Azure's Data Factory & Blob Storage

Lab 06

by

Diane Howard, Nishava Inc.

Deep Azure @McKesson

Overview

- What is Azure's Data Factory, How to use Data Factory, Costs
- Create Storage Account
- Create Active Directory for Client ID, Client Secret, Tenant ID
- Data Factory Demo in Python:
    1. Create Resource Group
    2. Create Data Factory
    3. Create Linked Storage for Blob Storage and Blob Sink
    4. Initialize Blob Storage for input/output data
    5. Create Pipeline
    6. Monitor your Pipeline
    7. Run your Pipeline

Objective of Demo

- Create a Data Factory in Azure within Python code that will ingest data from an Excel spreadsheet (or Blob), perform an identify transformation (identical copy) and transfer data to another Blob data store as a sink.

- This is accomplished via a workflow (job) initialized within the Data Factory.

Data Factory

- There are 2 versions of Data Factory within Azure: V1 and V2 (Preview)

- Defined as data-driven jobs (*aka workflows*) that have pipelines to move & transform data

1. **Ingest** data from disparate data stores (e.g., AZ blob/file/tables, SQL, Cosmos, Amazon Redshift, Informix, PostgresSQL, NoSQL, Amazon S3, FTP, HDFS, …)

https://docs.microsoft.com/en-us/azure/data-factory/concepts-datasets-linked-services

2. **Transform** or **process** the data by using compute services such as the following:
   - Azure HDInsight Hadoop
   - Spark
   - Azure Data Lake Analytics
   - Azure Machine Learning

3. **Publish output data** to data stores (e.g., AZ Blob, AZ Cosmos DB, SQL Server, Data Warehouse, Oracle, AZ Table storage, AZ Filesystem)

Example of Data Factory Usage

- Central place to manage processing of web log analytics, click stream analysis, social sentiment, sensor data analysis, geo-location analysis, etc.

TRANSFORMATIONS

- A gaming company collects logs from games in the cloud.
  - analyze logs to gain insights into customer preferences, demographics, usage behavior

Customers using Azure Data Factory

Data Factory V1 vs V2

V1

- ✓ Create data pipelines to move and transform data
- ✓ Run pipelines on a specified schedule (hourly, daily, weekly, etc.)
- ✓ Visualizations to display the lineage and dependencies between your data pipelines
- ✓ Monitor data pipelines
  - pinpoint issues and setup monitoring alerts.

V2 (Preview)

Primary:

1. **Control flow**:

   Branching, looping & conditional processing.
2. **Deploy and run SQL Server Integration Services (SSIS) packages** in Azure.

- ✓ Support for virtual network (VNET) environments.
- ✓ Scale out with on-demand processing power.
- ✓ Support on-demand Spark cluster.
- ✓ Flexible scheduling to support incremental data loads.
- ✓ Triggers for executing data pipelines.

Available APIs

V1 (Nov 2016)

Batch processing of time series data.

- AZ Portal
- Copy Wizard
- Visual Studio
- Azure PowerShell
- Azure Resource Manager template
- REST API
- .NET API

V2 (Sept 2017)

'General-purpose hybrid data integration service'

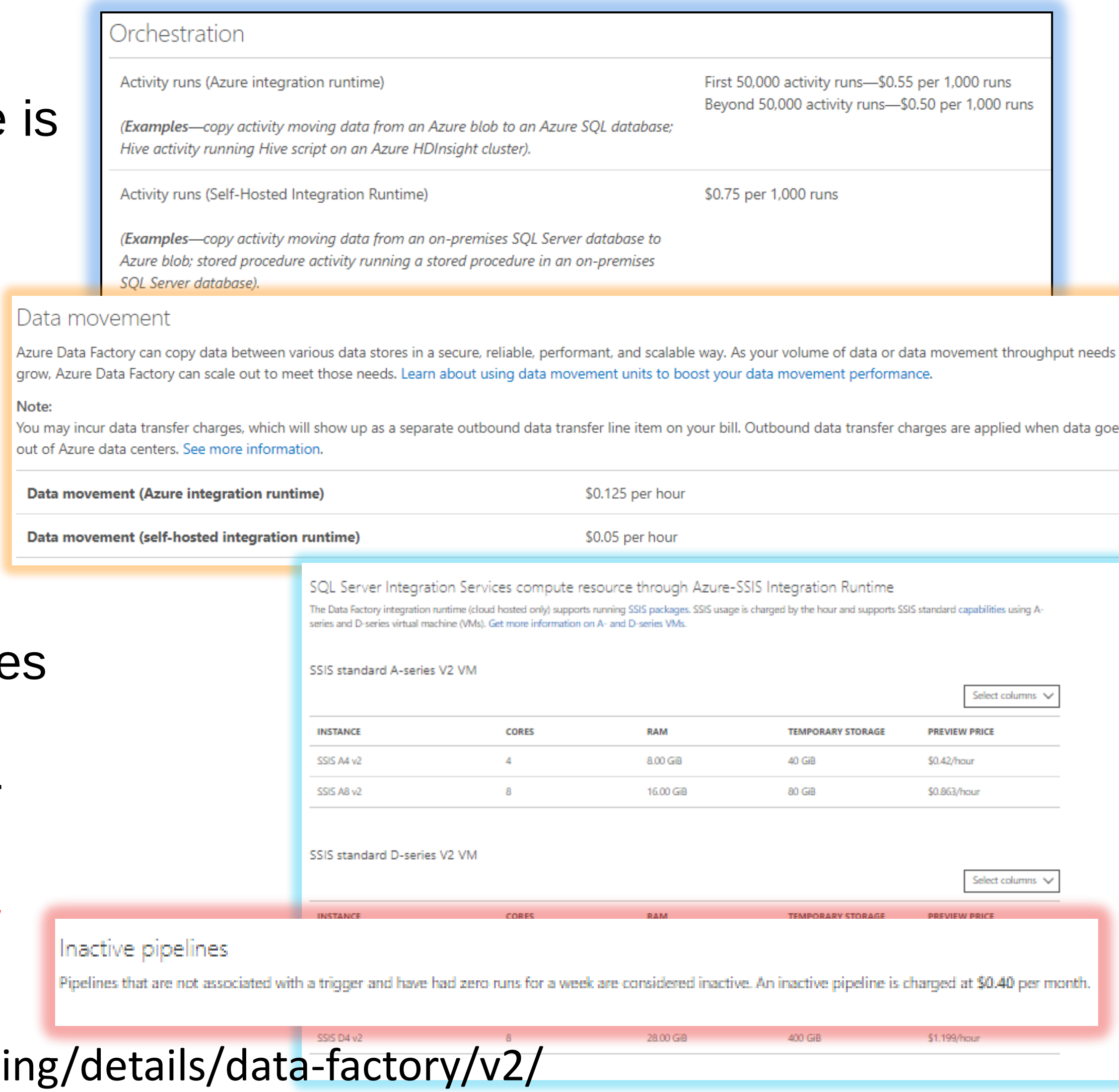
- AZ Portal (note: limited capabilities)
- Azure PowerShell
- Languages:

  .NET & Python
- REST API

Data Factory V2 Costs

The pricing for Data Factory usage is calculated based on the following factors:

1. Number of activities run.
   Orchestration of activities
2. Volume of data moved.
   Data Movement
3. SQL Server Integration Services (SSIS) compute hours.
4. Whether a pipeline is active or not.
   Inactive Pipelines are charged!



Orchestration

Activity runs (Azure integration runtime) *(**Examples**—copy activity moving data from an Azure blob to an Azure SQL database; Hive activity running Hive script on an Azure HDInsight cluster).*	First 50,000 activity runs—$0.55 per 1,000 runs Beyond 50,000 activity runs—$0.50 per 1,000 runs
Activity runs (Self-Hosted Integration Runtime) *(**Examples**—copy activity moving data from an on-premises SQL Server database to Azure blob; stored procedure activity running a stored procedure in an on-premises SQL Server database).*	$0.75 per 1,000 runs

Data movement

Azure Data Factory can copy data between various data stores in a secure, reliable, performant, and scalable way. As your volume of data or data movement throughput needs grow, Azure Data Factory can scale out to meet those needs. Learn about using data movement units to boost your data movement performance.

Note:
You may incur data transfer charges, which will show up as a separate outbound data transfer line item on your bill. Outbound data transfer charges are applied when data goe out of Azure data centers. See more information.

Data movement (Azure integration runtime)	$0.125 per hour
Data movement (self-hosted integration runtime)	$0.05 per hour

SQL Server Integration Services compute resource through Azure-SSIS Integration Runtime

The Data Factory integration runtime (cloud hosted only) supports running SSIS packages. SSIS usage is charged by the hour and supports SSIS standard capabilities using A-series and D-series virtual machine (VMs). Get more information on A- and D-series VMs.

SSIS standard A-series V2 VM

INSTANCE	CORES	RAM	TEMPORARY STORAGE	PREVIEW PRICE
SSIS A4 v2	4	8.00 GiB	40 GiB	$0.42/hour
SSIS A8 v2	8	16.00 GiB	80 GiB	$0.863/hour

SSIS standard D-series V2 VM

INSTANCE	CORES	RAM	TEMPORARY STORAGE	PREVIEW PRICE
SSIS D4 v2	8	28.00 GiB	400 GiB	$1.199/hour

Inactive pipelines

Pipelines that are not associated with a trigger and have had zero runs for a week are considered inactive. An inactive pipeline is charged at $0.40 per month.

https://azure.microsoft.com/en-us/pricing/details/data-factory/v2/

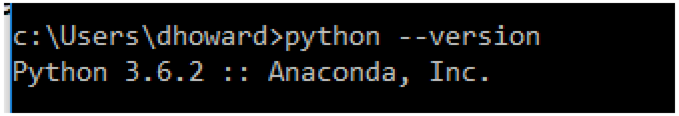
Prerequisites to Create a Data Factory V2

Python 2.7, 3.3, 3.4, 3.5 or 3.6

- Install Python SDK for Azure packages
    - Azure Management Resources
    - Data Factory
- Portal: Obtain your Subscription ID
- Portal: Create an Azure Storage Account & Blob container
- Portal: Create a data file and upload data file to Blob container
- Portal: Create an app in Active Directory (Client ID, Client Secret).
- Python: Create a Data Factory, Linked Service, Pipeline

My Development Environment

- Windows 10
- Python 3.6.2
- Anaconda



```
c:\Users\dhoward>python --version
Python 3.6.2 :: Anaconda, Inc.
```

ANACONDA.

Windows

Windows 10

Home

Python 3.6.2

Release Date: 2017-07-17

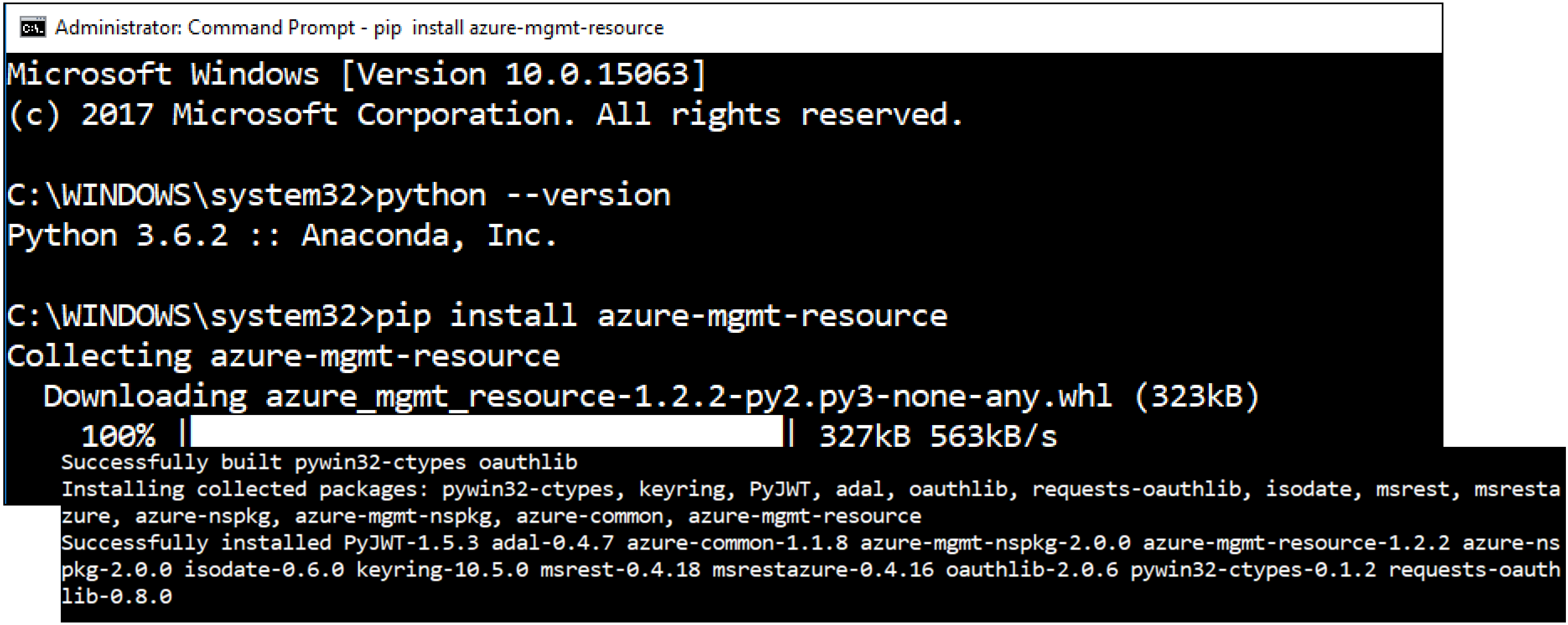
Install Python Package Azure Management Resources

Python SDK for Data Factory (supports Python 2.7, 3.3, 3.4, 3.5 and 3.6)

