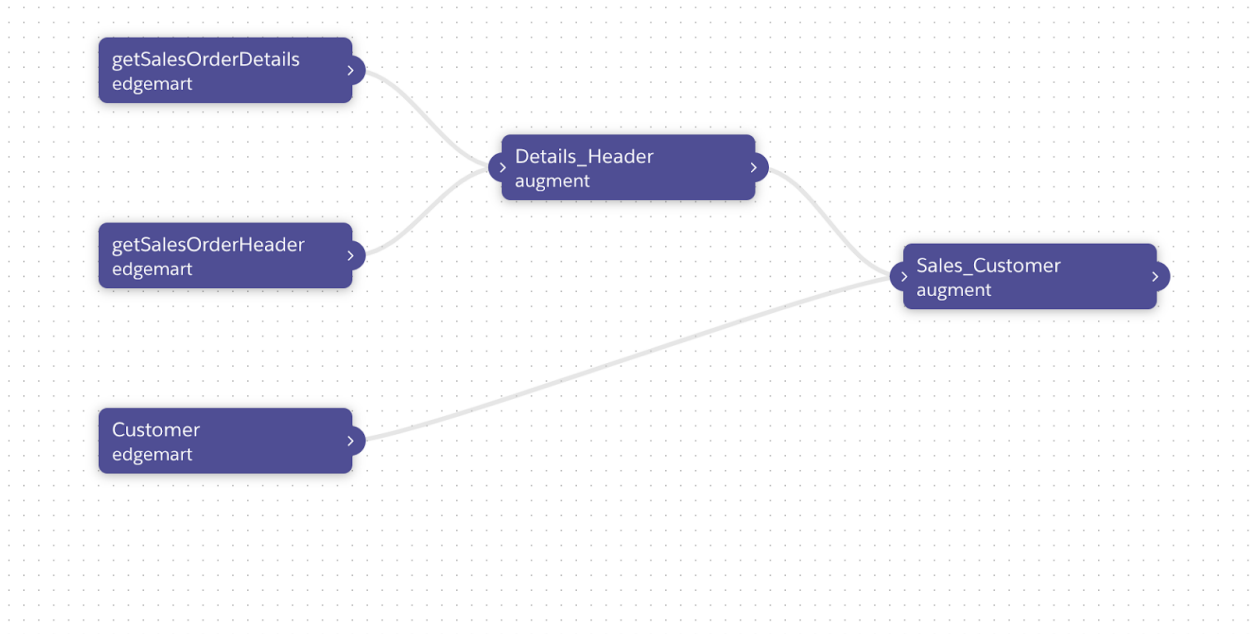
1.   Use SDFC Register to Create a DimSales Dataset
1.   Update and Run the Dataflow to create Dataset
1.   We can schedule the Dataflows in case of real time streaming of data.

Part 4: Augment Both Data Marts to Create AW Dataset

1. Now we have dimProduct and dimSales datasets, we augment this so that we have one single dataset which contains all required dimensions and measures.

2. Create a new Dataflow named as AW.
3. Select a Edgemart transformation with dimProduct Dataset and another edgemart with DimSales dataset
4. Use an augment transformation to join both the edgemarts on ProductID
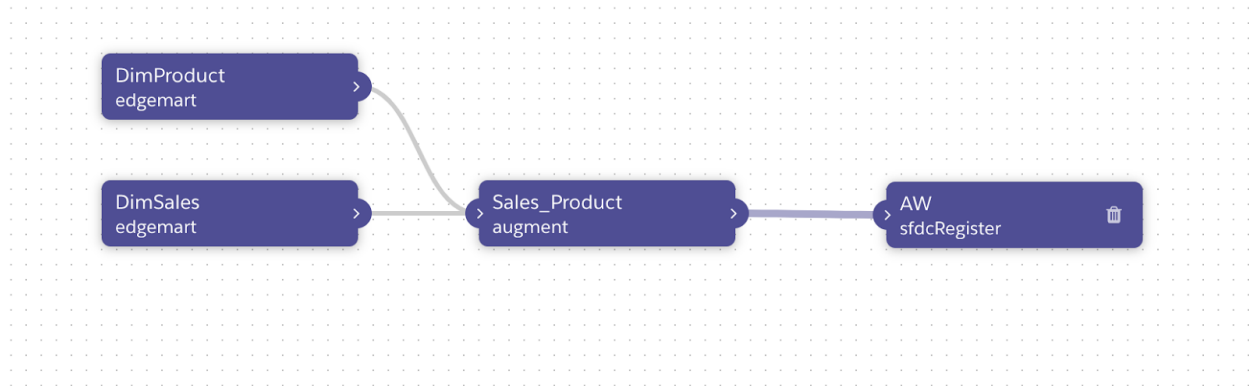
1. On completion of this task we have created a **single OLAP view of entities of our object model**

