Microsoft Cloud Workshop

Big Data & Visualization   
Hackathon Learner Guide

September 2016

© 2016 Microsoft Corporation. All rights reserved. This document is confidential and proprietary to Microsoft. Internal use only. This document is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS OR IMPLIED, IN THIS SUMMARY.

This document is provided "as-is." Information and views expressed in this document, including URL and other Internet Web site references, may change without notice. You bear the risk of using it.

Some examples are for illustration only and are fictitious. No real association is intended or inferred.

## Contents

[Big data & visualization hackathon learner guide 4](#_Toc469316062)

# Big data & visualization hackathon learner guide

## Overview

AdventureWorks Travel (AWT) provides concierge services for business travelers. In an increasingly crowded market, they are always looking for ways to differentiate themselves and provide added value to their corporate customers.

They are looking to pilot a web-app that their internal customer service agents can use to provide additional information useful to the traveler during the flight booking process. They want to enable their agents to enter in the flight information and produce a prediction as to if the departing flight will encounter a 15 minute or longer delay, taking into account the weather forecasted for the departure hour.

In this hackathon, learners will build an end-to-end solution to predict flight delays taking into account the weather forecast.

## Requirements

* A corporate e-mail address (e.g., your @microsoft.com email)
* Microsoft Azure Subscription must be pay-as-you-go or MSDN
  + Trial subscriptions will *not* work
* Local machine or a virtual machine configured with:
  + Visual Studio 2015 Community Edition or later
  + Azure SDK 2.8.2 for Visual Studio
  + Azure PowerShell 1.0.0 or later

## Exercise 0: Before the hackathon

Duration: 60 mins

Synopsis: Before attending the hackathon, you should follow these steps to prepare your environment for an efficient day.

### Task 1: Provision Power BI

1. If you do not already have a Power BI account, go to <https://www.powerbi.com>.
2. On the page, enter your work email address (it should be the same account as the one you use for your Azure subscription) and click **Use it for free**.

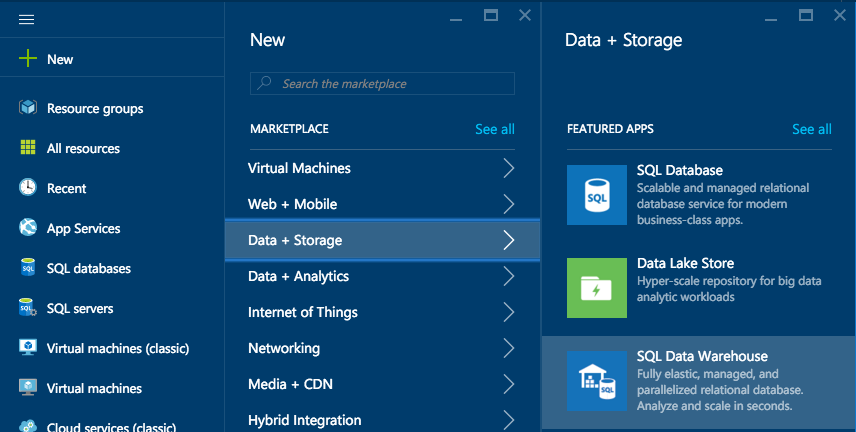


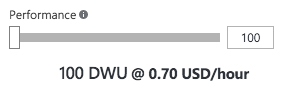
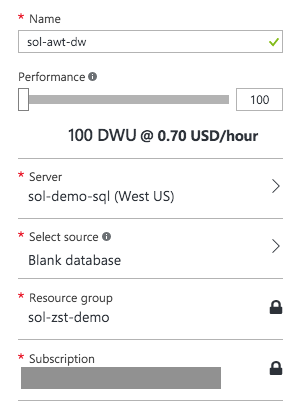
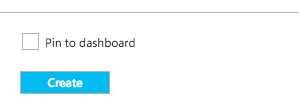
1. Follow the on-screen prompts and your Power BI environment should be ready within minutes. You can always return to it via <https://app.powerbi.com>

### Task 2: Provision Azure SQL Data Warehouse

Using the Azure Portal, provision a new instance of SQL Data Warehouse.