1. Open Command Prompt as Administrator
2. Install Python package for Azure Management Resources

   pip install azure-mgmt-resource
3. Install Python Package for Azure Data Factory

   pip install azure-mgmt-datafactory



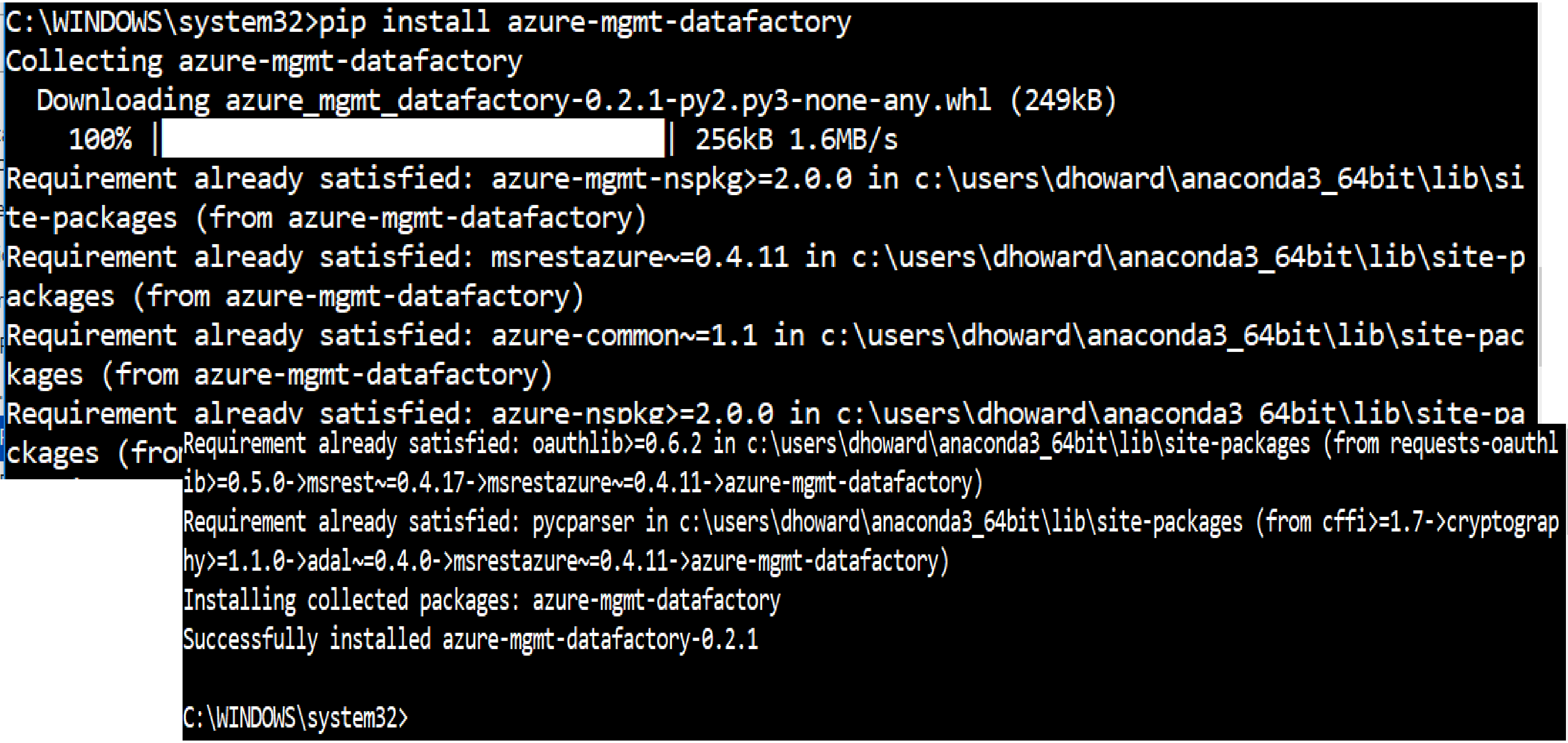
Administrator: Command Prompt - pip install azure-mgmt-resource

```
Microsoft Windows [Version 10.0.15063]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>python --version
Python 3.6.2 :: Anaconda, Inc.

C:\WINDOWS\system32>pip install azure-mgmt-resource
Collecting azure-mgmt-resource
  Downloading azure_mgmt_resource-1.2.2-py2.py3-none-any.whl (323kB)
    100% |████████████████████████████████| 327kB 563kB/s
```

```
Successfully built pywin32-ctypes oauthlib
Installing collected packages: pywin32-ctypes, keyring, PyJWT, adal, oauthlib, requests-oauthlib, isodate, msrest, msresta
zure, azure-nspkg, azure-mgmt-nspkg, azure-common, azure-mgmt-resource
Successfully installed PyJWT-1.5.3 adal-0.4.7 azure-common-1.1.8 azure-mgmt-nspkg-2.0.0 azure-mgmt-resource-1.2.2 azure-ns
pkg-2.0.0 isodate-0.6.0 keyring-10.5.0 msrest-0.4.18 msrestazure-0.4.16 oauthlib-2.0.6 pywin32-ctypes-0.1.2 requests-oauth
lib-0.8.0
```

Install Python Package Data Factory

- Open Command Prompt as Administrator
- Install Python Package for Azure Data Factory

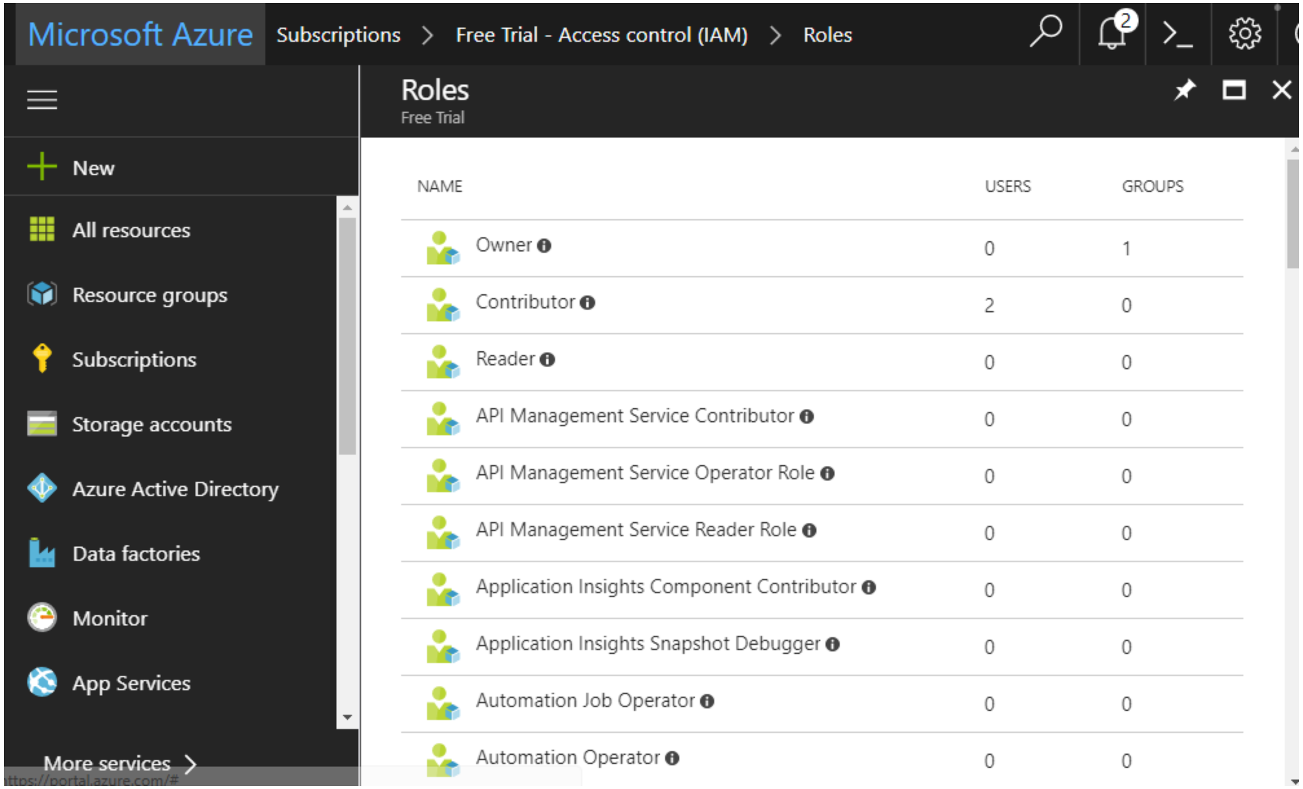


```
C:\WINDOWS\system32>pip install azure-mgmt-datafactory
Collecting azure-mgmt-datafactory
  Downloading azure_mgmt_datafactory-0.2.1-py2.py3-none-any.whl (249kB)
    100% |████████████████████████████████| 256kB 1.6MB/s
Requirement already satisfied: azure-mgmt-nspkg>=2.0.0 in c:\users\dhoward\anaconda3_64bit\lib\si
te-packages (from azure-mgmt-datafactory)
Requirement already satisfied: msrestazure~=0.4.11 in c:\users\dhoward\anaconda3_64bit\lib\site-p
ackages (from azure-mgmt-datafactory)
Requirement already satisfied: azure-common~=1.1 in c:\users\dhoward\anaconda3_64bit\lib\site-pac
kages (from azure-mgmt-datafactory)
Requirement already satisfied: azure-nspkg>=2.0.0 in c:\users\dhoward\anaconda3_64bit\lib\site-pa
ckages (fro
```

```
Requirement already satisfied: oauthlib>=0.6.2 in c:\users\dhoward\anaconda3_64bit\lib\site-packages (from requests-oauthl
ib>=0.5.0->msrest~=0.4.17->msrestazure~=0.4.11->azure-mgmt-datafactory)
Requirement already satisfied: pycparser in c:\users\dhoward\anaconda3_64bit\lib\site-packages (from cffi>=1.7->cryptograp
hy>=1.1.0->adal~=0.4.0->msrestazure~=0.4.11->azure-mgmt-datafactory)
Installing collected packages: azure-mgmt-datafactory
Successfully installed azure-mgmt-datafactory-0.2.1

C:\WINDOWS\system32>
```

Check your Roles in Your Subscription

- You should be a Contributor Role. If not someone within McKesson Admin should set users up as Contributors.



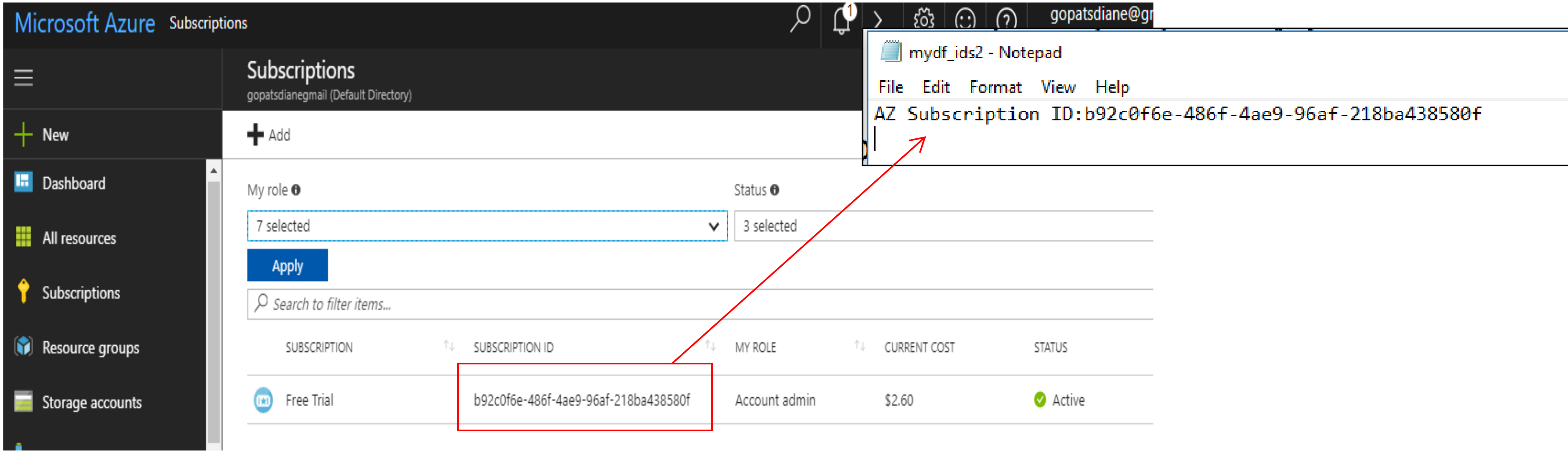
Roles

Free Trial

NAME	USERS	GROUPS
Owner	0	1
Contributor	2	0
Reader	0	0
API Management Service Contributor	0	0
API Management Service Operator Role	0	0
API Management Service Reader Role	0	0
Application Insights Component Contributor	0	0
Application Insights Snapshot Debugger	0	0
Automation Job Operator	0	0
Automation Operator	0	0

Obtain your Subscription ID

- Uniquely identifies your subscription to use AZ services.
- In the left navigation panel, click Subscriptions. Copy your subscription ID.



Subscriptions
gopatsdianegmail (Default Directory)

My role: 7 selected
Status: 3 selected

SUBSCRIPTION	SUBSCRIPTION ID	MY ROLE	CURRENT COST	STATUS
Free Trial	b92c0f6e-486f-4ae9-96af-218ba438580f	Account admin	$2.60	Active

```
AZ Subscription ID:b92c0f6e-486f-4ae9-96af-218ba438580f
```

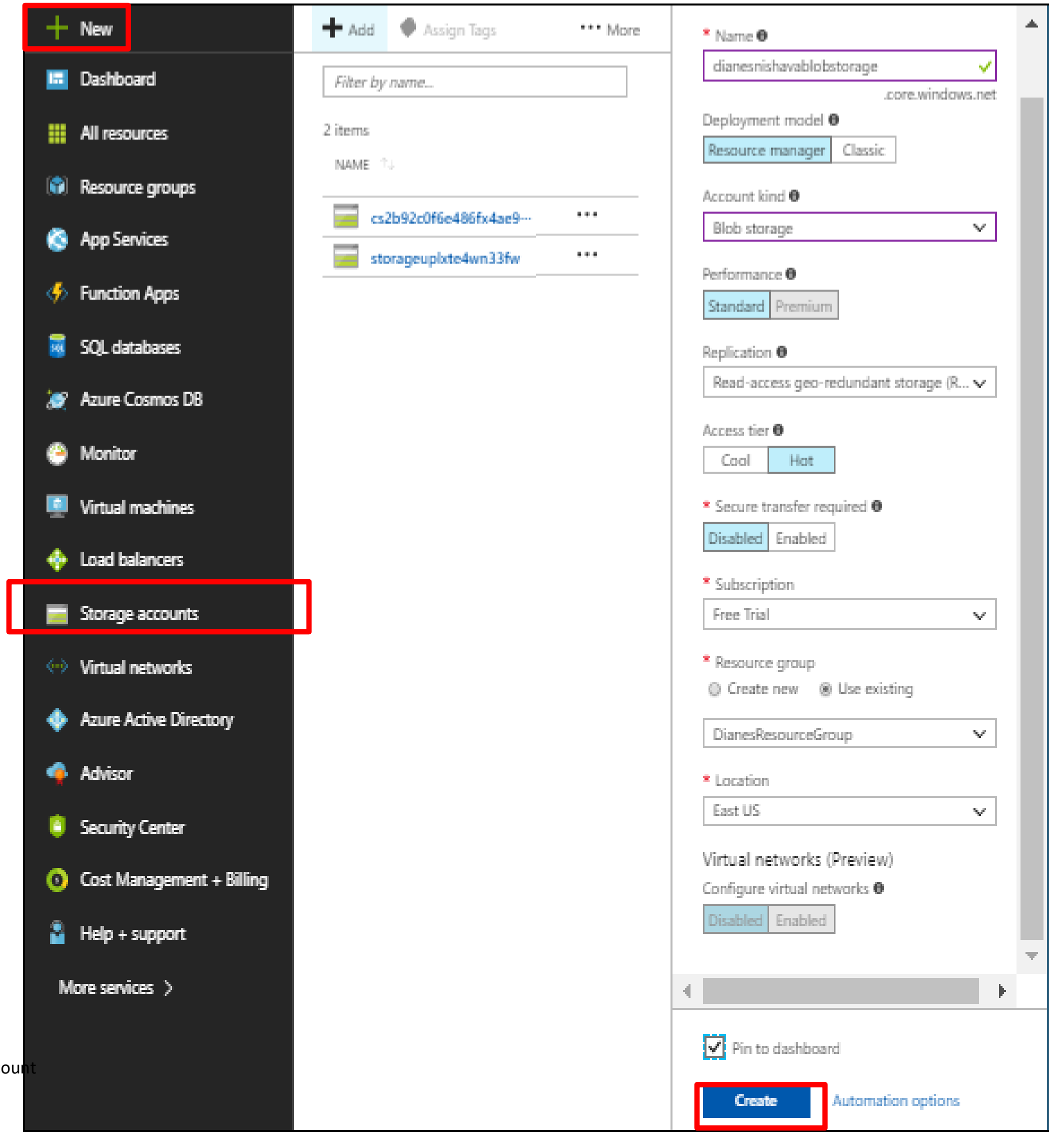
Create a Blob Storage Account

Select:

1. Name: Must be unique within Azure and lowercase.
2. Deployment model: Resource Mgr
3. Account Kind: Blob or Gen Purpose
4. Use default for Replication. (Other options changes price.)
5. Create new resource group or use an existing one.
6. Location: select any location close to where you are located. (I use East US).
7. Virtual Networks (Preview): Disabled

Optional: Pin to Dashboard

https://docs.microsoft.com/en-us/azure/storage/common/storage-create-storage-account#create-a-storage-account



2 items

cs2b92c0f6e486fx4ae9··· •••

storageuplxte4wn33fw •••

* Name: dianesnishavablobstorage ✓ .core.windows.net

Deployment model: Resource manager | Classic

Account kind: Blob storage

Performance: Standard | Premium

Replication: Read-access geo-redundant storage (R...

