**Final Project**

Power BI

**Problem Statement:**

Data is continually published either internally by the company or on the web by external sources. We need to be able to refresh our data easily and report on it. We need an easy way to be able to analyze the data and create visualizations that are targeted towards specific user groups.

**Overview of the Technology:**

## Power BI is a suite of business analytics tools that deliver insights throughout your organization. Connect to hundreds of data sources, simplify data prep, and drive ad hoc analysis. Produce beautiful reports, then publish them for your organization to consume on the web and across mobile devices. Everyone can create personalized dashboards with a unique, 360-degree view of their business. And scale across the enterprise, with governance and security built-in.

In this project, I have used just one data source but have split it into multiple .csv files(one per each year with data) and also created a Windows PowerShell script to continually refresh power BI dataset from these files using the Power BI API. In the Power BI app, I have also created reports and dashboards that provide different views of the same data.

**High Level Steps:**

1. Download data and split it into multiple files by year
2. Installed PowerShell module PowerBIPS
3. Created PowerShell script to upload .csv file from local file system to Power BI
4. load to Power BI dataset one file at a time to show dataset refresh.
5. Created reports and dashboards in Power BI app.

**Data Source:**

<https://catalog.data.gov/dataset/age-adjusted-death-rates-for-the-top-10-leading-causes-of-death-united-states-2013>

**Hardware Used:**

Windows 7 64 bit processor laptop

**Software Used:**

Windows PowerShell

**Lessons Learnt:**

**Pros:**

* Visualizations are easy to create and there is a wide variety of visualizations that can be downloaded from the marketplace within Power BI app.
* Reports and dashboards can be easily refreshed with every data refresh.

**Cons:**

* Documentation on calling Power BI API using PowerShell is limited.
* The Power BI API does not allow refresh of dataset if the dataset was created within the Power BI App instead of being created via the API.

**GitHub Link:**

<https://github.com/vasaviramesh/AzureLearning/tree/master/FinalProject>

**YouTube Links:**

2 Min: <https://youtu.be/rxLlbJBK3AY>

15 Min: <https://youtu.be/UjSxATsSgrw>

Power BI is a suite of business analytics tools that help visualize data from different sources. Data can be loaded from different sources:

* My Organization - Data published by others within your organization to Office365 groups.
* Online Services - Data from online services like JIRA, GitHUb etc.
* Files – Reports, Workbooks or data from excel, Power BI Desktop(.PBIX file) or from CSV files
* Databases – Azure SQL Database, Spark on Azure HDInsight etc.

This solution uses csv files as data source and uploads the data to Power BI dataset using Windows PowerShell script and Power BI API.

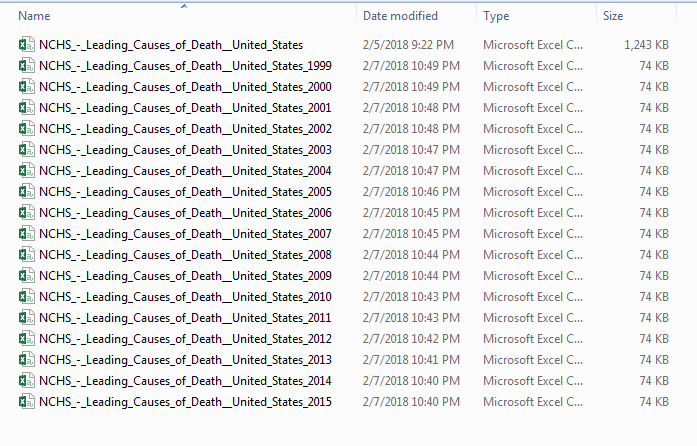
## Step 1: Download data and split it into multiple files by year

Downloaded the data for cause of death csv from the below url…

<https://catalog.data.gov/dataset/age-adjusted-death-rates-for-the-top-10-leading-causes-of-death-united-states-2013>

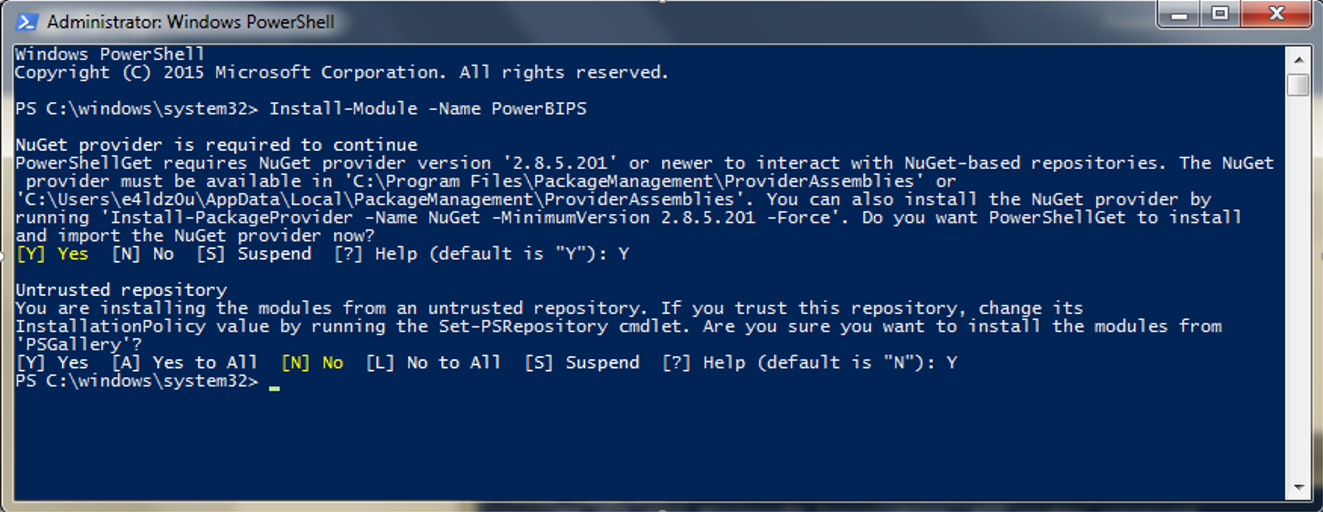


This file has data for all the states categorized by cause of death, year, state and average age. I manually split this file into multiple csv files based on year.