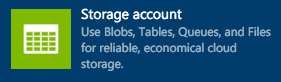
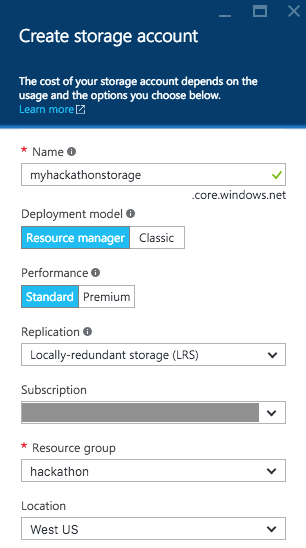
1. Click **+New**, select **Data + Storage**, **SQL Data Warehouse**.



1. Provide a Name for the SQL Data Warehouse.  
   
2. Set the **Performance** to **100 DWU**. (You will not need any more for this hackathon.)  
     
   
3. Select Server.  
   ../../../../../../Captures/Screen%20Shot%202016-04-05%20at%208.49.42%20AM.png
4. Create a new server or use an existing server as desired.
5. Select **Create**.  
     
   

### Task 3: Provision a Storage Account

Using the Azure Portal, provision a new Azure Storage Account to use for this Hackathon.

1. Click **+New**, select **Data + Storage**, **Storage Account**.   
   
2. Provide a Name for the storage account.
3. For the resource group, add it to the Resource Group you are using for this Hackathon.
4. For the location, choose the same Location as your SQL Data Warehouse.   
   
5. Select **Create**.

## Exercise 1: Environment setup

Duration: 20 minutes

AdventureWorks Travel has provided a sample solution you can use as a starting point for integrating the features they have requested.

### Task 1: Download and unzip the sample

### Tasks to Complete

* Download the sample project from the following link: <http://bit.ly/238cJDY>
* Open the solution in Visual Studio
* Publish the web project to a new Azure Web App

### Exit criteria

* You can browse to the deployed sample (https://<yourappname>.azurewebsites.net/Main) and view the starting point of the prediction page.

### Task 2: Register Web App with Azure Active Directory

### Tasks to Complete

* Login to the Manage Portal, and within your default Azure Active Directory add an application that represents your web app (whose sign-in URL and APP ID URL are both the deployment address of your web app).
* Configure the app to have a key and copy this key value.
* Paste the key value into the ClientSecret setting present in the Visual Studio Project properties, settings.
* From the app in Azure Active Directory, also copy the ClientID and paste this key value into the ClientID setting in the Project properties, settings.
* Also within the settings pane, set the RedirectURL property to be the URL of your newly deployed Web App, with /Redirect appended to the end (e.g., <http://yourapp.azurewebsites.net/Redirect)>.
* Save the project.

### Exit criteria

* You have set the values for the ClientID, ClientSecret and RedirectURL settings.

### Task 3: Update sample app with Storage Account credentials

### Tasks to Complete

* From the Portal, copy the Connecting String to your Storage Account
* Paste the Connection String value into the project properties, settings for the setting named “ML\_StorageAccount”
* Save the project.

### Exit criteria

* You have set the value for the ML\_StorageAccount setting.

## Exercise 2: Check the weather

Duration: 20 minutes

AdventureWorks has researched providers of Weather Forecasts and determined that they prefer the data from WeatherUnderground, which they would like to see the 10-day weather forecast integrated into starter web app provided.

### Task 1: Register for trial account in WeatherUnderground.com

### Tasks to Complete

* Register for a developer trial account for WeatherUnderground, from <http://www.wunderground.com/weather/api/>
* Confirm your email address
* Login to the site with your wunderground.com account and get your API Key for the Anvil Plan at the Developer level

### Exit criteria

* You are able to copy your API Key (e.g., the “Key ID”)
* You can navigate to [http://api.wunderground.com/api/<YOURAPIKEY>/hourly10day/q/CA/SEATAC.json](http://api.wunderground.com/api/%3cYOURAPIKEY%3e/hourly10day/q/CA/SEATAC.json) in the browser after substituting in your API Key and see the 10-day hourly weather forecast for the Seattle-Tacoma International Airport

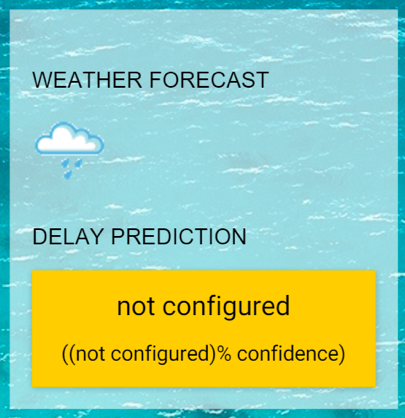
### Task 2: Update sample Web App with API Key

### Tasks to Complete

* In Visual Studio, navigate to the project properties, settings and set the Weather\_APIKey to have the value of the API Key you were provided.
* Deploy the updated web app.