Access tier: Cool | Hot

* Secure transfer required: Disabled | Enabled

* Subscription: Free Trial

* Resource group: ○ Create new ● Use existing

DianesResourceGroup

* Location: East US

Virtual networks (Preview)

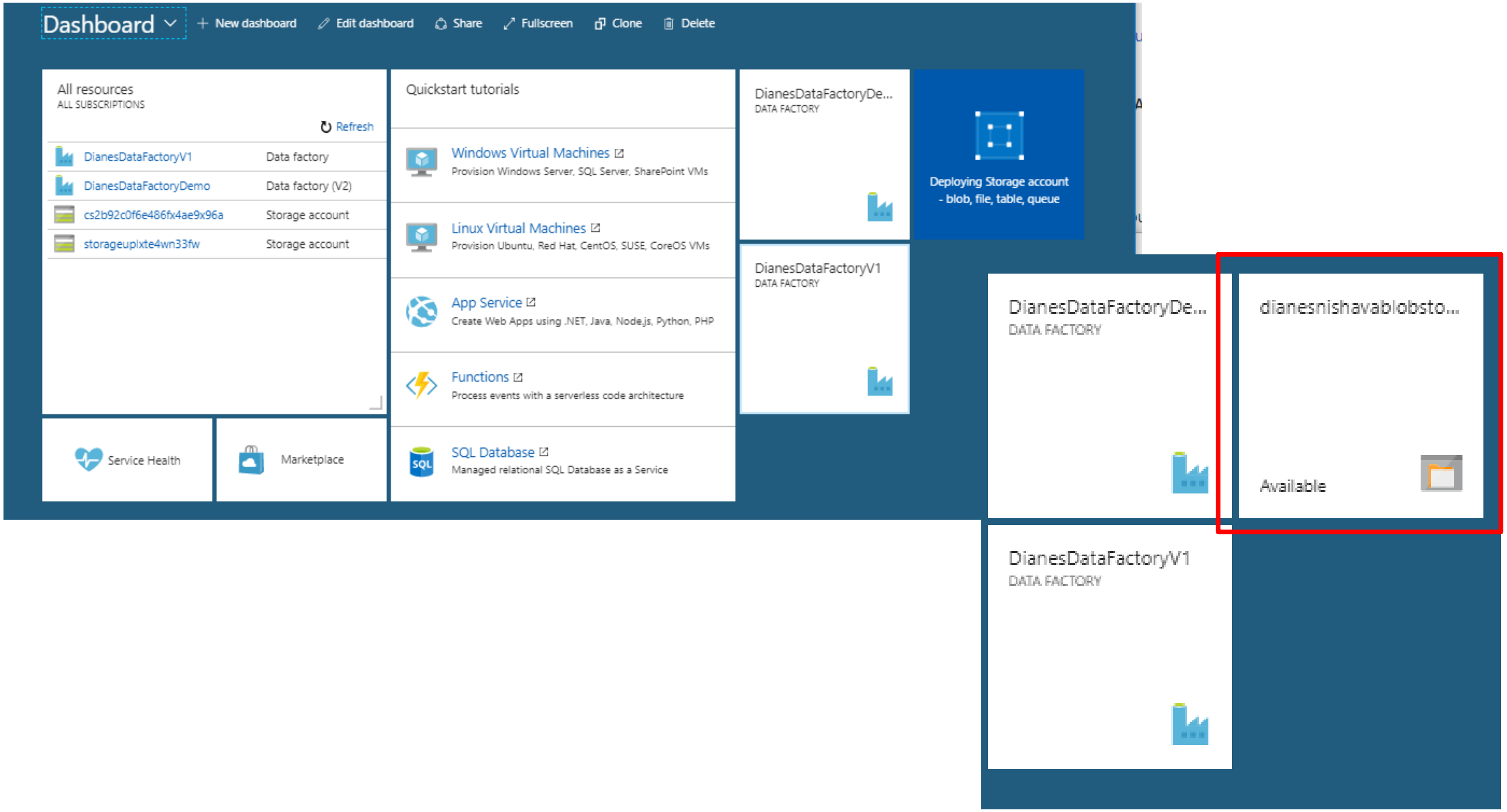
Configure virtual networks: Disabled | Enabled

☑ Pin to dashboard

Create

Automation options

Wait for your Deployment



All resources
ALL SUBSCRIPTIONS

Refresh

DianesDataFactoryV1	Data factory
DianesDataFactoryDemo	Data factory (V2)
cs2b92c0f6e486fx4ae9x96a	Storage account
storageuplxte4wn33fw	Storage account

Quickstart tutorials

Windows Virtual Machines
Provision Windows Server, SQL Server, SharePoint VMs

Linux Virtual Machines
Provision Ubuntu, Red Hat, CentOS, SUSE, CoreOS VMs

App Service
Create Web Apps using .NET, Java, Node.js, Python, PHP

Functions
Process events with a serverless code architecture

SQL Database
Managed relational SQL Database as a Service

DianesDataFactoryDe...
DATA FACTORY

Deploying Storage account - blob, file, table, queue

DianesDataFactoryV1
DATA FACTORY

Service Health

Marketplace

DianesDataFactoryDe...
DATA FACTORY

dianesnishavablobsto...

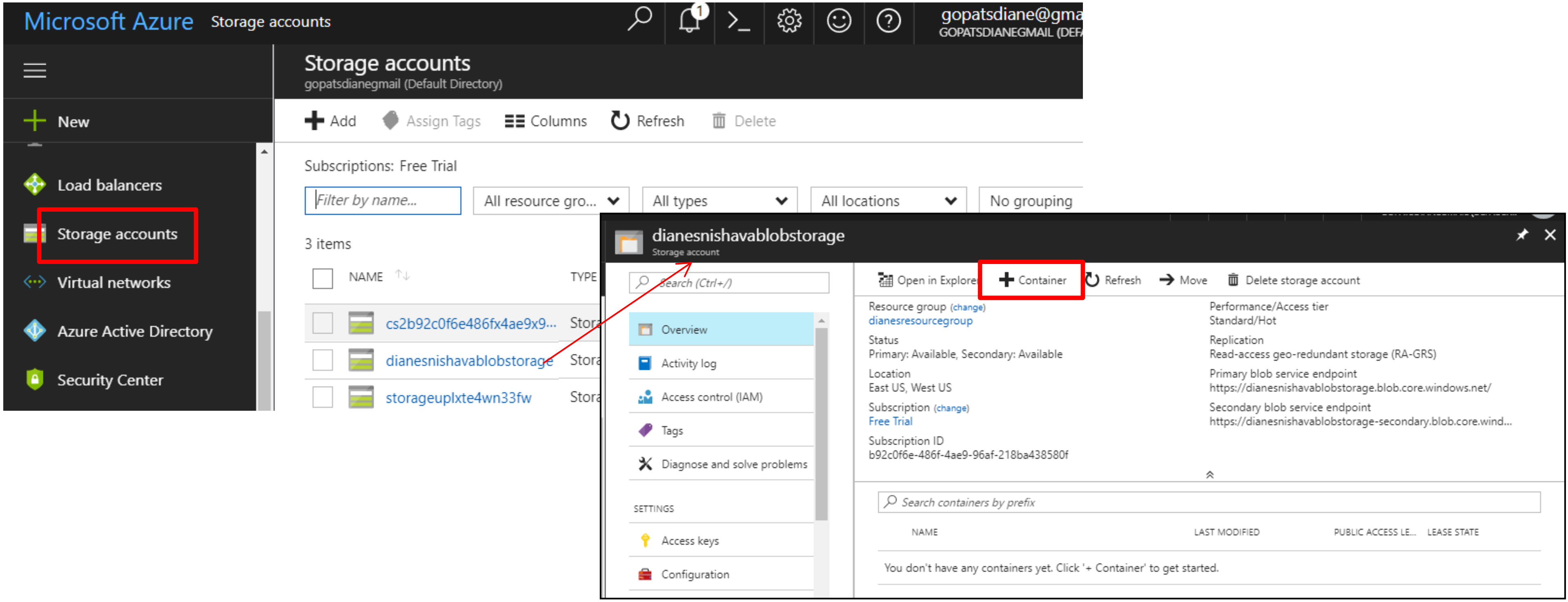
Available

DianesDataFactoryV1
DATA FACTORY

@Diane Howard, Nishava Inc.

Create a Container for your Blob Storage

- Stores unlimited number of blobs.
- Container name must be lowercase.
- Blob: A file of any type and size.
- **Azure Storage** offers three types of blobs: block blobs, page blobs & append blobs.



Storage accounts
gopatsdianegmail (Default Directory)

Subscriptions: Free Trial

Filter by name...

3 items

NAME	TYPE
cs2b92c0f6e486fx4ae9x9...	Stor
dianesnishavablobstorage	Stora
storageuplxte4wn33fw	Stora

dianesnishavablobstorage
Storage account

Resource group (change)
dianesresourcegroup

Status
Primary: Available, Secondary: Available

Location
East US, West US

Subscription (change)
Free Trial

Subscription ID
b92c0f6e-486f-4ae9-96af-218ba438580f

Performance/Access tier
Standard/Hot

Replication
Read-access geo-redundant storage (RA-GRS)

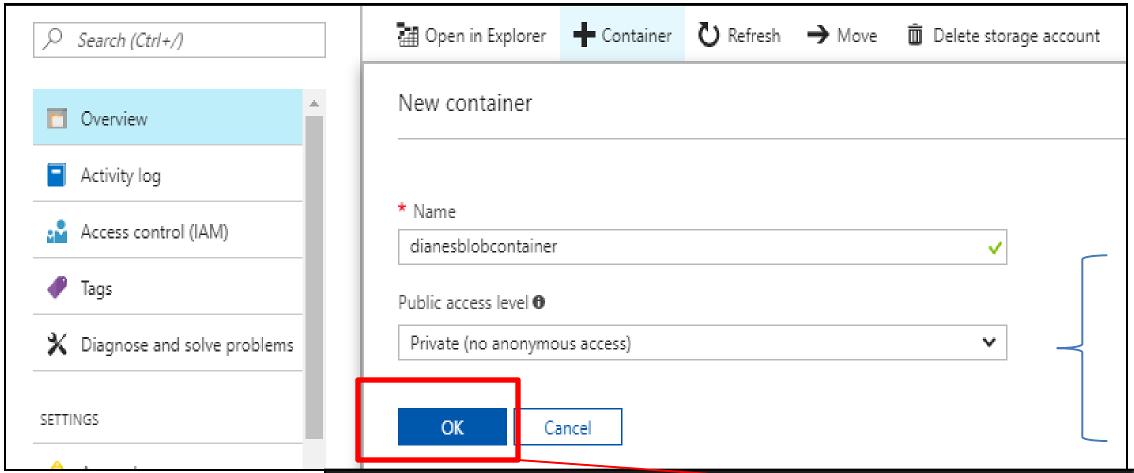
Primary blob service endpoint
https://dianesnishavablobstorage.blob.core.windows.net/

Secondary blob service endpoint
https://dianesnishavablobstorage-secondary.blob.core.wind...

You don't have any containers yet. Click '+ Container' to get started.

@Diane Howard, Nishava Inc.

Create Storage Container



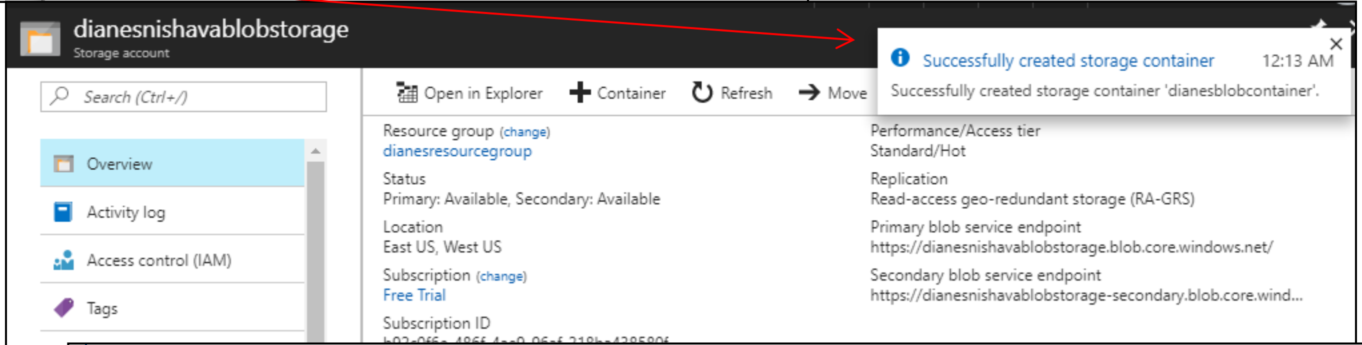
New container

* Name: dianesblobcontainer ✓

Public access level: Private (no anonymous access)

OK | Cancel

Note: Keys are automatically generated when you create your storage.



Successfully created storage container 12:13 AM

Successfully created storage container 'dianesblobcontainer'.

Resource group (change)
dianesresourcegroup

Status
Primary: Available, Secondary: Available

Location
East US, West US

Subscription (change)
Free Trial

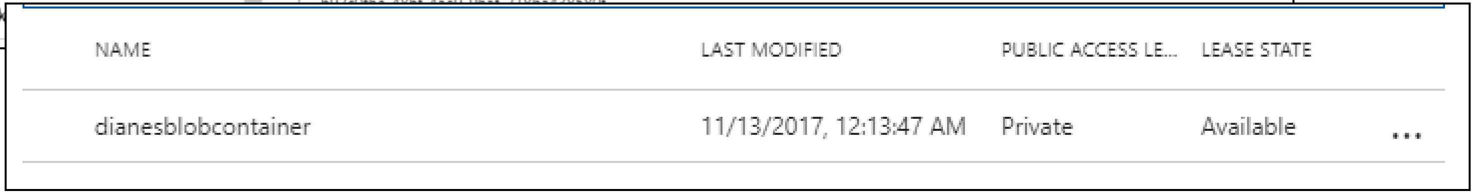
Subscription ID

Performance/Access tier
Standard/Hot

Replication
Read-access geo-redundant storage (RA-GRS)

Primary blob service endpoint
https://dianesnishavablobstorage.blob.core.windows.net/

Secondary blob service endpoint
https://dianesnishavablobstorage-secondary.blob.core.wind...