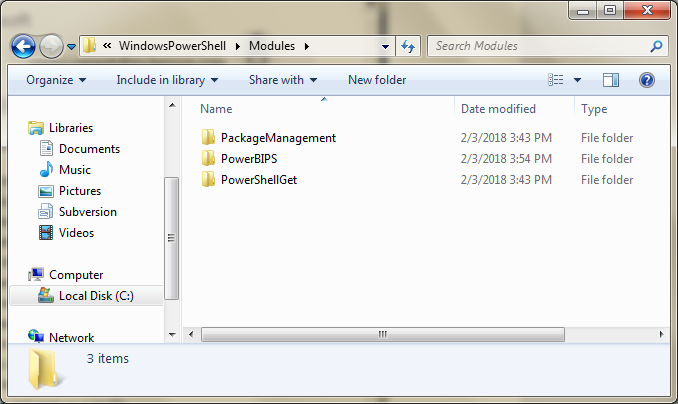


## Step 2: Installed PowerShell module PowerBIPS

Downloaded the PowerBIPS module for Windows PowerShell from <https://github.com/DevScope/powerbi-powershell-modules> (NOTE: Run PowerShell as administrator)



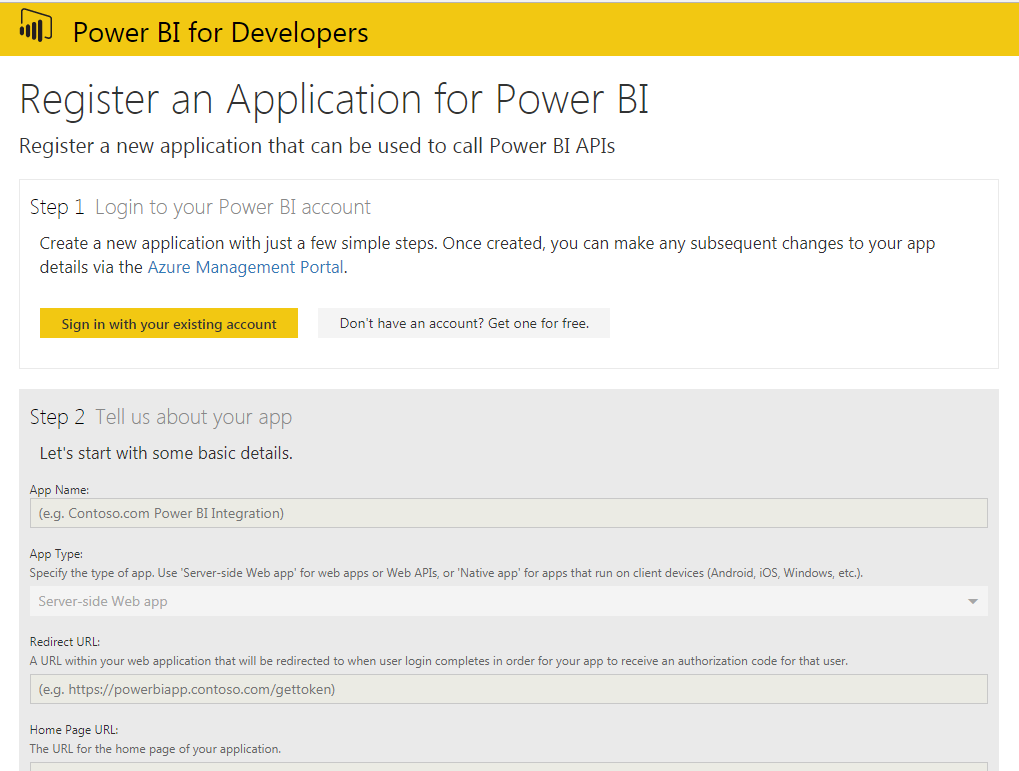
Once the install is complete, you should be able to browse to the PowerBIPS module located @Program Files\WindowsPowerShell\Modules



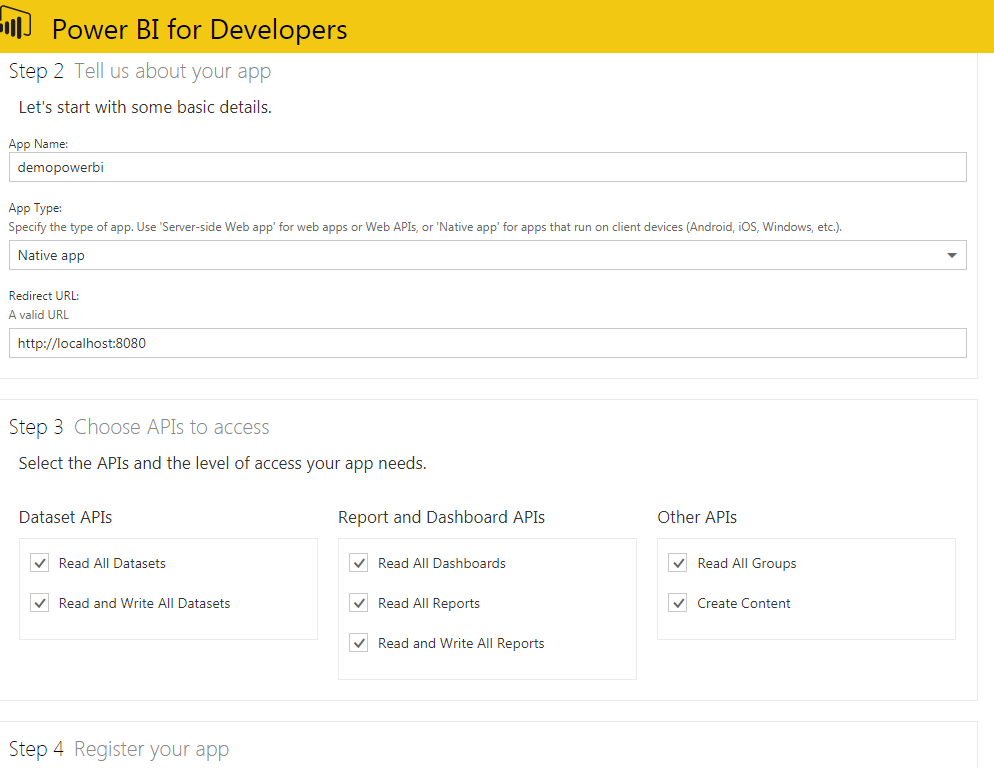
## Step 3: Create PowerShell script to upload .csv file from local file system to Power BI

Following the instructions on the <https://github.com/DevScope/powerbi-powershell-modules> page, I created a PowerShell script to look for any csv file in my current path and when found, try to move that to the dataset specified in the script.

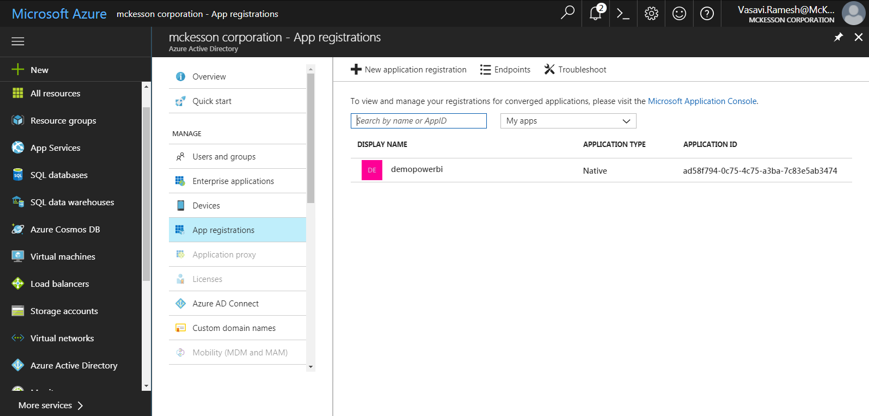
But before doing that, we have to register our Power BI app with Azure Active Directory and setup the right permissions. To register the app navigate to  [dev.powerbi.com/apps](https://dev.powerbi.com/apps). (NOTE: can be done via azure portal too. Instructions are located [here](https://docs.microsoft.com/en-us/power-bi/developer/walkthrough-push-data-register-app-with-azure-ad)).