### Exit criteria

* With the deployed web app running in your browser, click Predict. You should be able to see a prediction for just the weather, similar to the following:



## Exercise 3: Building a ML Model

Duration: 90 minutes

AdventureWorks Travel is interested in building a Machine Learning solution to power their delay predictions. They have heard of the capabilities to build and operationalize Machine Learning using Azure Machine Learning, but are relying on you to implement it for them.

### Task 1: Create ML Workspace

### Tasks to Complete

* Create the Machine Learning Workspace and open it within ML Studio. You should follow the detailed step by step for this Task available in the Hackathon Leader Guide.

### Exit criteria

* You can open your Workspace in ML Studio.

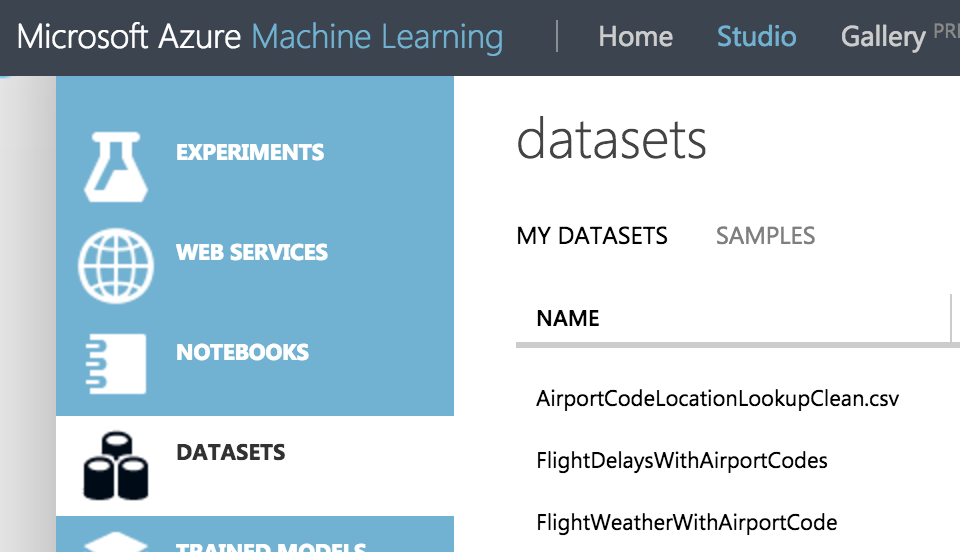
### Task 2: Upload the Sample Datasets

### Tasks to Complete

* Download the three CSV sample datasets from here: <http://bit.ly/1Hrm5es>
* Extract the ZIP and verify you have the following files:
  + FlightDelaysWithAirportCode.csv
  + FlightWeatherWithAirportCodes.csv
  + AirportCodeLocationClean.csv
* Upload them as new Datasets available in ML Studio

### Exit criteria

* You should be able to navigate to the Datasets page in ML Studio and see your three CSV datasets:



### Task 3: Start a New Experiment

### Tasks to Complete

* Create the Machine Learning Experiment within ML Studio. You should follow the detailed step by step for this Task available in the Hackathon Leader Guide.

### Exit criteria

* You can have an Experiment that prepare the data from the FlightDelaysWithAirportCodes dataset.

### Task 4: Prepare the Weather Data

### Tasks to Complete

* Update Machine Learning Experiment within ML Studio to prepare the FlightWeatherWithAirportCode. You should follow the detailed step by step for this Task available in the Hackathon Leader Guide.

### Exit criteria

* You can have an Experiment that prepare the data from the FlightWeatherWithAirportCode dataset.