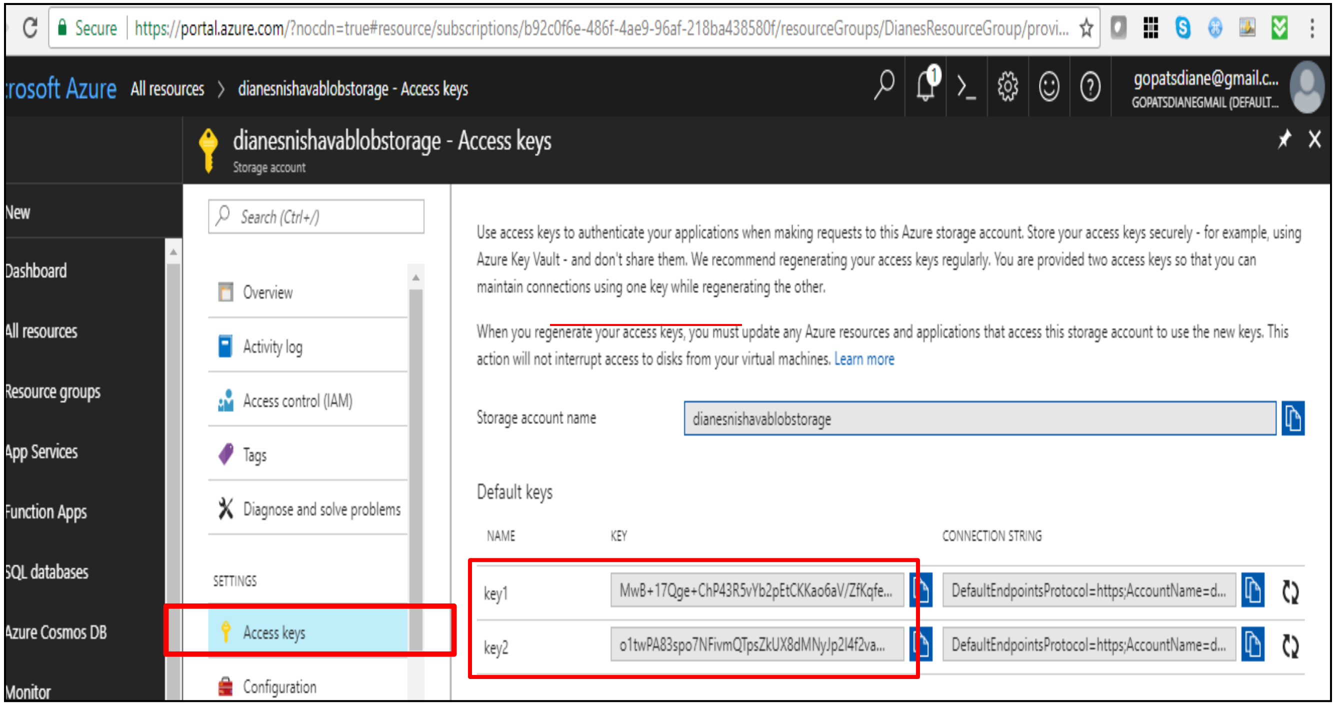


NAME	LAST MODIFIED	PUBLIC ACCESS LE...	LEASE STATE	
dianesblobcontainer	11/13/2017, 12:13:47 AM	Private	Available	...

@Diane Howard, Nishava Inc.

Storage Access Keys

- Two 512-bit storage access keys are created automatically when you create general-purpose or blob storage.
- Used for authentication to access your storage.
- Never share your storage access keys!



dianesnishavablobstorage - Access keys

Storage account

Use access keys to authenticate your applications when making requests to this Azure storage account. Store your access keys securely - for example, using Azure Key Vault - and don't share them. We recommend regenerating your access keys regularly. You are provided two access keys so that you can maintain connections using one key while regenerating the other.

When you regenerate your access keys, you must update any Azure resources and applications that access this storage account to use the new keys. This action will not interrupt access to disks from your virtual machines. Learn more

Storage account name: dianesnishavablobstorage

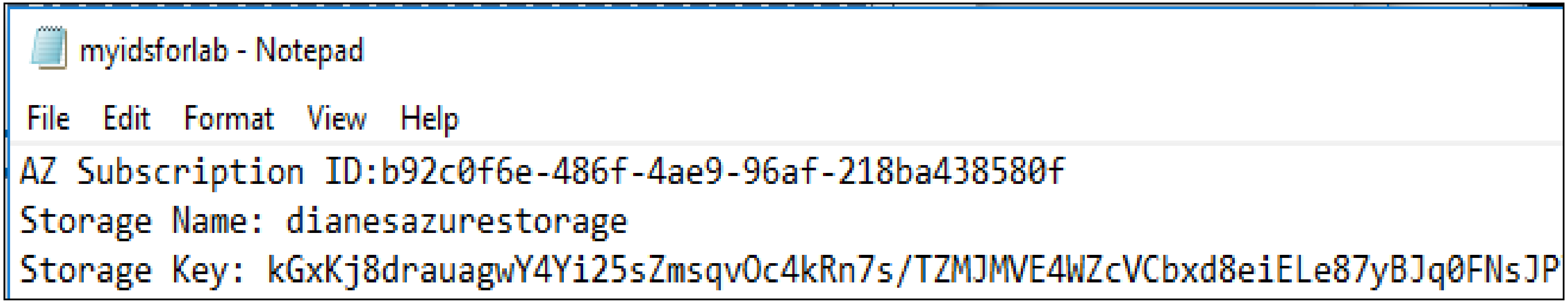
Default keys

NAME	KEY	CONNECTION STRING
key1	MwB+17Qge+ChP43R5vYb2pEtCKKao6aV/ZfKqfe...	DefaultEndpointsProtocol=https;AccountName=d...
key2	o1twPA83spo7NFivmQTpsZkUX8dMNyJp2l4f2va...	DefaultEndpointsProtocol=https;AccountName=d...

@Diane Howard, Nishava Inc.

Save Subscription ID, Storage Info and Key in Notepad

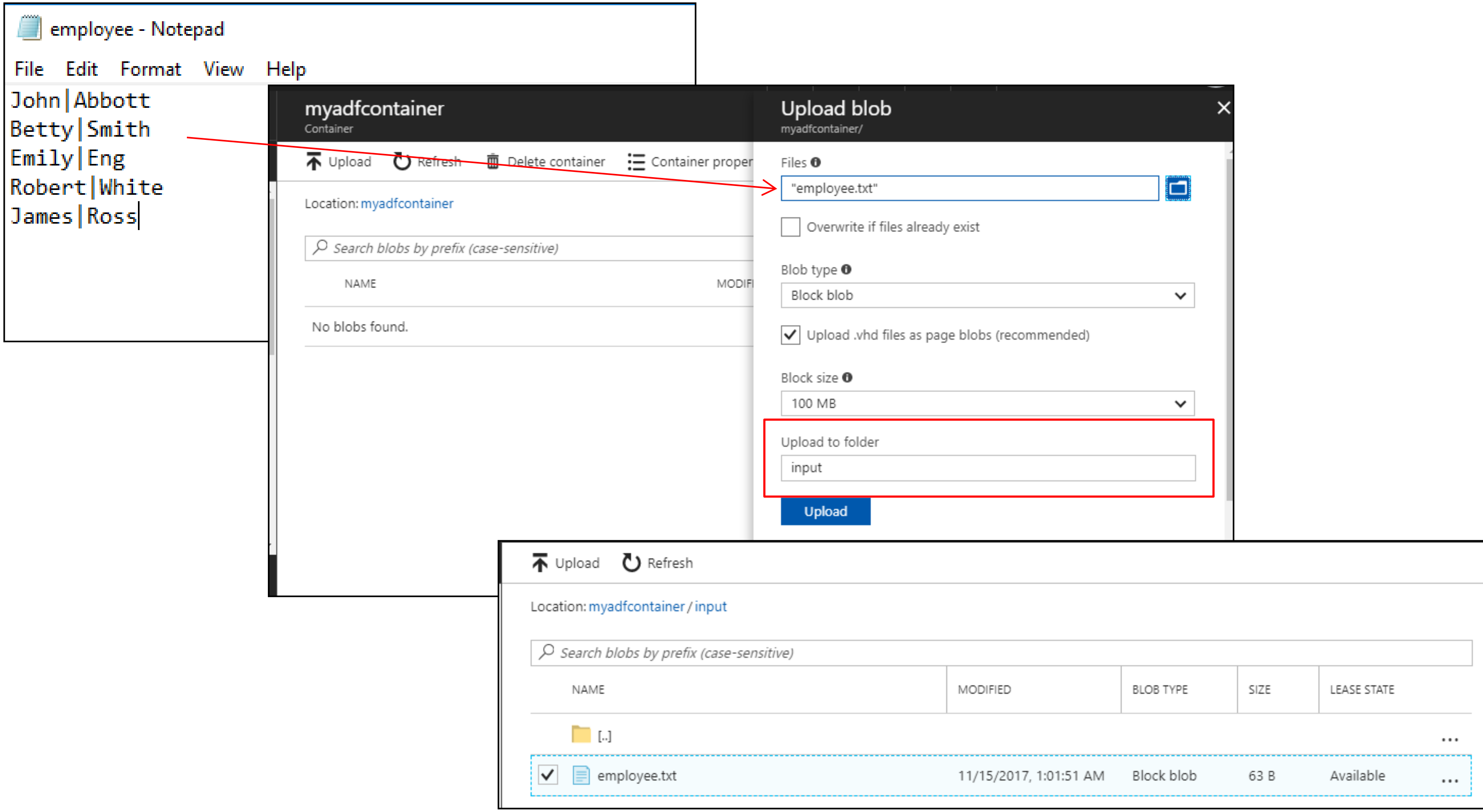
- We will need these later in our Python Code.



```
AZ Subscription ID:b92c0f6e-486f-4ae9-96af-218ba438580f
Storage Name: dianesazurestorage
Storage Key: kGxKj8drauagwY4Yi25sZmsqvOc4kRn7s/TZMJMVE4WZcVCbxd8eiELe87yBJq0FNsJP
```

Upload Data File to new Blob Container

employee.txt



employee - Notepad

```
John|Abbott
Betty|Smith
Emily|Eng
Robert|White
James|Ross
```

myadfcontainer
Container

Location: myadfcontainer

Search blobs by prefix (case-sensitive)

No blobs found.

Upload blob
myadfcontainer/

Files: "employee.txt"

□ Overwrite if files already exist

Blob type: Block blob

☑ Upload .vhd files as page blobs (recommended)

Block size: 100 MB

Upload to folder: input

Location: myadfcontainer / input

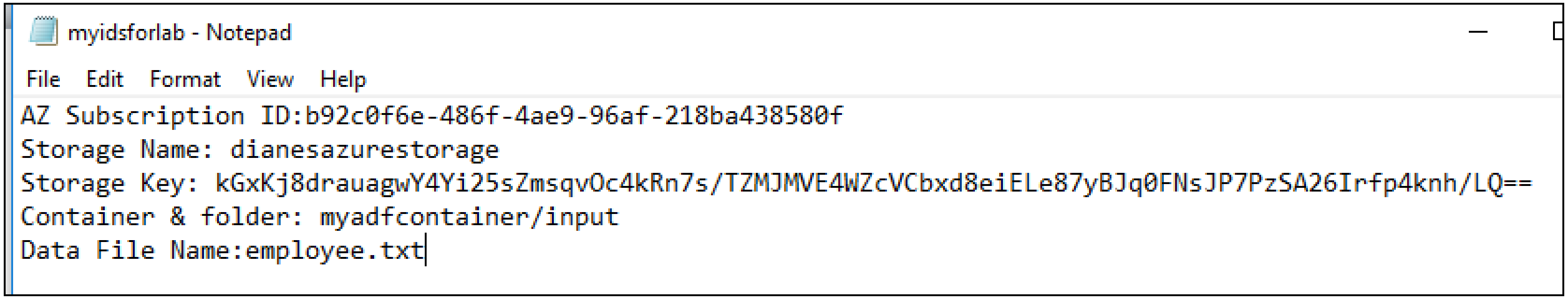
Search blobs by prefix (case-sensitive)

	NAME	MODIFIED	BLOB TYPE	SIZE	LEASE STATE	
	[..]					...
☑	employee.txt	11/15/2017, 1:01:51 AM	Block blob	63 B	Available	...

@Diane Howard, Nishava Inc.

Save Container Name and folder and data file name in Notepad

- We will need these later in our Python Code.



```
AZ Subscription ID:b92c0f6e-486f-4ae9-96af-218ba438580f
Storage Name: dianesazurestorage
Storage Key: kGxKj8drauagwY4Yi25sZmsqvOc4kRn7s/TZMJMVE4WZcVCbxd8eiELe87yBJq0FNsJP7PzSA26Irfp4knh/LQ==
Container & folder: myadfcontainer/input
Data File Name:employee.txt
```

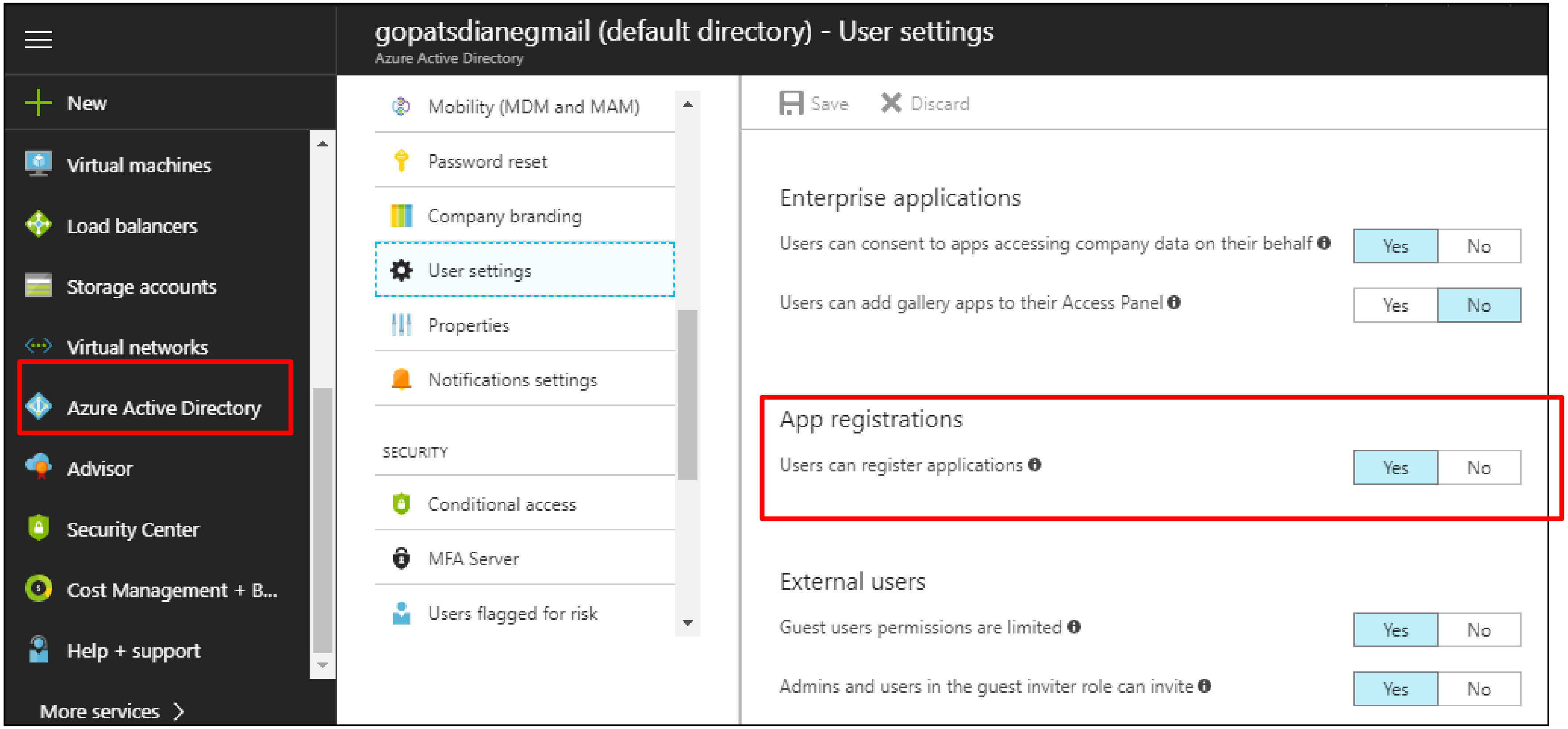
Prerequisites to Create a Data Factory V2