Part 5 : Creating a Lens

1. **Lens can be considered as one view point of our dataset.** By selecting different dimensions and measure values, we can create different lenses of our data and use these visualizations in our final dashboard.
2. To create a lens. **Navigate to your dataset and click on it to create a view of a new lens**
3. A dataset lens will always have a count of rows measure by default.
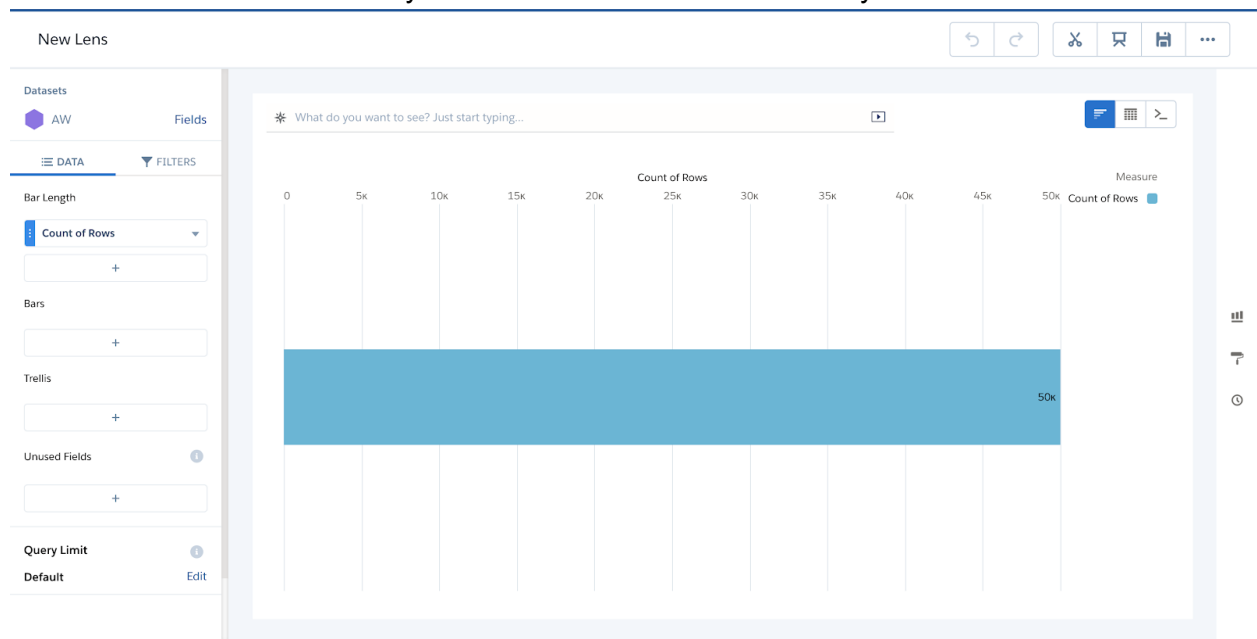
1. You can use a filter condition to select only rows that are required.