### Task 5: Join the Flight and Weather datasets

### Tasks to Complete

* Join the two datasets- the data prepared from FlightDelaysWithAirportCodes and the data prepared from FlightWeatherWithAirportCode. You should follow the detailed step by step for this Task available in the Hackathon Leader Guide.

### Exit criteria

* You can have an Experiment that has a single dataset that provides information about flights and the weather at the time of departure.

### Task 6: Train the model

### Tasks to Complete

* Complete the Experiment by training a model using a Two-Class Logistic Regression. You should follow the detailed step by step for this Task available in the Leader Hackathon Guide.

### Exit criteria

* You should be able to Evaluate your model’s performance and verify that its predictions are performing better than random.

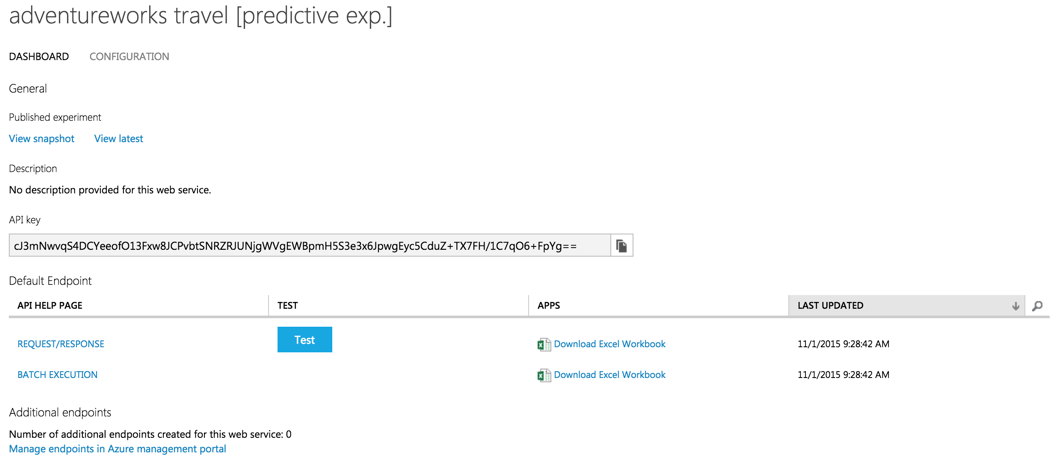
### Task 7: Operationalize the experiment

### Tasks to Complete

* Operationalize the model by packaging it up as a Predictive Web Service. You should follow the detailed step by step for this Task available in the Hackathon Leader Guide.

### Exit criteria

* You should be able to view the Web Service dashboard for your deployed Predictive Web Service, similar to the following:



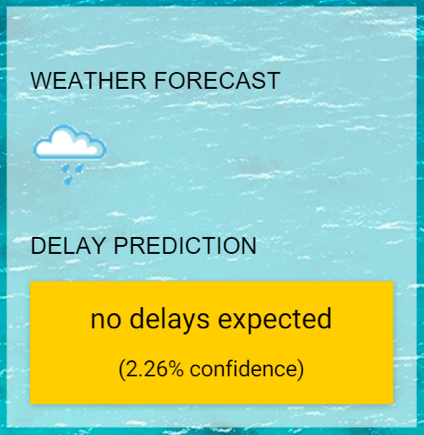
### Task 8: Integrate the Web Service with the Sample Web App

### Tasks to Complete

* Copy the value of the API Key from the Web Service Dashboard.
* Paste this value for the setting named ML\_APIKey.in the sample web app project.
* Retrieve your Workspace ID and Service ID and paste these values into the settings named ML\_RegionPrefix, ML\_WorkspaceID and ML\_ServiceID, respectively.
* Publish the solution and navigate to the home page.

### Exit criteria

* You should be able to fill out the flight criteria, and click Predict Delays (remember to update the departure date so it is within 10 days of today). In addition to the Weather Forecast, you should see the delay prediction and confidence, similar to the following:



## Exercise 4: Batch score data with Azure ML and summarize with SQL Data Warehouse

Duration: 60 minutes

Besides being able to predict delays for an individual flight from the web app, AdventureWorks Travel would like to be able to display a map that shows the number of predicted flight delays by airport for a given period of time. Each airport would have a circle above it whose diameter increases as the number of predicted delays increases. Fabrikam would like to use the Batch Scoring capabilities of Azure ML to score all departing flights for a particular period, and then summarize these results by Airport using SQL Data Warehouse.