Python 2.7, 3.3, 3.4, 3.5 or 3.6

- ✓ Install Python SDK for Azure packages
    - Azure Management Resources
    - Data Factory
- ✓ Portal: Obtain your Subscription ID
- ✓ Create an Azure Storage Account & Blob container
- ✓ Portal: Create a data file and upload data file to Blob container
- Portal: Create an app in Active Directory (Client ID, Client Secret).
- Python: Create a Data Factory, Linked Service, Pipeline

Check Azure Active Directory Permissions

- Active Directory Permissions allows resources to be created by apps in AZ.
- Go to Active Directory to check if you can register an App.
- Default is set to Yes!



gopatsdianegmail (default directory) - User settings

Azure Active Directory

Enterprise applications

Users can consent to apps accessing company data on their behalf [✓ Yes] No

Users can add gallery apps to their Access Panel Yes [✓ No]

App registrations

Users can register applications [✓ Yes] No

External users

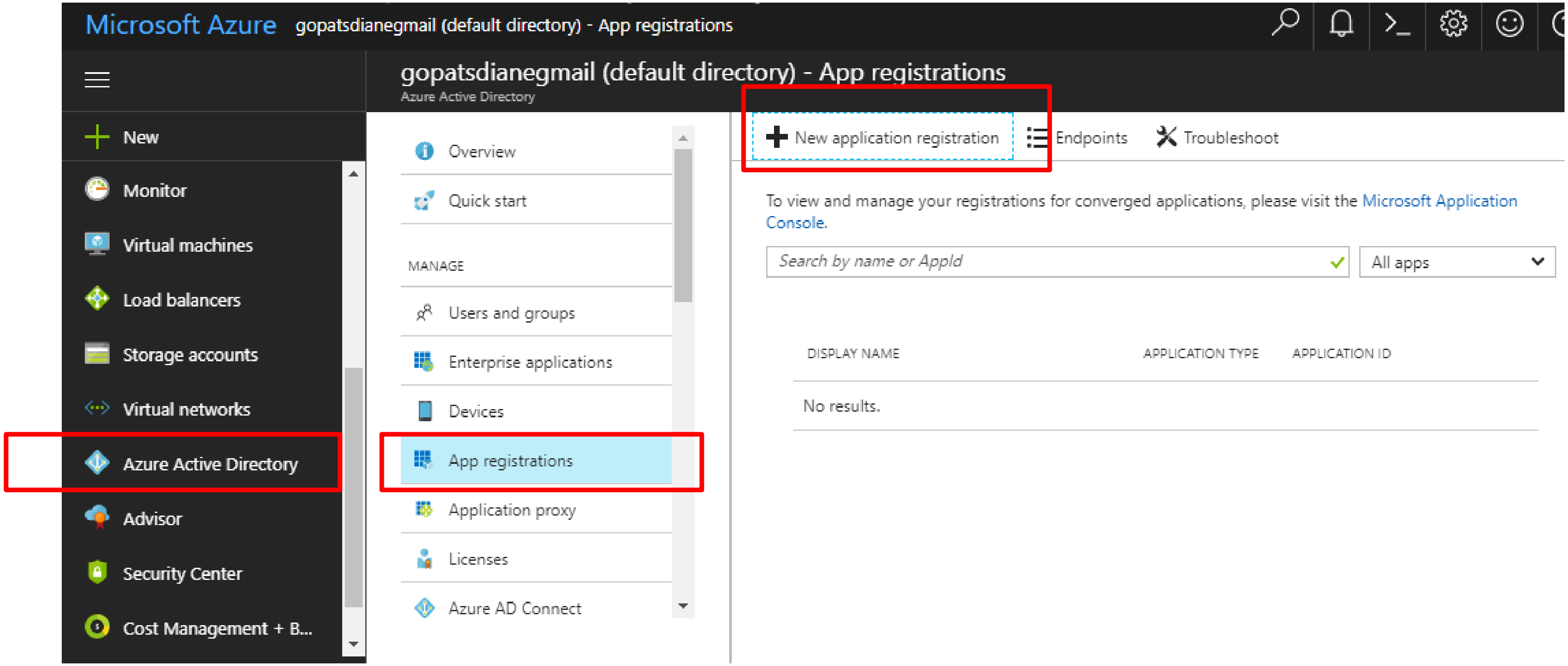
Guest users permissions are limited [✓ Yes] No

Admins and users in the guest inviter role can invite [✓ Yes] No

https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-create-service-principal-portal

Create an Azure Active Directory application

- Set up an Azure Active Directory (AD) application and assign permissions.
- Used by an application that will need to access or modify resources.



gopatsdianegmail (default directory) - App registrations

To view and manage your registrations for converged applications, please visit the Microsoft Application Console.

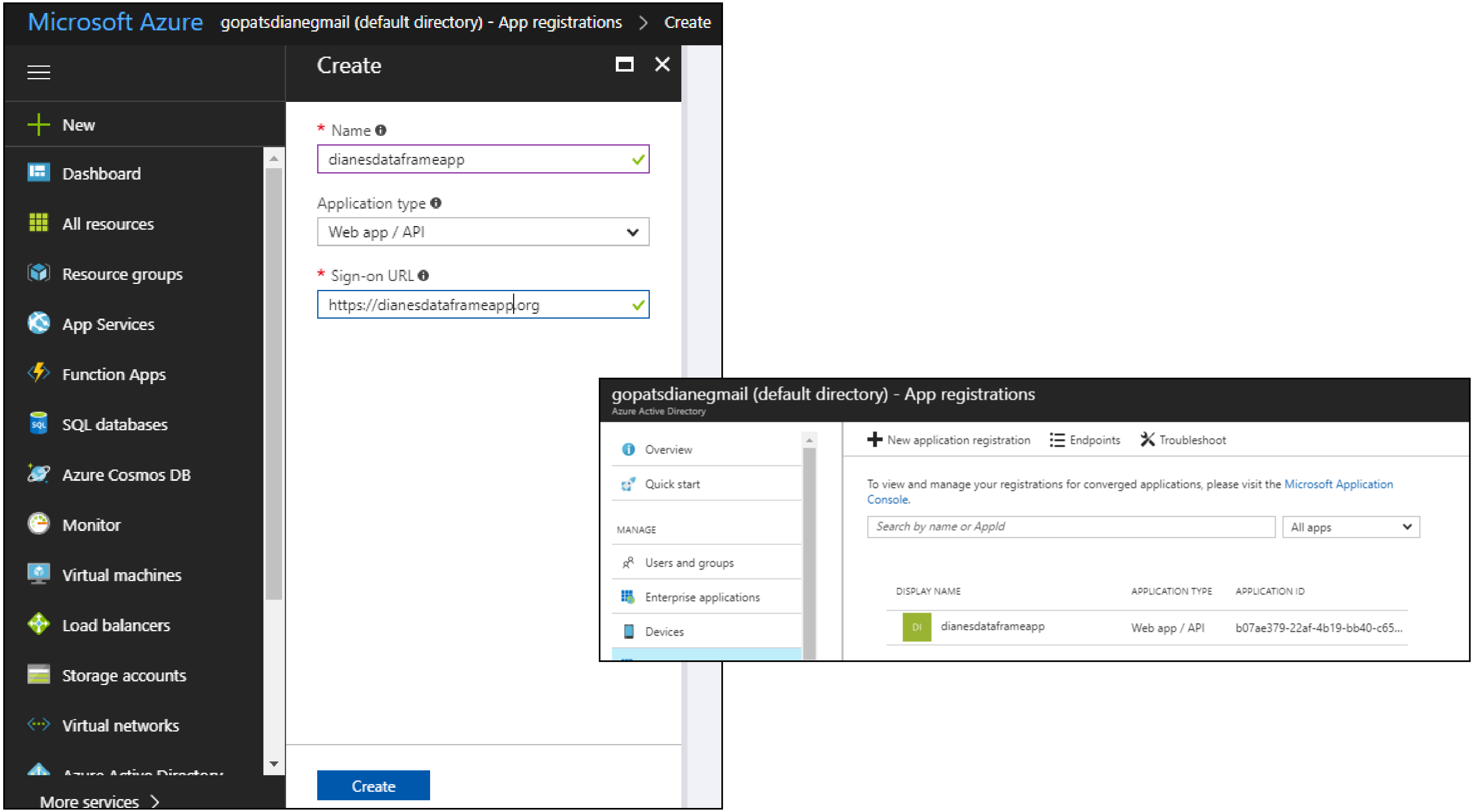
Search by name or AppId

All apps

DISPLAY NAME	APPLICATION TYPE	APPLICATION ID
No results.		

@Diane Howard, Nishava Inc.

Create an Azure Active Directory application



* Name: dianesdataframeapp

Application type: Web app / API

* Sign-on URL: https://dianesdataframeapp.org

gopatsdianegmail (default directory) - App registrations

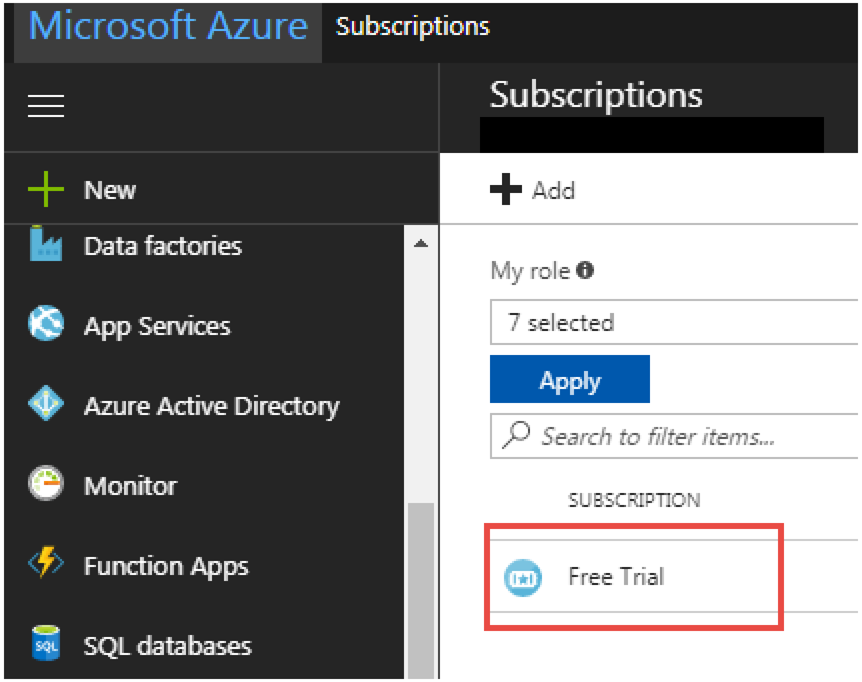
To view and manage your registrations for converged applications, please visit the Microsoft Application Console.

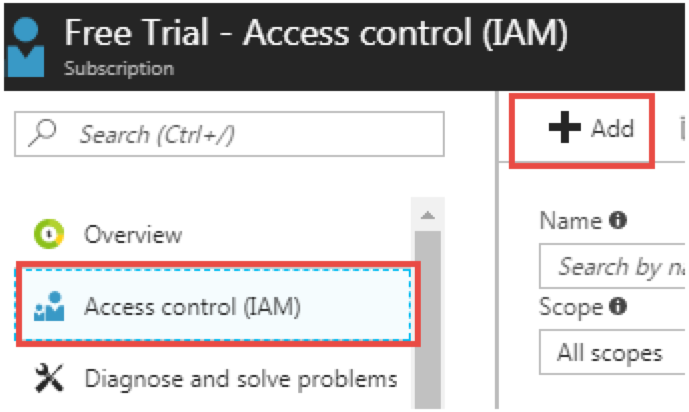
DISPLAY NAME	APPLICATION TYPE	APPLICATION ID
DI dianesdataframeapp	Web app / API	b07ae379-22af-4b19-bb40-c65...

@Diane Howard, Nishava Inc.

Assign a Role to your new App

- Under New -> More Services -> Subscriptions
- Select your Subscription -> Access Control (IAM) -> Add

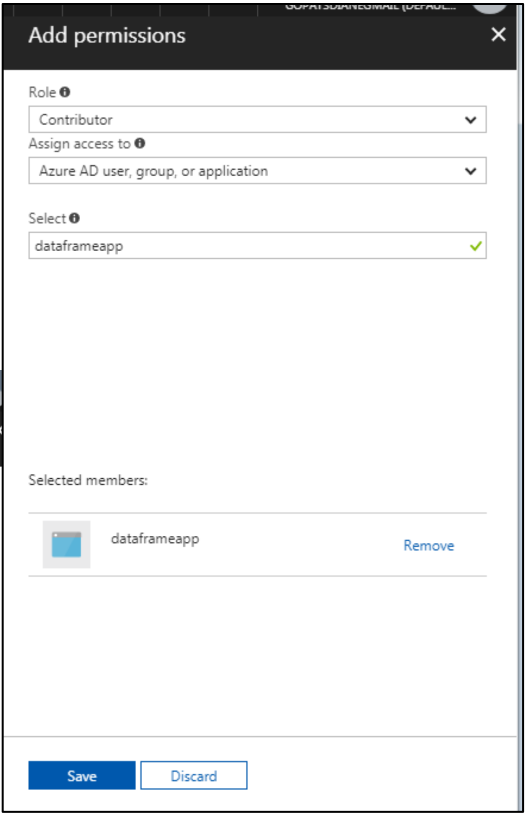




Assign a Role

- Under Role select Contributor. Note:
- You need to have this permission allowed if you are using the McKesson subscription.