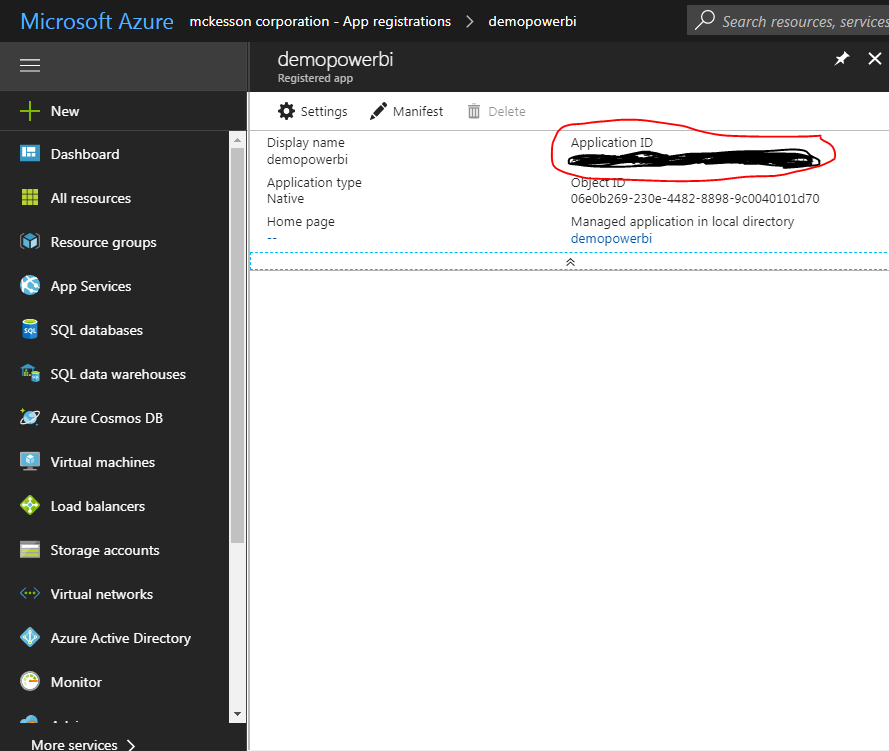
**Note:** Make sure you register the app as a ‘Native App’ and check all the checkboxes under Step3 Choose APIs to access and then click ‘Register App’. This should populate the Client ID on the screen which will be your secret key for API access.



Once the registration is complete, you should be able to view it view the azure portal(Azure Active Directory 🡪 App Registrations).



**NOTE:** If you forgot to save your client ID when you registered your app, you can still get to it via the portal by clicking on the registered app and it is listed as the application ID



Now we are ready to make REST API calls to Power BI to load our dataset using the PowerShell script. This script will look for any csv file in the current path and load it as a dataset to Power BI. Once the load is completed, the file will be moved to the archive directory. Below is the script used to load the dataset….

cls

$ErrorActionPreference = "Stop"

$currentPath = (Split-Path $MyInvocation.MyCommand.Definition -Parent)

#Create Archive Folder

new-item -Name "Archive" -Force -ItemType directory -Path "$currentPath" | Out-Null

Import-Module "C:\Program Files\WindowsPowerShell\Modules\PowerBIPS" -Force

while($true)

{

# Iterate each CSV file and send to PowerBI

Get-ChildItem "$currentPath" -Filter "\*.csv" |% {

$file=$\_

#Import csv and add column with filename

$data = Import-Csv $\_.FullName

# Send data to PowerBI

$data | Out-PowerBI -dataSetName "DeathByYear" -tableName "DeathByYear" -batchSize 300 -verbose

# Archive the file

Move-Item $file.FullName "$currentPath\Archive\" -Force

}

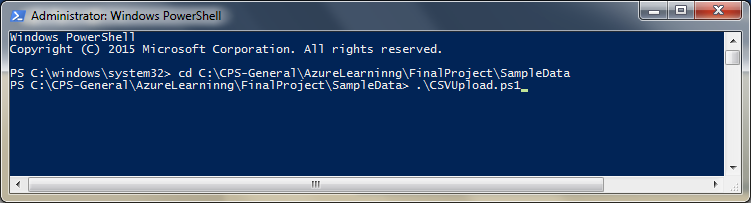
Write-Output "Sleeping..."