1. On the left we have bar length which basically takes input of only measures.
1. Below that we have a bar, this is where we can select our dimensions for visualization
1. **Trellis** lets us to perform a group by and **resulting summary statistics** are seperated and represented
1. The icons on the extreme right of bar are options for charts, formatting and maintains a change log of previous views on editing.
1. We can select Top field by using **Query Limit**

Examples of Lenses for our data

1. Pie chart for B2B vs B2C sales
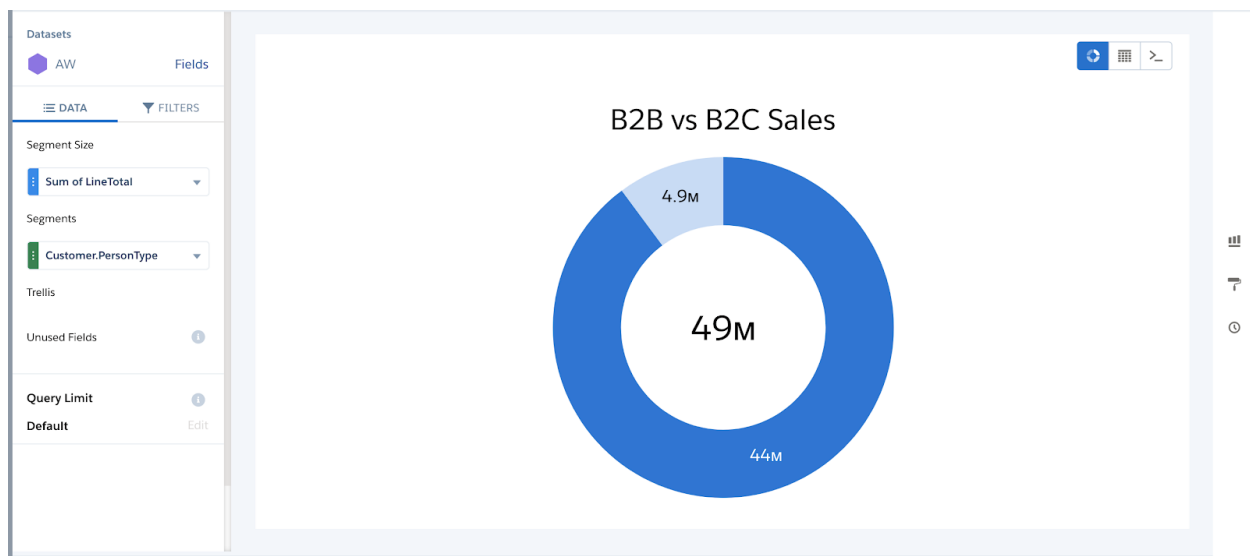
1. Waterfall charts Sales by Sub-Category
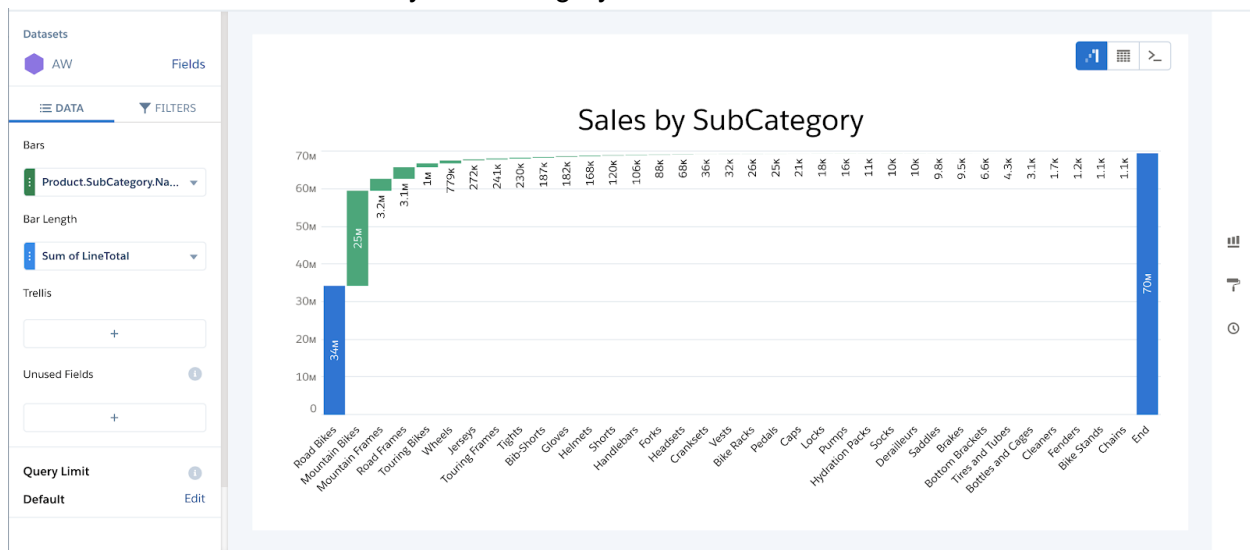
1.   Pyramid Chart Comparison for Highest Sales vs Sales Quantity
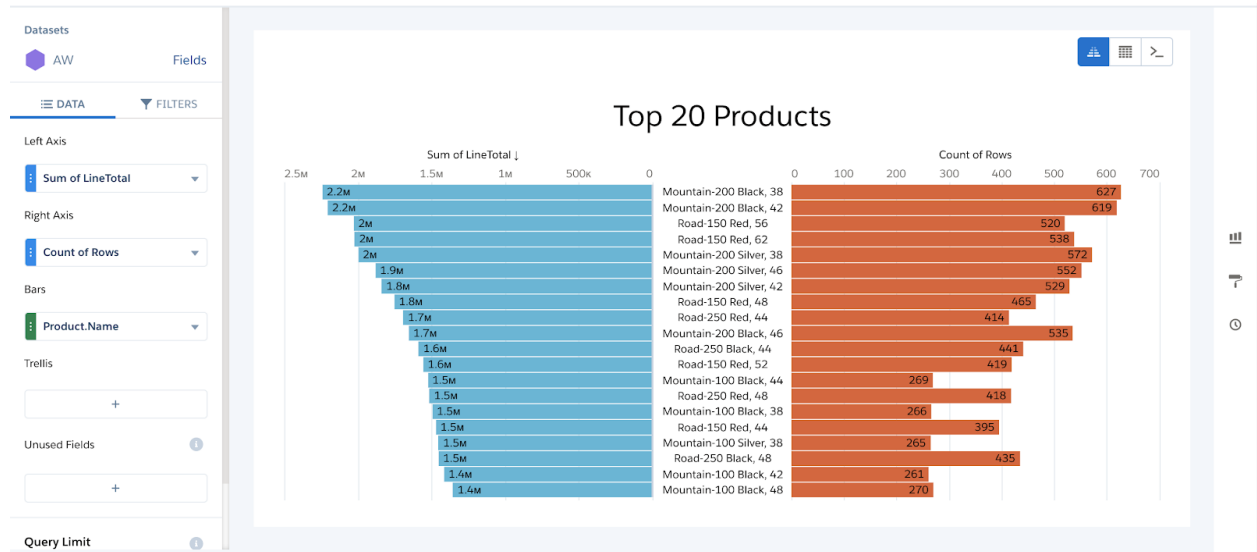
1.   A trellis chart to show Promotion Description grouped by Promotion Type
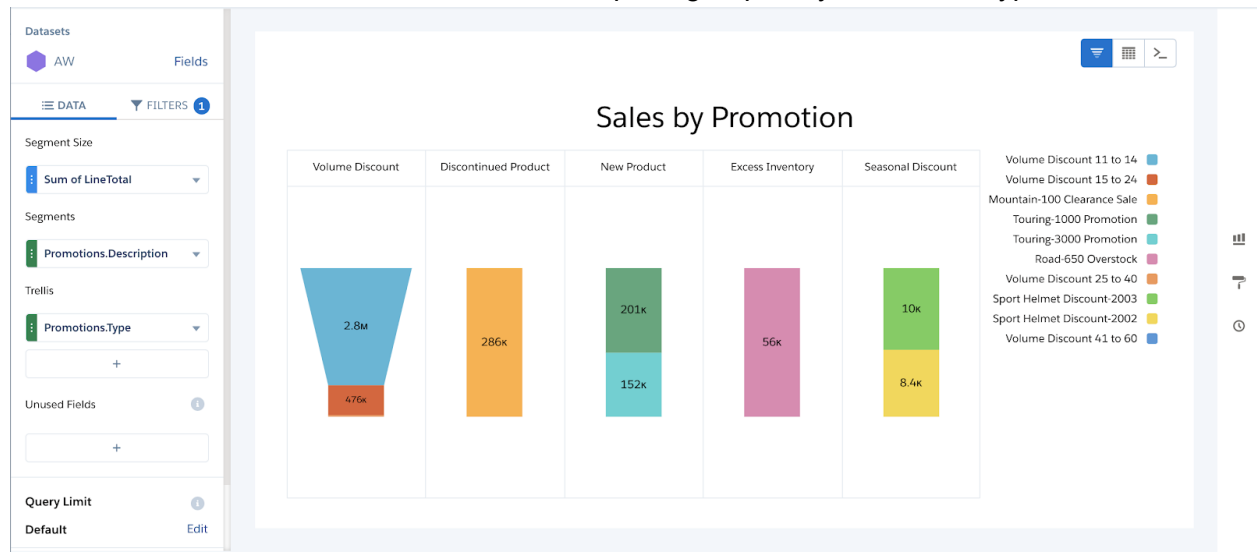
1.   A Rating Chart to show single measured values