- Search for your data frame app, select it and then select Save



Add permissions

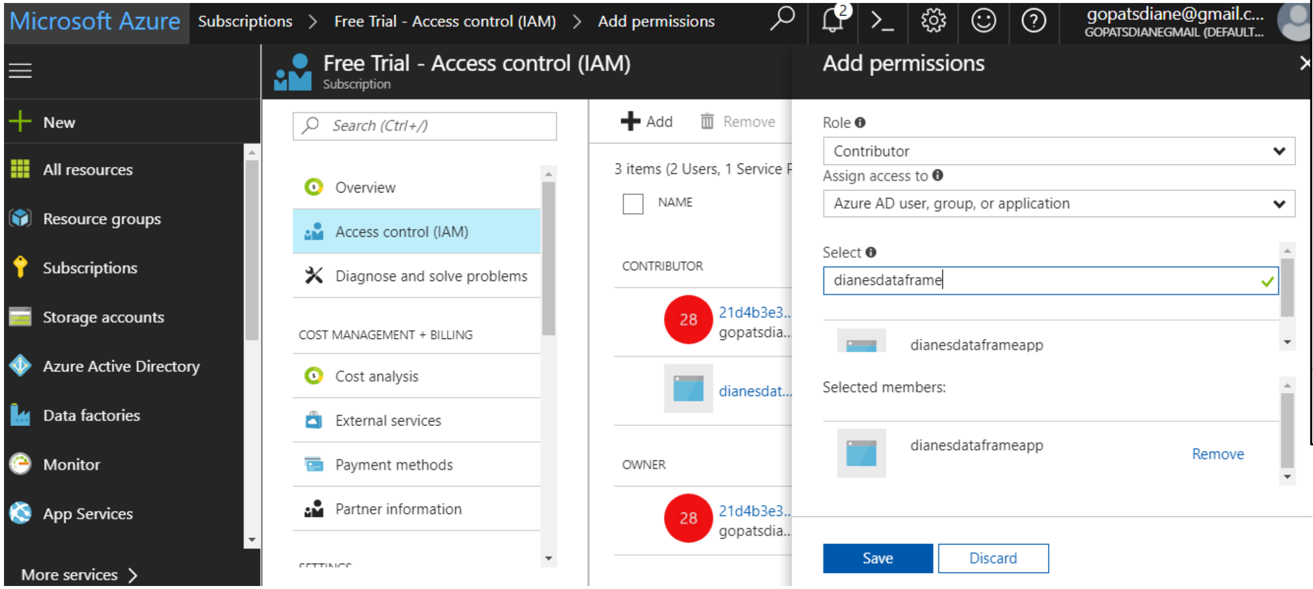
Role: Contributor

Assign access to: Azure AD user, group, or application

Select: dataframeapp

Selected members:

dataframeapp Remove



Microsoft Azure Subscriptions > Free Trial - Access control (IAM) > Add permissions

gopatsdiane@gmail.c... GOPATSDIANEGMAIL (DEFAULT...

Free Trial - Access control (IAM)
Subscription

3 items (2 Users, 1 Service P

NAME
CONTRIBUTOR
21d4b3e3... gopatsdia...
dianesdat...
OWNER
21d4b3e3... gopatsdia...

Add permissions

Role: Contributor

Assign access to: Azure AD user, group, or application

Select: dianesdataframe

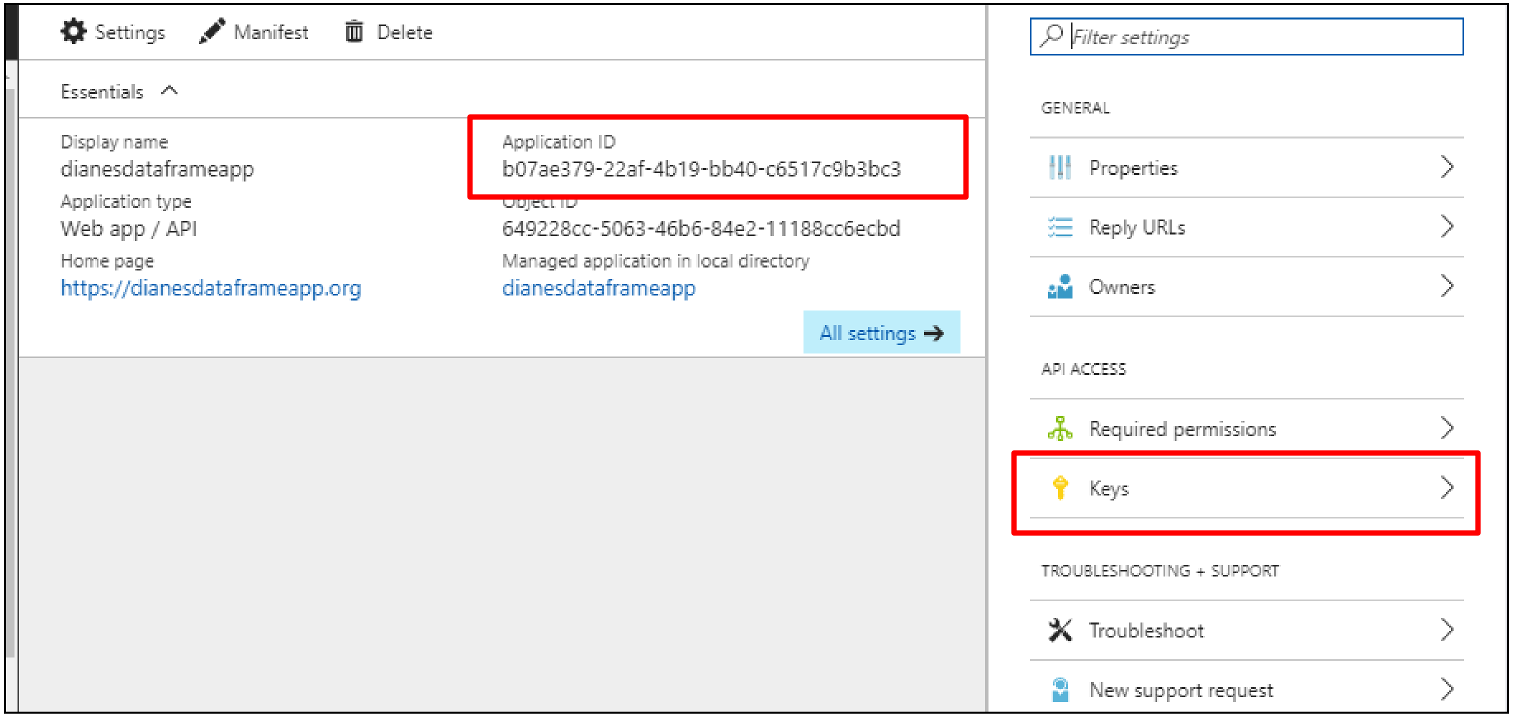
dianesdataframeapp

Selected members:

dianesdataframeapp Remove

Obtain the Application ID

- Need the Application ID for your Python and .NET code
- Save it into Notepad as you will need to add it to your Python or .NET code.



Display name
dianesdataframeapp

Application type
Web app / API

Home page
https://dianesdataframeapp.org

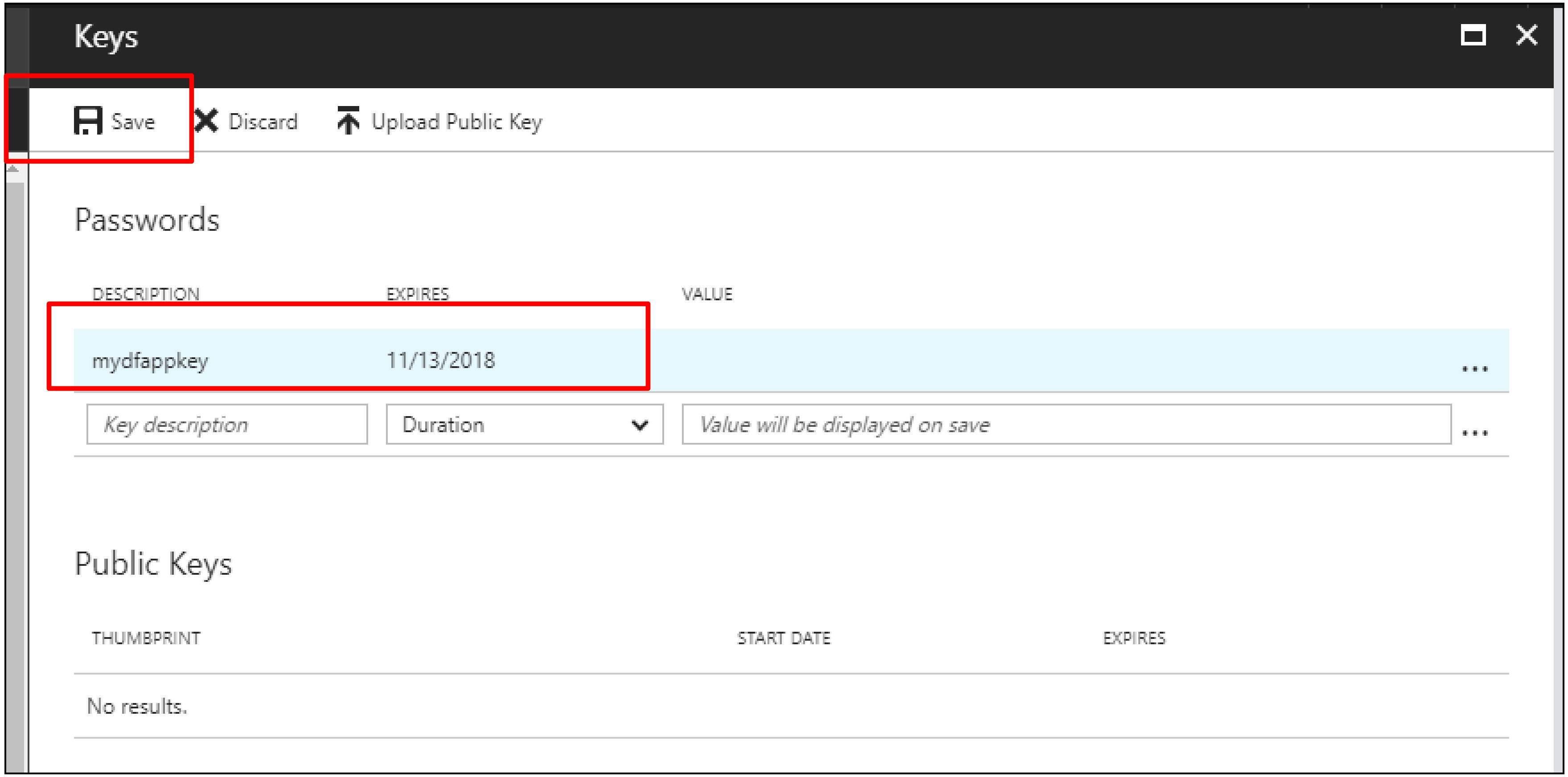
Application ID
b07ae379-22af-4b19-bb40-c6517c9b3bc3

Object ID
649228cc-5063-46b6-84e2-11188cc6ecbd

Managed application in local directory
dianesdataframeapp

Create Authentication (Secret Key)

- Set up your Secret Key



Passwords

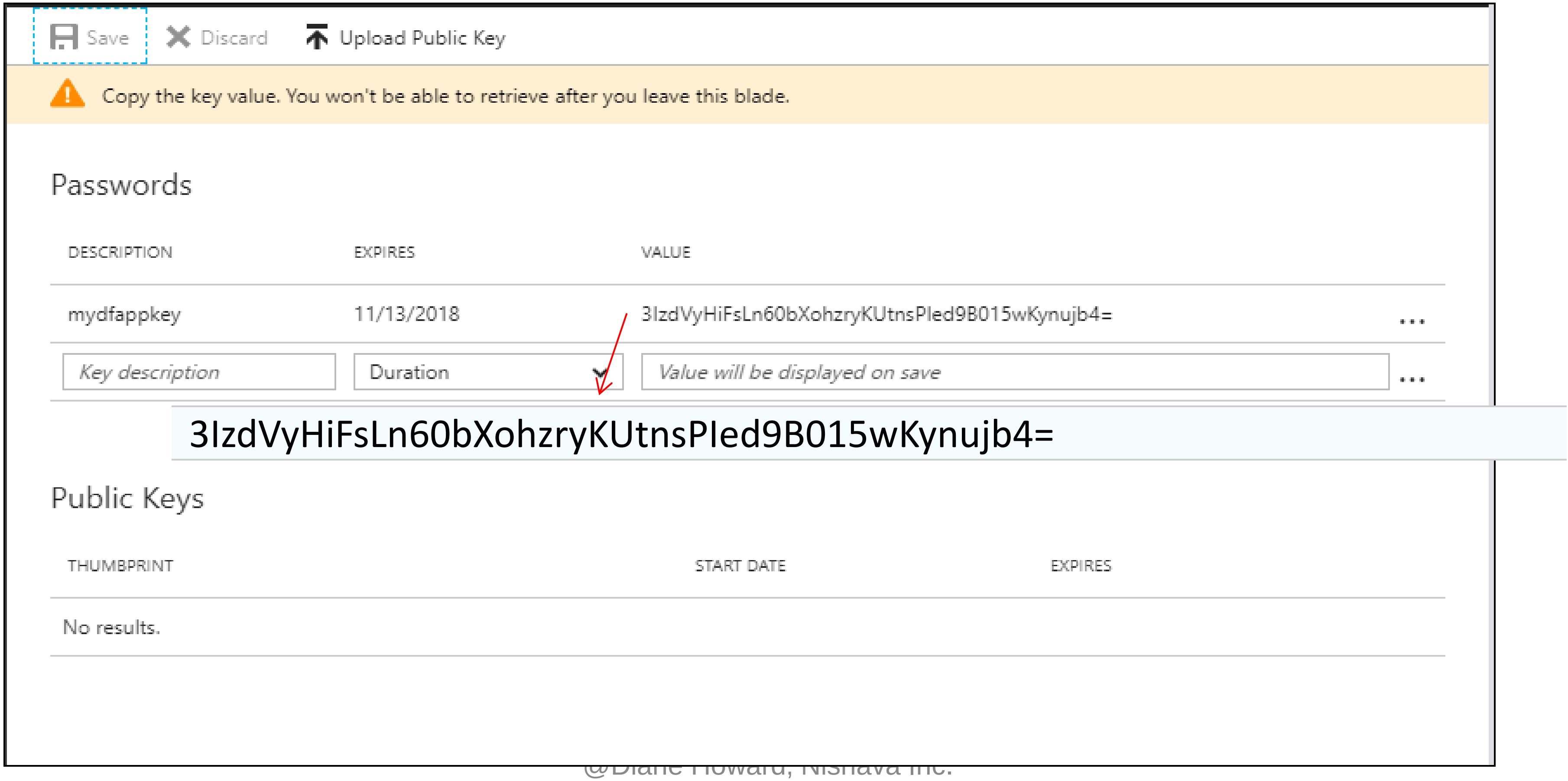
DESCRIPTION	EXPIRES	VALUE
mydfappkey	11/13/2018	
Key description	Duration	Value will be displayed on save

Public Keys

THUMBPRINT	START DATE	EXPIRES
No results.		

Copy your Secret Key Value

- Appears one time only!
- This is also known as your *Tenant ID*.
- Save your Secret Key in Notepad!



Copy the key value. You won't be able to retrieve after you leave this blade.