Sleep -Seconds 5

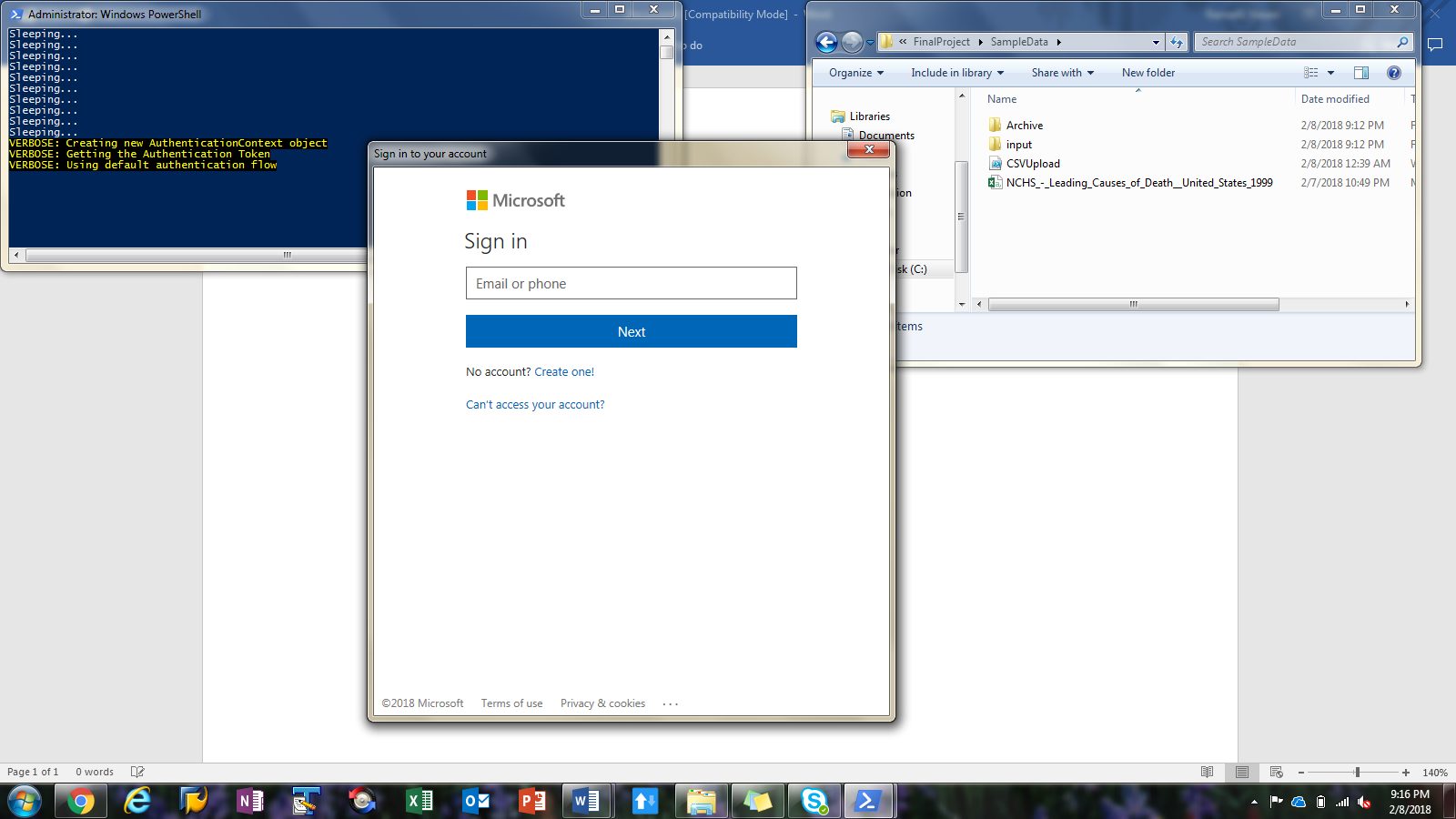
}

## Step 4: load to Power BI dataset one file at a time to show dataset refresh.

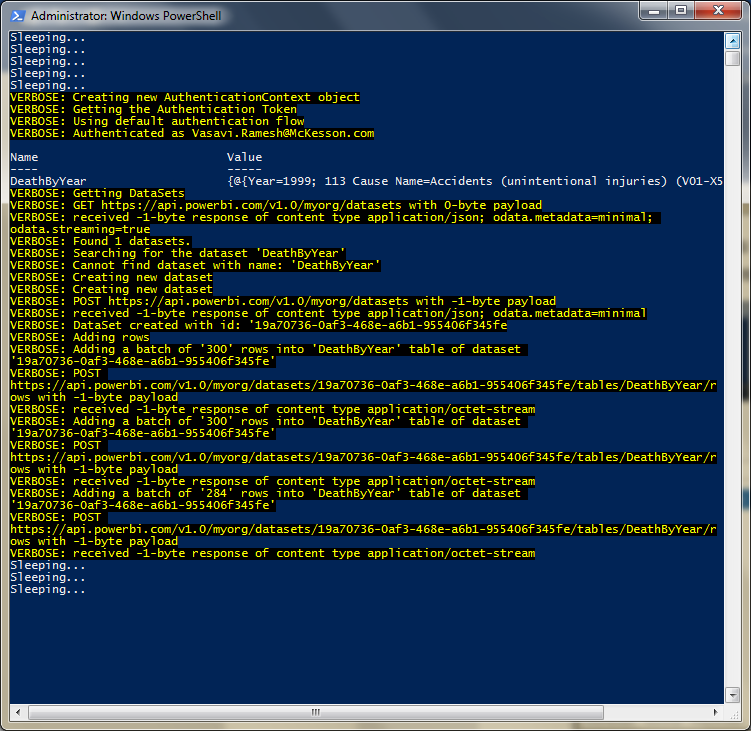
Run the script from powershell…



The script will start listening for csv files in the current path every 5 seconds…The very first time it finds a csv file, it prompts user to connect to power BI