While building these we could add this to clip designer for this to be attached as a **Step** to our Dashboard

Part 6: Creating a Dashboard

1. The last part would be to build an Analytics Dashboard.
2. Navigate to your app and under Create select Dashboard.
3. This gives you an option of creating dashboard from scratch or to use some pre exisitng templates. For our example we would be considering a blank dashboard. It would take you to the dashboard page which looks like below

1. The icons on the left are the widegets, which enable us to create the dashboards, some of the ones commonly used are filters, containers, navigation, toggle and text and numeric containers.
1. The steps that we would have attached using Clip would be present on right below Create Step. Similarly we can also directly create a lens from a dataset by selecting Create Step > Dataset and save it to attach to dashboard
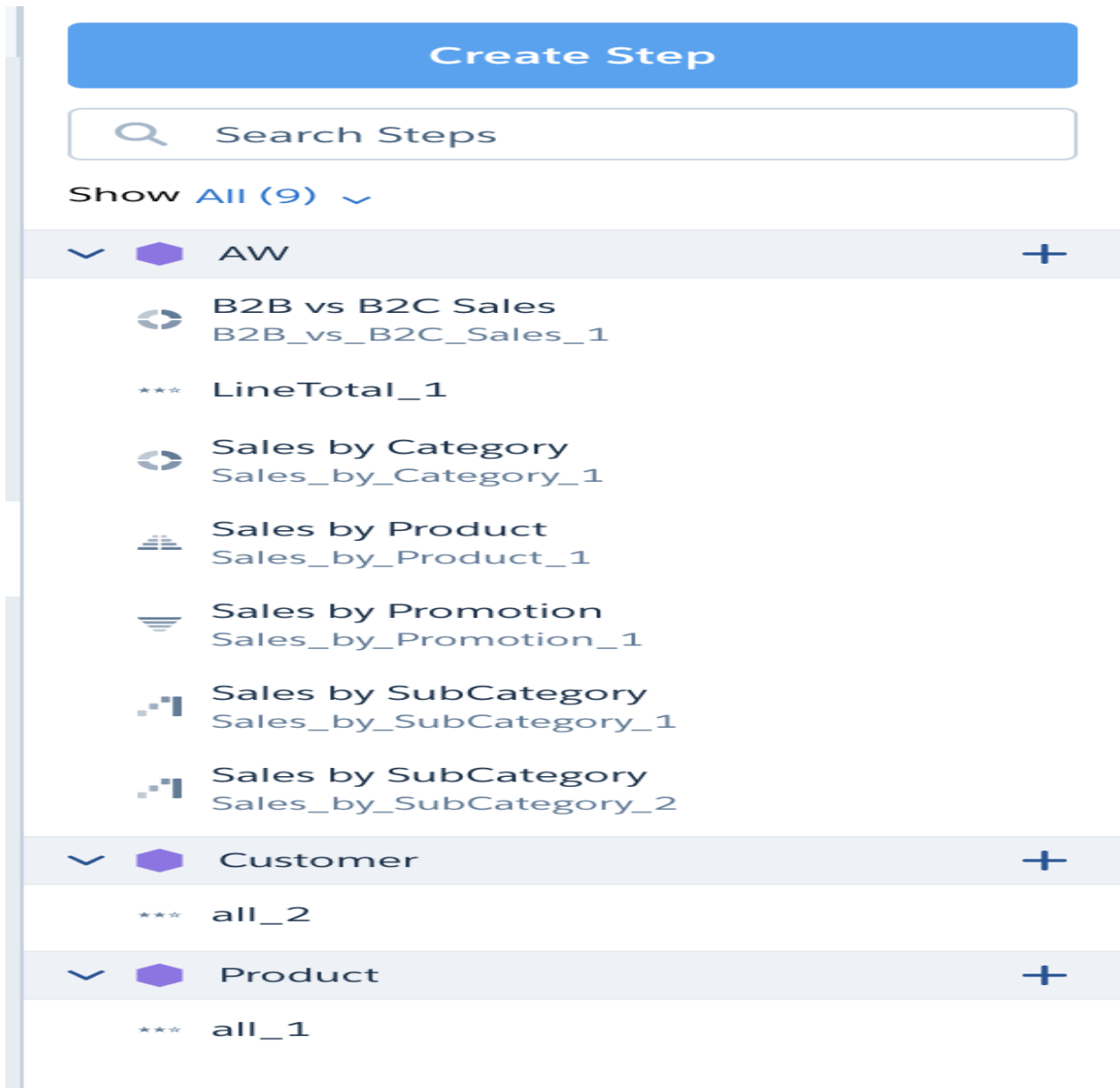
1. Bindings enable interactions among different components in a dashboard. You control the interactions by binding steps to each other. It is in Json format and can be viewed by using Ctrl+E
1. Below is a screenshot of the dashboard we created in our demo