Passwords

DESCRIPTION	EXPIRES	VALUE
mydfappkey	11/13/2018	3IzdVyHiFsLn60bXohzryKUtnsPIed9B015wKynujb4=
Key description	Duration	Value will be displayed on save

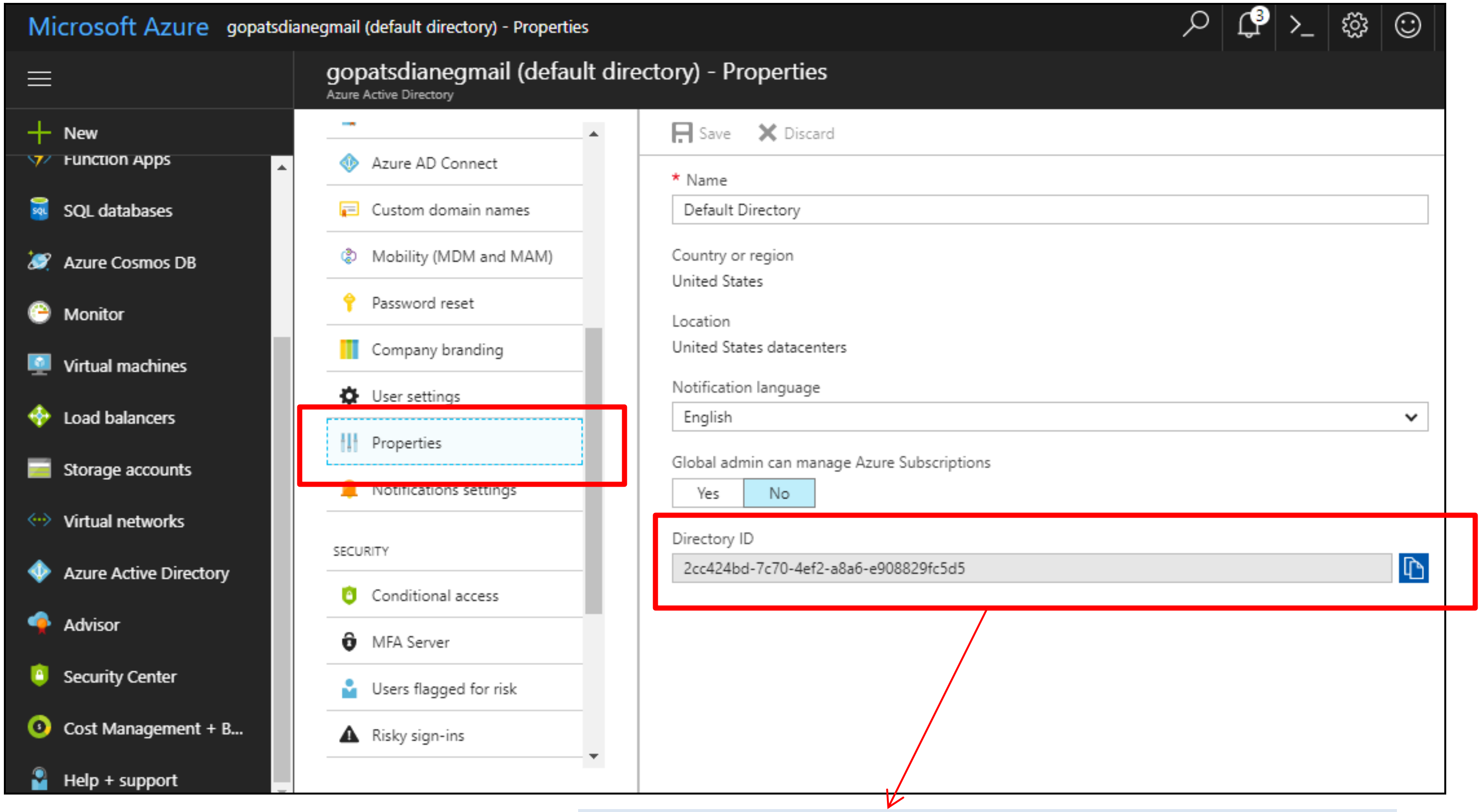
3IzdVyHiFsLn60bXohzryKUtnsPIed9B015wKynujb4=

Public Keys

THUMBPRINT	START DATE	EXPIRES
No results.		

Get your Tenant ID

- The Directory ID = your Tenant ID.
- Copy it to Notepad



* Name: Default Directory

Country or region: United States

Location: United States datacenters

Notification language: English

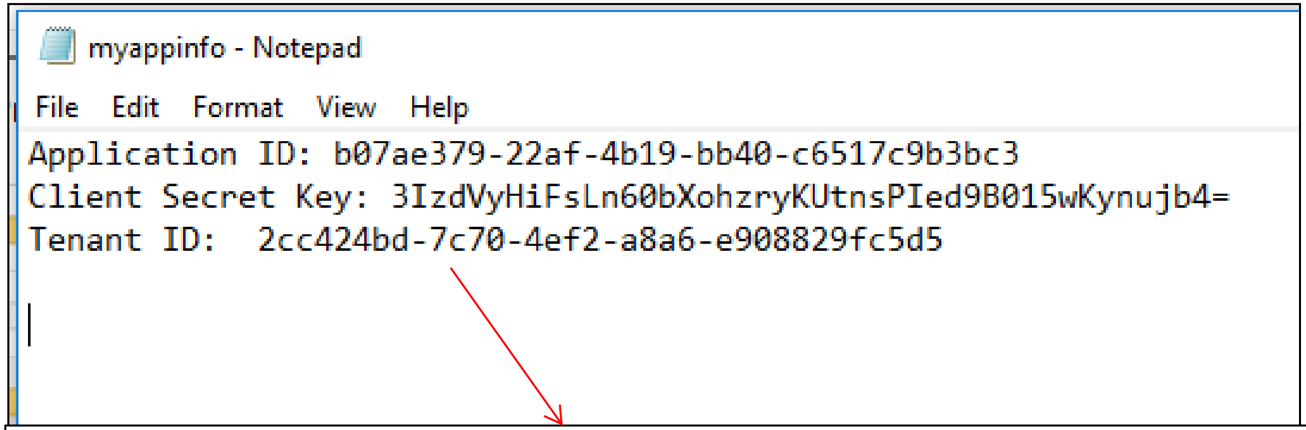
Global admin can manage Azure Subscriptions: Yes [✓ No]

Directory ID: 2cc424bd-7c70-4ef2-a8a6-e908829fc5d5

2cc424bd-7c70-4ef2-a8a6-e908829fc5d5

Save in Notepad

- Application ID: b07ae379-22af-4b19-bb40-c6517c9b3bc3
- Client Secret Key: 3IzdVyHiFsLn60bXohzryKUtnsPIed9B015wKynujb4=
- Tenant ID: 2cc424bd-7c70-4ef2-a8a6-e908829fc5d5
- We will need these values later in our Python or .NET code.



```
Application ID: b07ae379-22af-4b19-bb40-c6517c9b3bc3
Client Secret Key: 3IzdVyHiFsLn60bXohzryKUtnsPIed9B015wKynujb4=
Tenant ID:  2cc424bd-7c70-4ef2-a8a6-e908829fc5d5
```

Python Code

```
# Specify your Active Directory client ID, client secret, and tenant ID
   credentials = ServicePrincipalCredentials(client_id='<Active Directory application/client ID>',
secret='<client secret>', tenant='<Active Directory tenant ID>')
   resource_client = ResourceManagementClient(credentials, subscription_id)
   adf_client = DataFactoryManagementClient(credentials, subscription_id)
```

@Diane Howard, Nishava Inc.

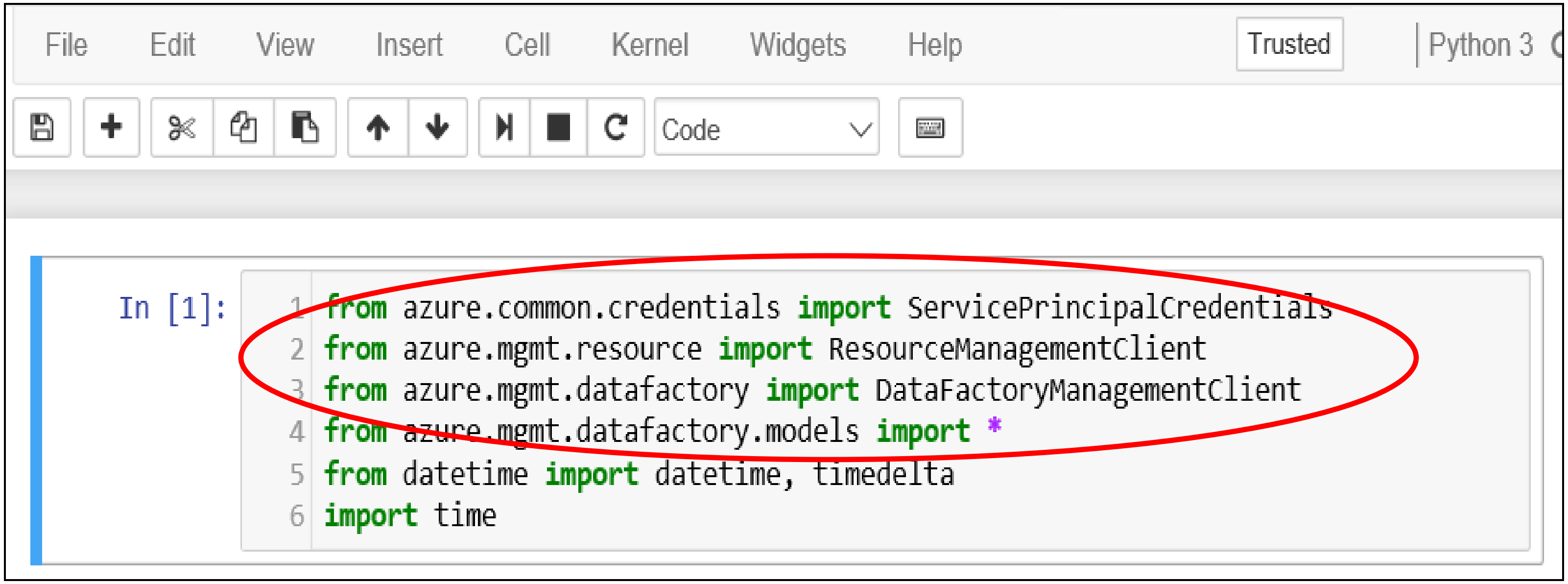
Prerequisites to Create a Data Factory V2

Python 2.7, 3.3, 3.4, 3.5 or 3.6

- ✓ Install Python SDK for Azure packages
    - Azure Management Resources
    - Data Factory
- ✓ Portal: Obtain your Subscription ID
- ✓ Create an Azure Storage Account & Blob container
- ✓ Portal: Create a data file and upload data file to Blob container
- ✓ Portal: Create an app in Active Directory (App ID, Client Secret Key, Directory ID/Tenant ID).
- Python: Create a Data Factory, Linked Service, Pipeline