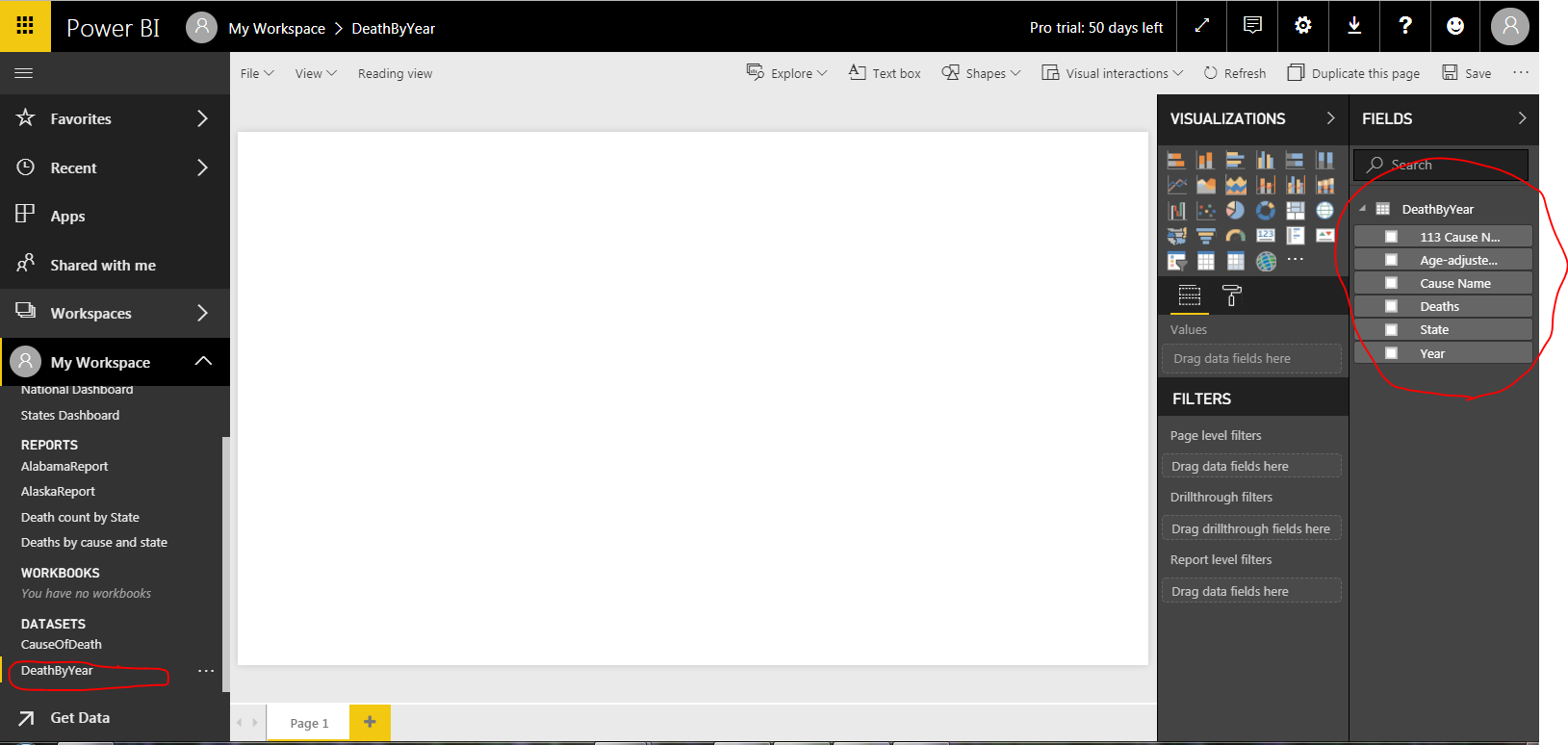


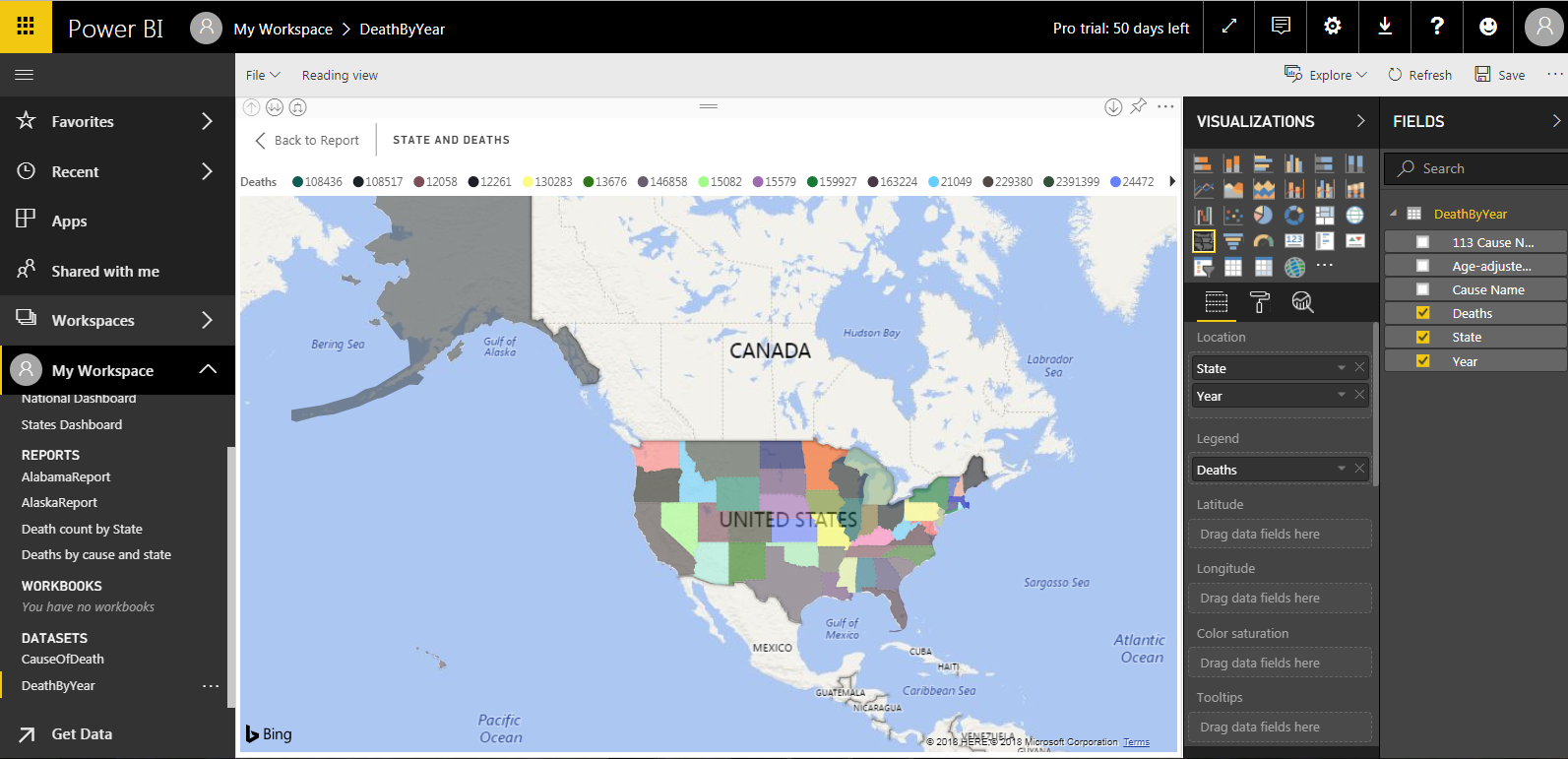
Once successfully logged in, the script continues to load the dataset…