### Task 1: Batch score flight data

### Tasks to Complete

* Download the sample file AdventureWorks Travel has provided for batch scoring: <http://bit.ly/1XGg6M5>
* Using Visual Studio or your favorite Azure Storage explorer, upload this file to the container you created in your Storage Account. You should place it under the folder “flights”.
* Within Visual Studio, in the project properties, settings set the value of the ML\_StorageAccount to be the connection string to the Storage Account.
* Publish the web app and navigate to /BatchScore
* Within the BatchScore web page, modify the values of relatives paths for the location of the file to score and the location of scored output to that it is correct for your environment
* Submit, Start and Check the Job Status until the Job Status reads “Job is finished.”

### Exit criteria

* The scoring job should have completed successfully.
* If desired, using the Azure Storage explorer of your choice, verify that you can see the scored CSV file in the location you configured.

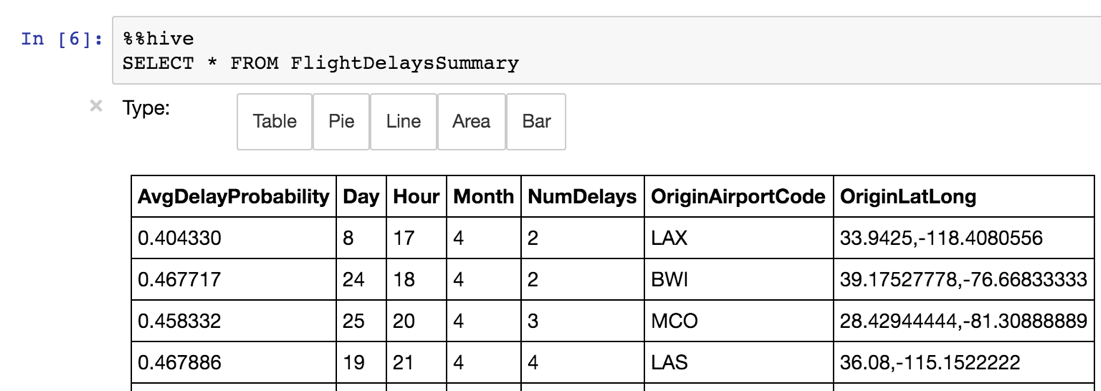
### Task 2: Summarize delays by airport

### Tasks to Complete

* Using Visual Studio or SQL Server Management Studio, connect to your SQL Data Warehouse.
* Create a table called FlightDelays to represent the scored CSV file you uploaded. It should have the columns and types (in parenthesis): OriginAirportCode (String), OriginLatLong (String), Month (Int), Day (Int), Hour (Int), Carrier, DelayPredicted (Int) and DelayProbability (Double). The OriginLatLong field should have values that are of the form “OriginLatitude,OriginLongitude”, such as 29.9832,-95341.
* Create another table called FlightDelaysSummary that summarizes the former and has the following columns: OriginAirportCode, OriginLatLong, Month, Day, Hour, NumDelays and AvgDelayProbability. NumDelays is computed by summing the values for DelayPredicted when the dataset is filtered to Month = 4 and the rows are grouped by OriginAirportCode, OriginLatLong, Month, Day and Hour. AvgDelayProbability is similarly computed, as the average of the DelayProbability column.

### Exit criteria

* You should be able to query the FlightDelaysSummary table and see results similar to the following:



## Exercise 5: Visualizing in Power BI

Duration: 20 minutes

AdventureWorks Travel would like to visualize the summarized predicted flight delays in a map visualization, using an embedded Power BI tile.