Total Sales
●●●●●●●●● 77M

Total Customers
■■■■■■■■ 8.2K

Total Products
★★★★★★★★★★ 504

Category	Subcategory	Product Name	Promotion Type
All	All	All	All

B2B vs B2C Sales
4.9M
49M
44M

Sales by Category
7.6M
70M
61M

Sales by SubCategory

70M · 34M (Road Bikes) · 25M (Mountain Bikes) · 3.2M (Mountain Frames) · 3.1M (Road Frames) · 1M (Touring Bikes) · 779K (Wheels) · 272K (Jerseys) · 241K (Touring Frames) · 230K (Tights) · 187K (Bib-Shorts) · 182K (Gloves) · 168K (Helmets) · 120K (Shorts) · 106K (Handlebars) · 88K (Forks) · 68K (Headsets) · 36K (Cranksets) · 32K (Vests) · 26K (Bike Racks) · 25K (Pedals) · 21K (Caps) · 18K (Locks) · 18K (Pumps) · 16K (Hydration Packs) · 11K (Socks) · 10K (Derailleurs) · 10K (Saddles) · 9.8K (Brakes) · 9.5K (Bottom Brackets) · 6.6K (Tires and Tubes) · 4.3K (Bottles and Cages) · 3.1K (Cleaners) · 1.7K (Fenders) · 1.2K (Bike Stands) · 1.1K (Chains) · 1.1K · 70M (End)

Sales by Promotion

Volume Discount	Discontinued Product	New Product	Excess Inventory	Seasonal Discount
2.8M / 476K	286K	201K / 152K	56K	10K / 8.4K