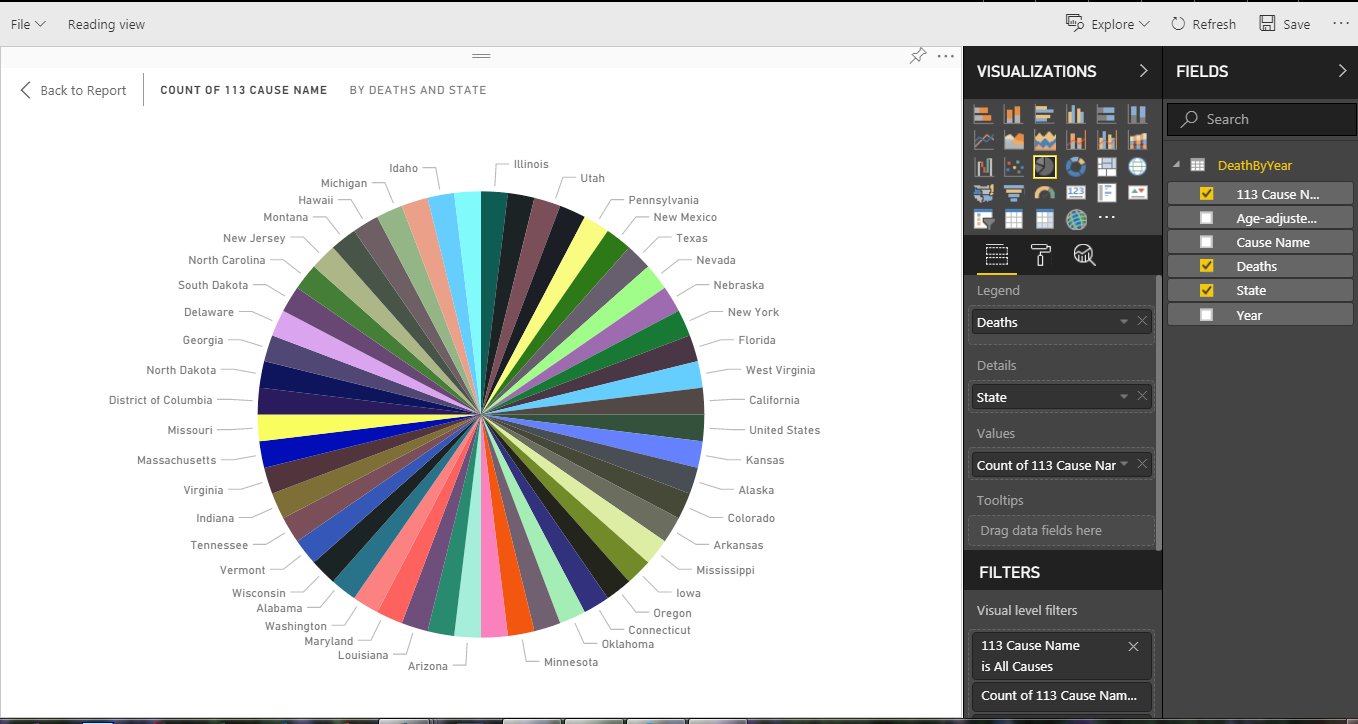


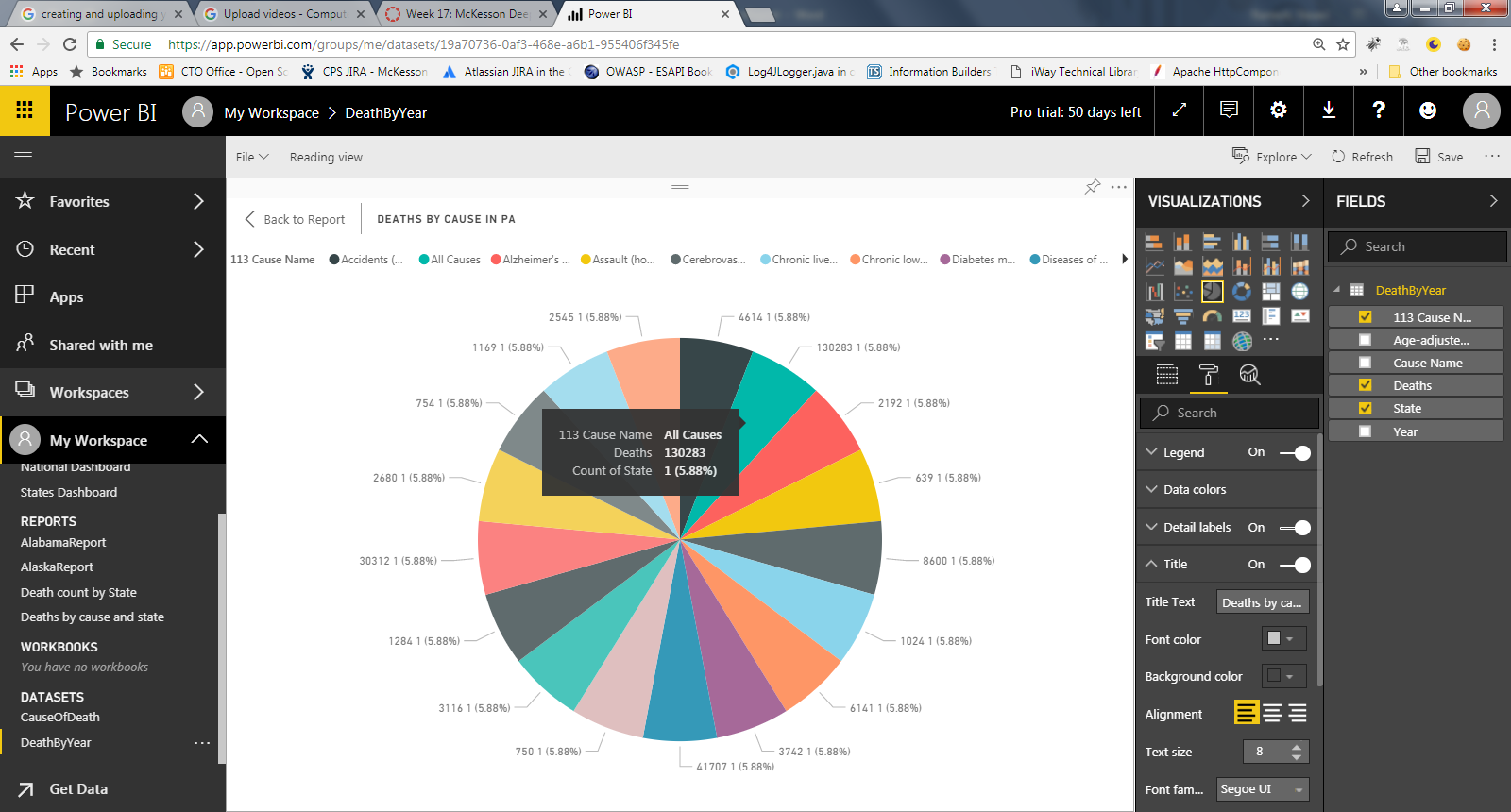
After load is completed, it will keep listening for additional csv files. If we login to <https://app.powerbi.com>, we can see the dataset loaded by the script and start building our visualizations.