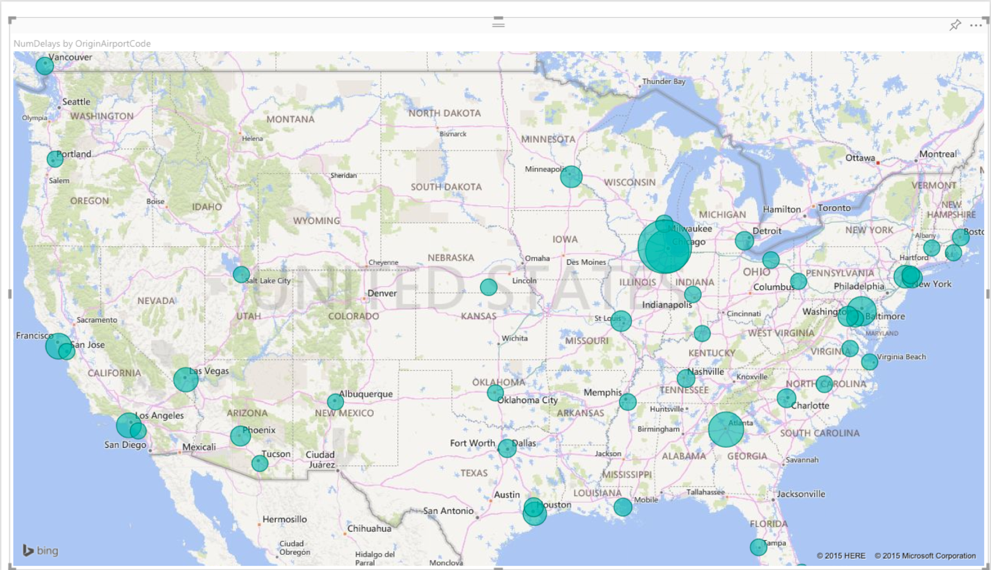
### Task 1: Connect Power BI to SQL Data Warehouse

### Tasks to Complete

* From Power BI, connect to your SQL Data Warehouse.
* View the dataset that represents your data.
* Add a Map visualization. It should use the OriginLatLong for the location and the NumDelays field for the Values.

### Exit criteria

* Your map visualization should display as follows:



### Task 2: Create a Dashboard

### Tasks to Complete

* Save the report you created in the previous task with the name FlightDelays.
* Pin this report to a new Dashboard that is named AdventureWorks.

### Exit criteria

* You should be able to see your map on the AdventureWorks dashboard.

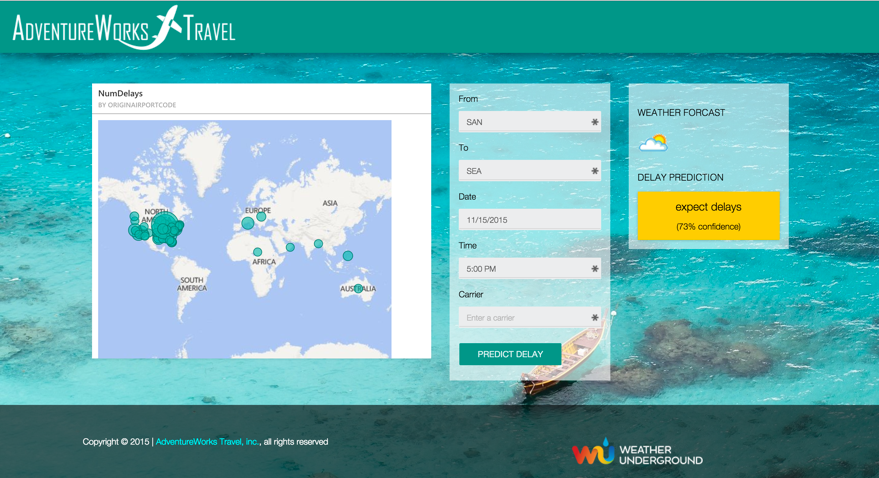
### Task 3: Observe the batch score results

### Tasks to Complete

* Open your deployed sample app in the browser, and navigate to /Main.
* Click the button labeled Sign In to Power BI and sign in with your Power BI credentials.

### Exit criteria

* Your browser should display the sample web app with the Power BI tile showing the FlightDelays map visualization:



## Exercise 6: Cleanup

### Tasks to Complete

* Delete your SQL Data Warehouse.
* Delete the Web App and the App Service Plan that hosts it.

### Exit criteria

* You have deleted or stopped the major, cost incurring services used in the hackathon. You may optionally choose to also delete the associated Storage account, but the costs for this are very minimal (and by keeping the Storage account you can re-create your clusters in the future to read from the data sets you have created).