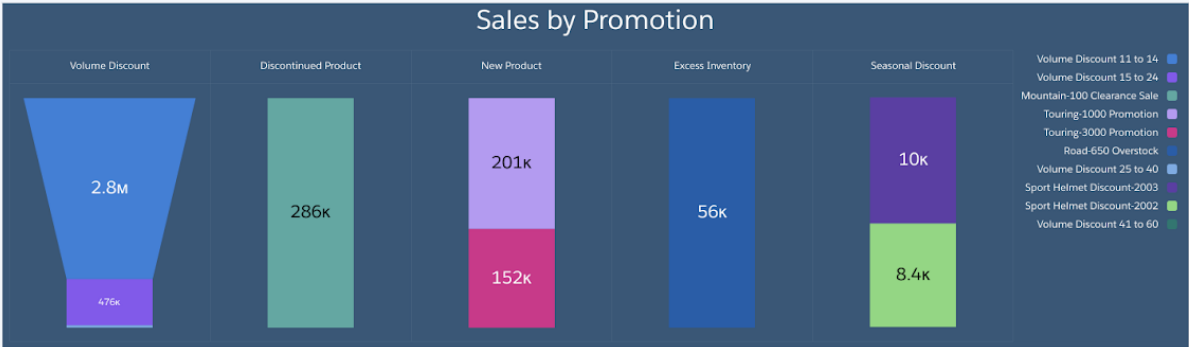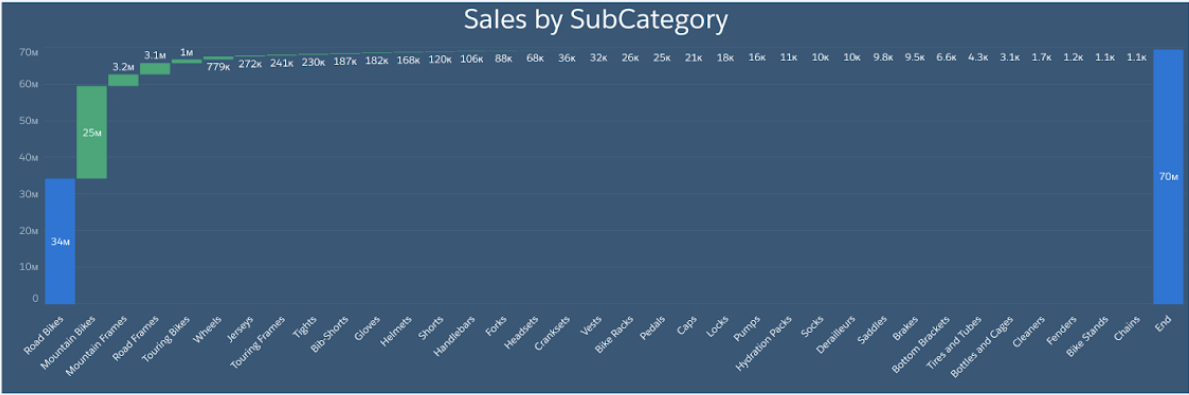Legend:
- Volume Discount 11 to 14
- Volume Discount 15 to 24
- Mountain-100 Clearance Sale
- Touring-1000 Promotion
- Touring-3000 Promotion
- Road-650 Overstock
- Volume Discount 25 to 40
- Sport Helmet Discount-2003
- Sport Helmet Discount-2002
- Volume Discount 41 to 60

Top 20 Products

Product	Sum of LineTotal ↓	Count of Rows
Mountain-200 Black, 38	2.2M	627
Mountain-200 Black, 42	2.2M	619
Road-150 Red, 56	2M	520
Road-150 Red, 62	2M	538
Mountain-200 Silver, 38	2M	572
Mountain-200 Silver, 46	1.9M	552
Mountain-200 Silver, 42	1.8M	529
Road-150 Red, 48	1.8M	465
Road-250 Red, 44	1.7M	414
Mountain-200 Black, 46	1.7M	535
Road-250 Black, 44	1.6M	441
Road-150 Red, 52	1.6M	419
Mountain-100 Black, 44	1.5M	269
Road-250 Red, 48	1.5M	418
Mountain-100 Black, 38	1.5M	266
Road-150 Red, 44	1.5M	395
Mountain-100 Silver, 38	1.5M	265
Road-250 Black, 48	1.5M	435
Mountain-100 Black, 42	1.4M	261
Mountain-100 Black, 48	1.4M	270

-----This concludes our first demo----

Conclusion

We now have a good understanding of importing the data from various local, external tools(both salesforce and syced remote source) and build our models to bring out insights of our data.

Further reading
- [Various Transformations in Dataflow](#)
- [Recipe Functions](#)
- [Youtube tutorials](#)

Reference docs
- http://www.einsteinanalyticslearningmap.com
- https://trailhead.salesforce.com
- https://www.kaggle.com/datafiniti/womens-shoes-prices