## Step 5: Create reports and dashboards in Power BI app.

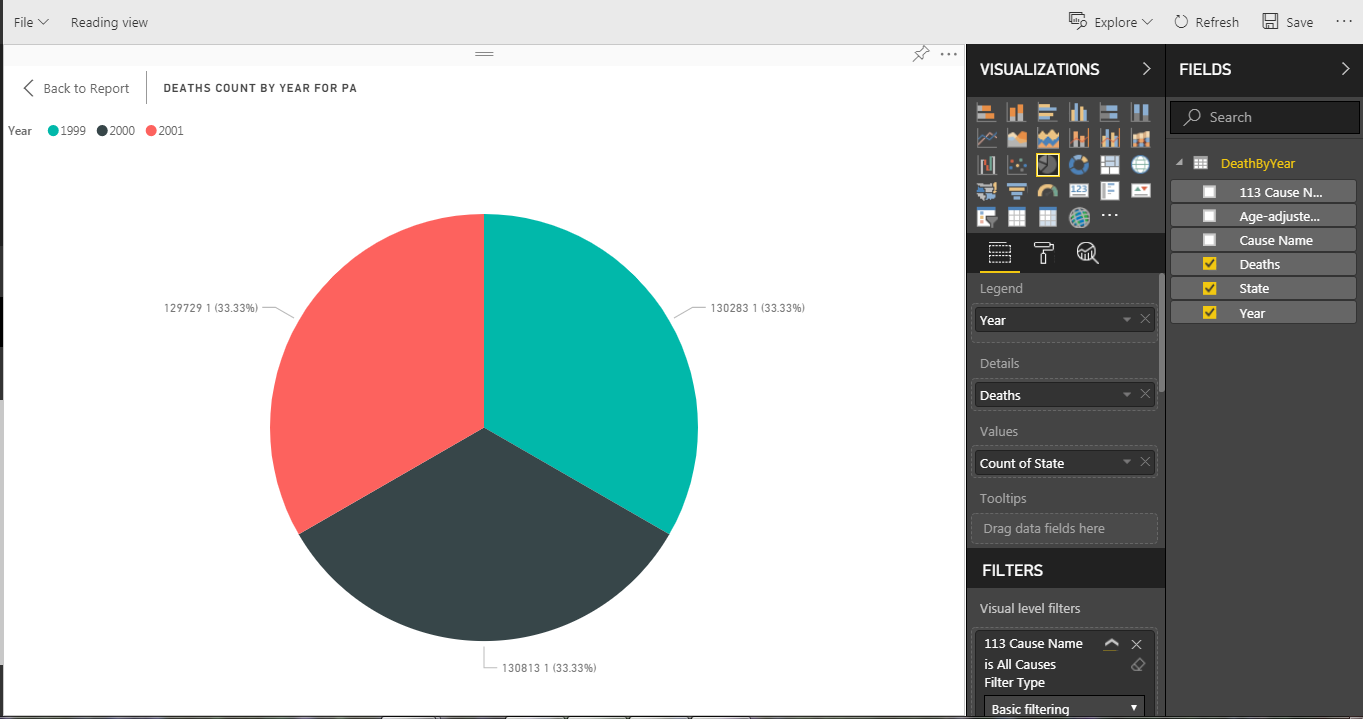


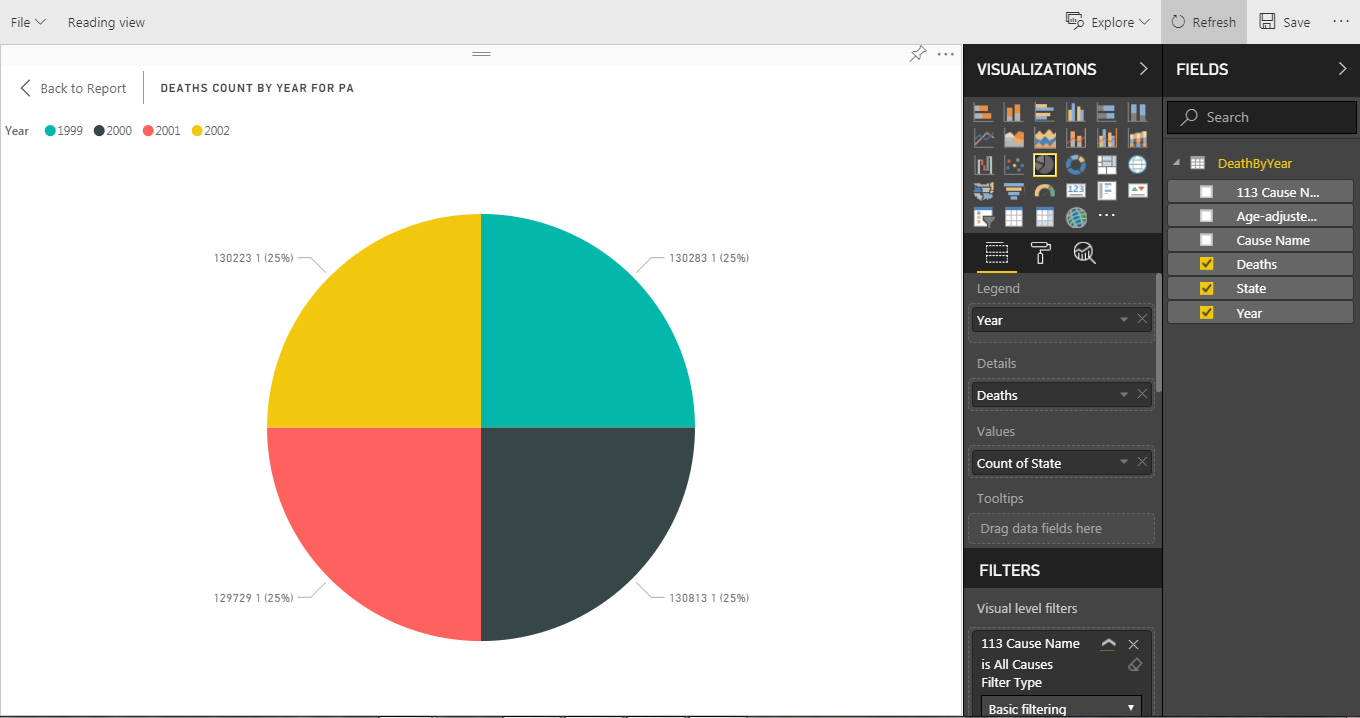




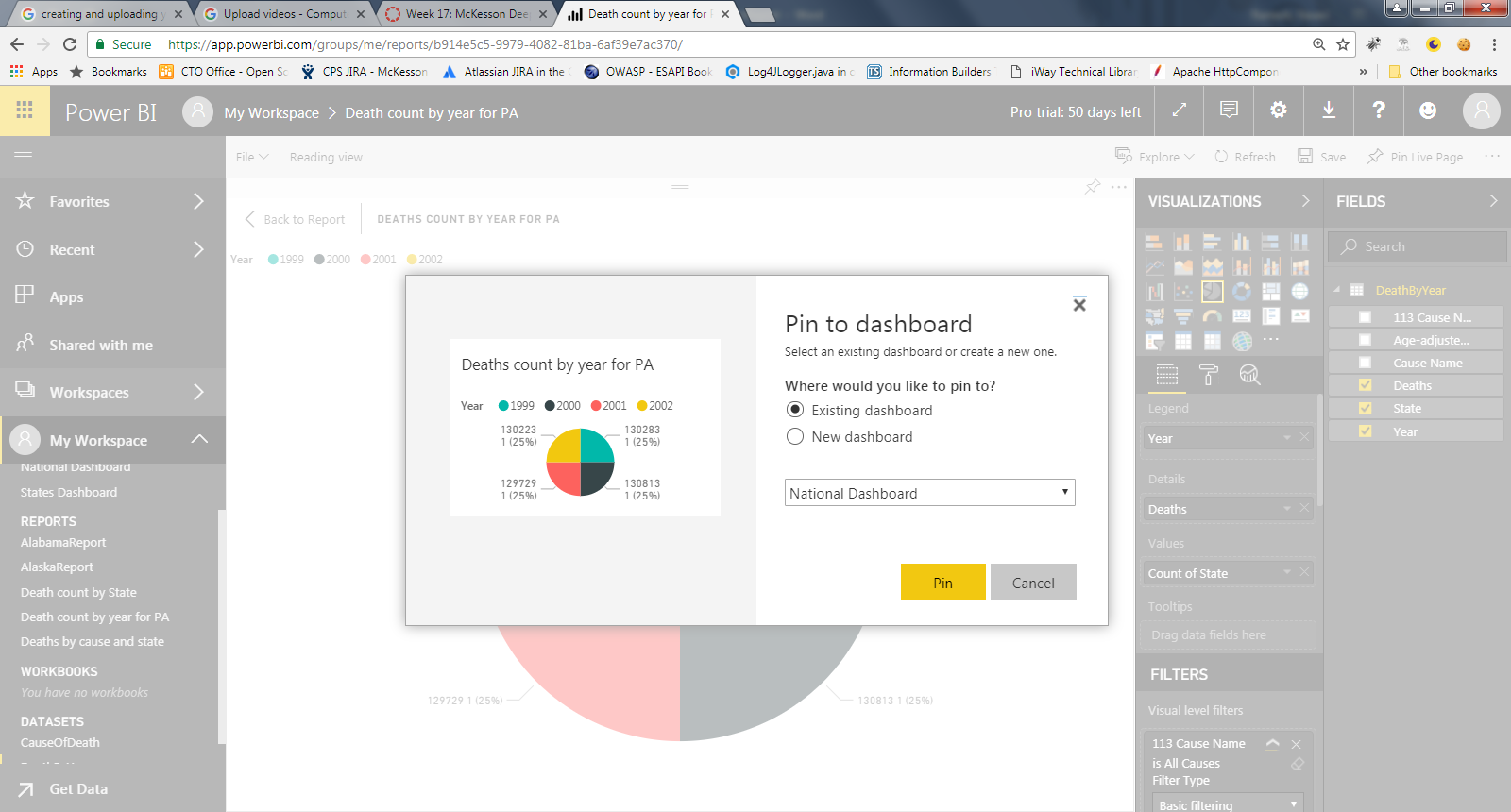


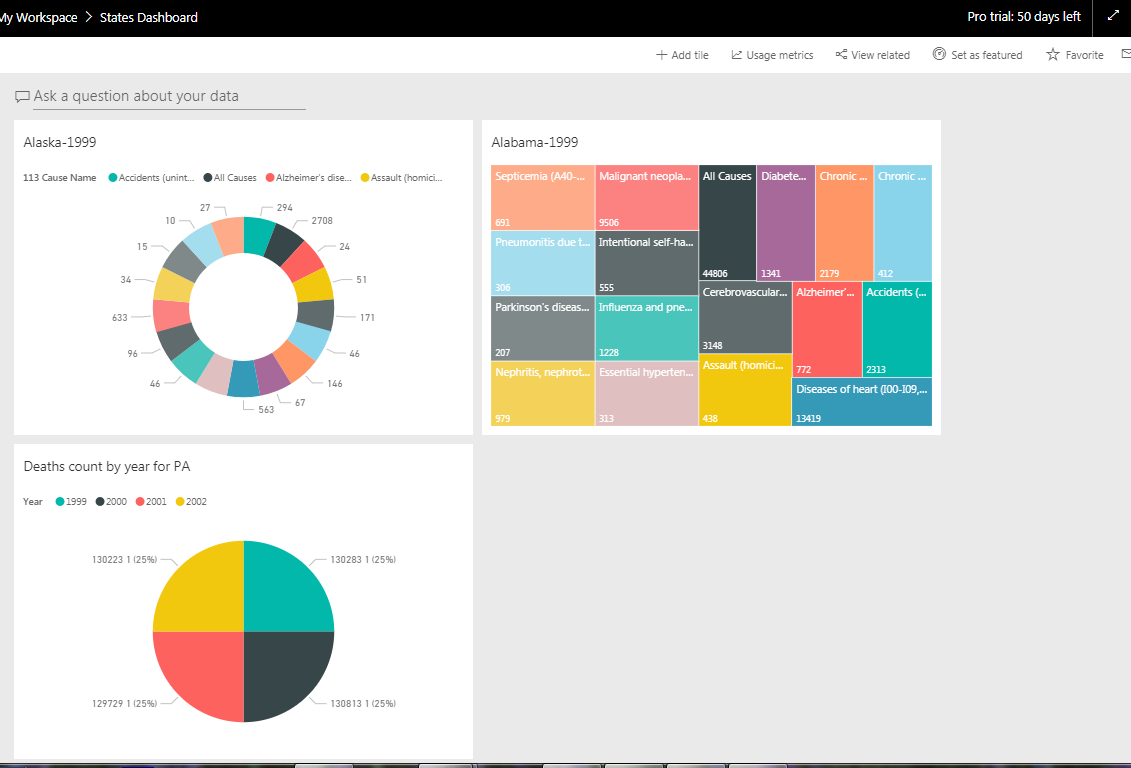
Once you load data for multiple years, the data will be refreshed in the Power BI dataset and the report visual can be refreshed to reflect latest data.

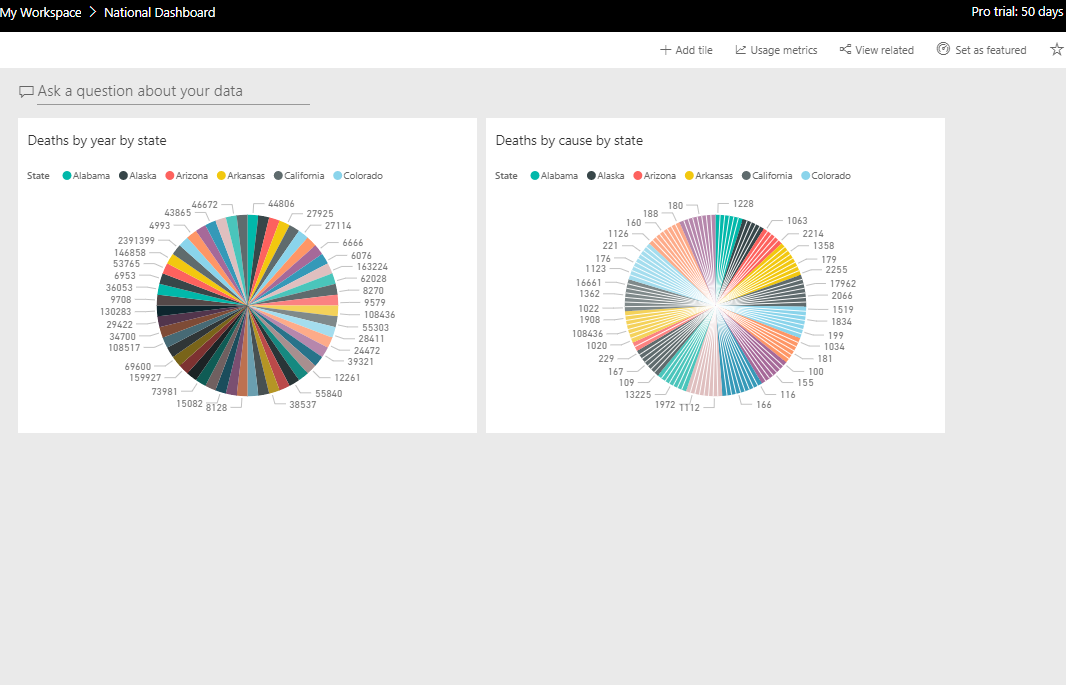




The reports can be saved and pinned to multiple dashboards and shared either internally or externally. (NOTE: people viewing these reports need to have access to the dashboard).







Further Learning: The visualizations created with Power BI can be embedded into applications. This will help the business get real-time data access and visualization. But to do this the user creating the reports needs access to create groups in Office 365.