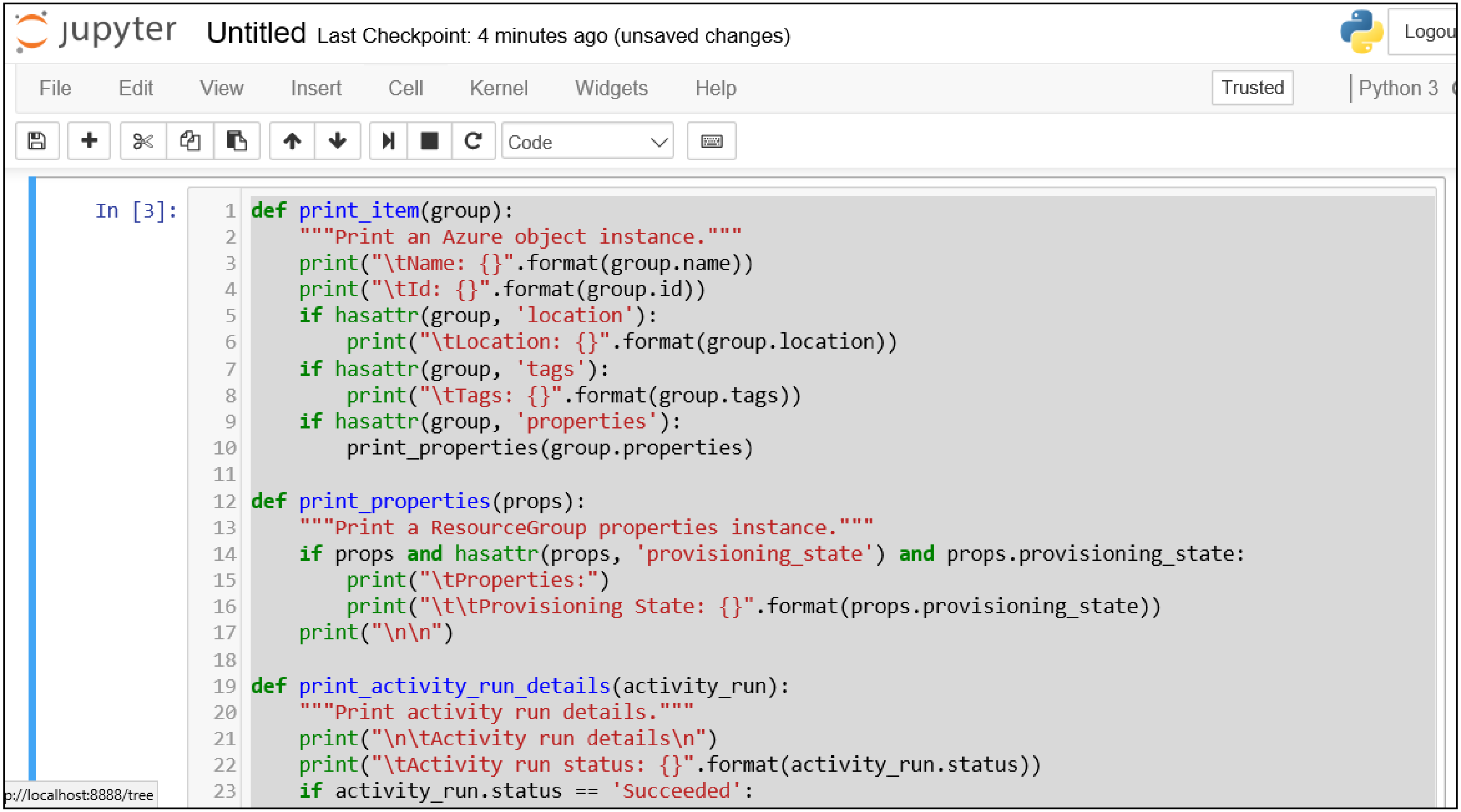
Python Code

- Add Azure imports



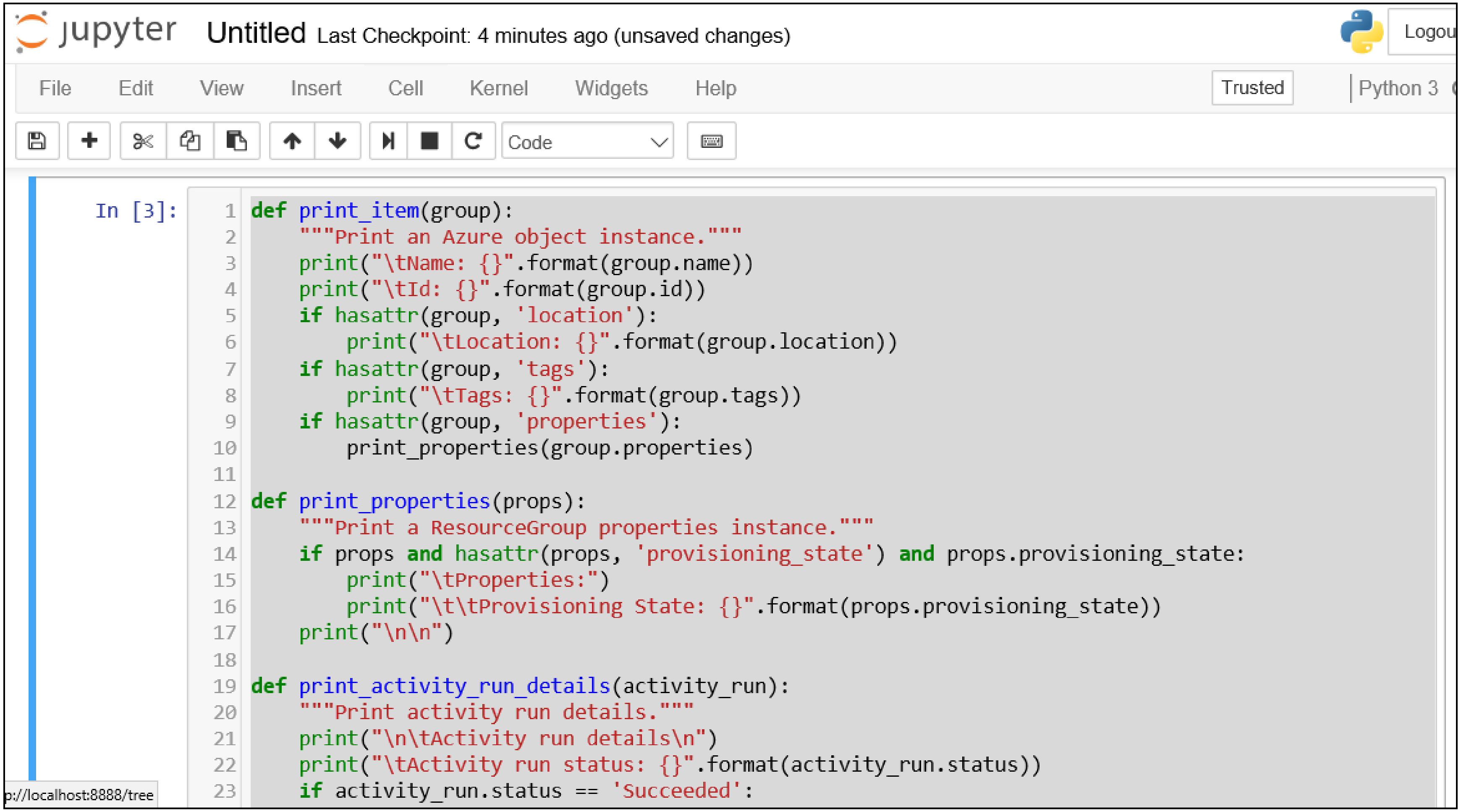
```python
from azure.common.credentials import ServicePrincipalCredentials
from azure.mgmt.resource import ResourceManagementClient
from azure.mgmt.datafactory import DataFactoryManagementClient
from azure.mgmt.datafactory.models import *
from datetime import datetime, timedelta
import time
```

Print Azure Resources & Status



```python
def print_item(group):
    """Print an Azure object instance."""
    print("\tName: {}".format(group.name))
    print("\tId: {}".format(group.id))
    if hasattr(group, 'location'):
        print("\tLocation: {}".format(group.location))
    if hasattr(group, 'tags'):
        print("\tTags: {}".format(group.tags))
    if hasattr(group, 'properties'):
        print_properties(group.properties)

def print_properties(props):
    """Print a ResourceGroup properties instance."""
    if props and hasattr(props, 'provisioning_state') and props.provisioning_state:
        print("\tProperties:")
        print("\t\tProvisioning State: {}".format(props.provisioning_state))
    print("\n\n")

def print_activity_run_details(activity_run):
    """Print activity run details."""
    print("\n\tActivity run details\n")
    print("\tActivity run status: {}".format(activity_run.status))
    if activity_run.status == 'Succeeded':
```

Print AZ Resources and Status - continued

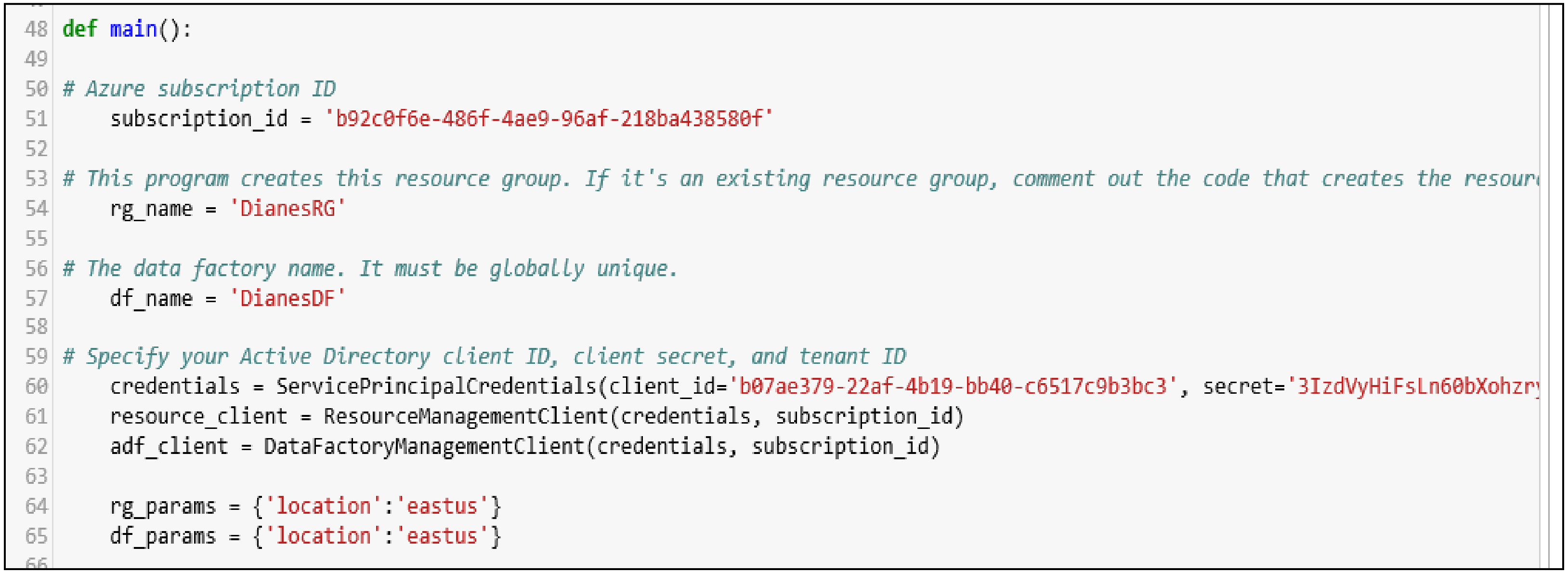


```python
def print_item(group):
    """Print an Azure object instance."""
    print("\tName: {}".format(group.name))
    print("\tId: {}".format(group.id))
    if hasattr(group, 'location'):
        print("\tLocation: {}".format(group.location))
    if hasattr(group, 'tags'):
        print("\tTags: {}".format(group.tags))
    if hasattr(group, 'properties'):
        print_properties(group.properties)

def print_properties(props):
    """Print a ResourceGroup properties instance."""
    if props and hasattr(props, 'provisioning_state') and props.provisioning_state:
        print("\tProperties:")
        print("\t\tProvisioning State: {}".format(props.provisioning_state))
    print("\n\n")

def print_activity_run_details(activity_run):
    """Print activity run details."""
    print("\n\tActivity run details\n")
    print("\tActivity run status: {}".format(activity_run.status))
    if activity_run.status == 'Succeeded':
```

@Diane Howard, Nishava Inc.

Main: Initialize Variables

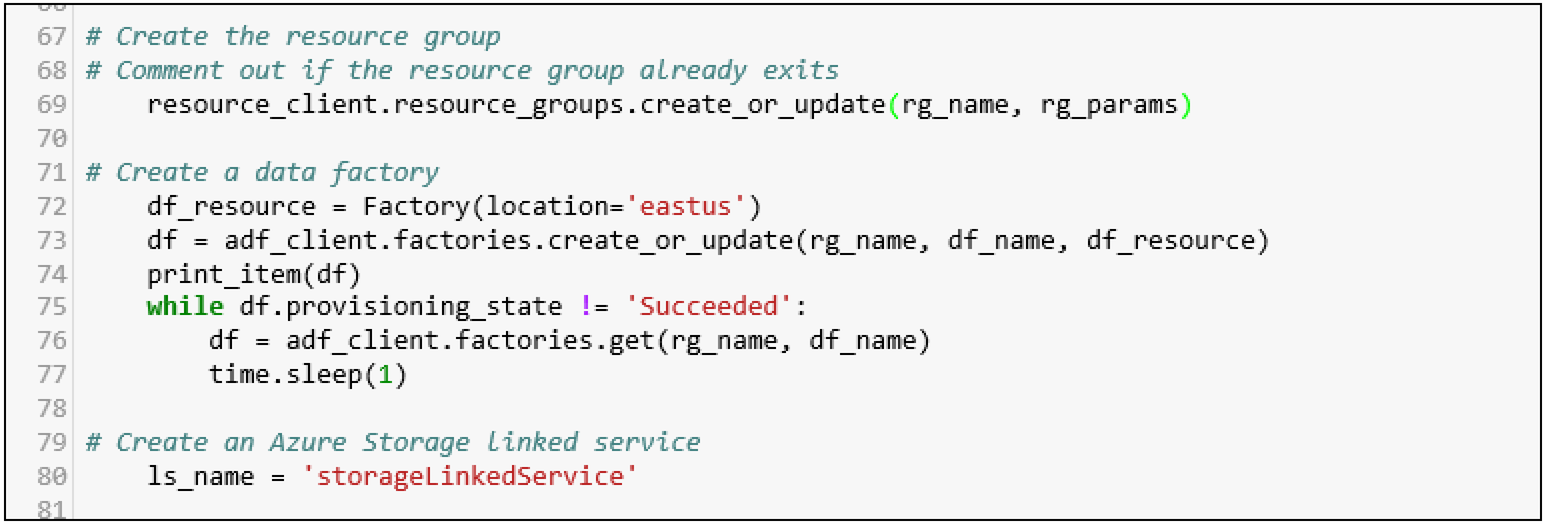
- Initialize Resource Group Name, Data Factory Name, Subscription ID and Active Directory credentials



```python
def main():

# Azure subscription ID
    subscription_id = 'b92c0f6e-486f-4ae9-96af-218ba438580f'

# This program creates this resource group. If it's an existing resource group, comment out the code that creates the resour
    rg_name = 'DianesRG'

# The data factory name. It must be globally unique.
    df_name = 'DianesDF'

# Specify your Active Directory client ID, client secret, and tenant ID
    credentials = ServicePrincipalCredentials(client_id='b07ae379-22af-4b19-bb40-c6517c9b3bc3', secret='3IzdVyHiFsLn60bXohzr
    resource_client = ResourceManagementClient(credentials, subscription_id)
    adf_client = DataFactoryManagementClient(credentials, subscription_id)

    rg_params = {'location':'eastus'}
    df_params = {'location':'eastus'}
```

Main: Create AZ Resources

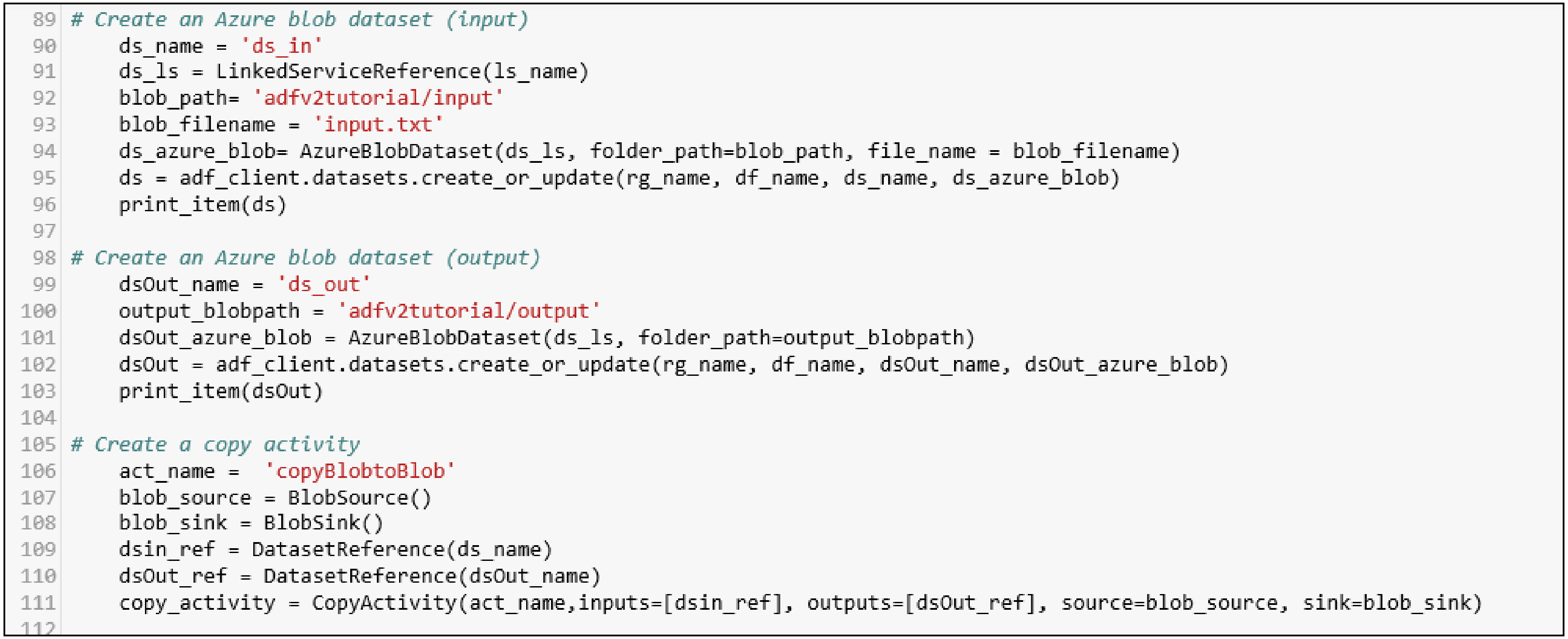
- Create Resource Group, Data Factory, and Storage Linked Service



```
# Create the resource group
# Comment out if the resource group already exits
    resource_client.resource_groups.create_or_update(rg_name, rg_params)

# Create a data factory
    df_resource = Factory(location='eastus')
    df = adf_client.factories.create_or_update(rg_name, df_name, df_resource)
    print_item(df)
    while df.provisioning_state != 'Succeeded':
        df = adf_client.factories.get(rg_name, df_name)
        time.sleep(1)

# Create an Azure Storage linked service
    ls_name = 'storageLinkedService'

```

Main: Create AZ Resources

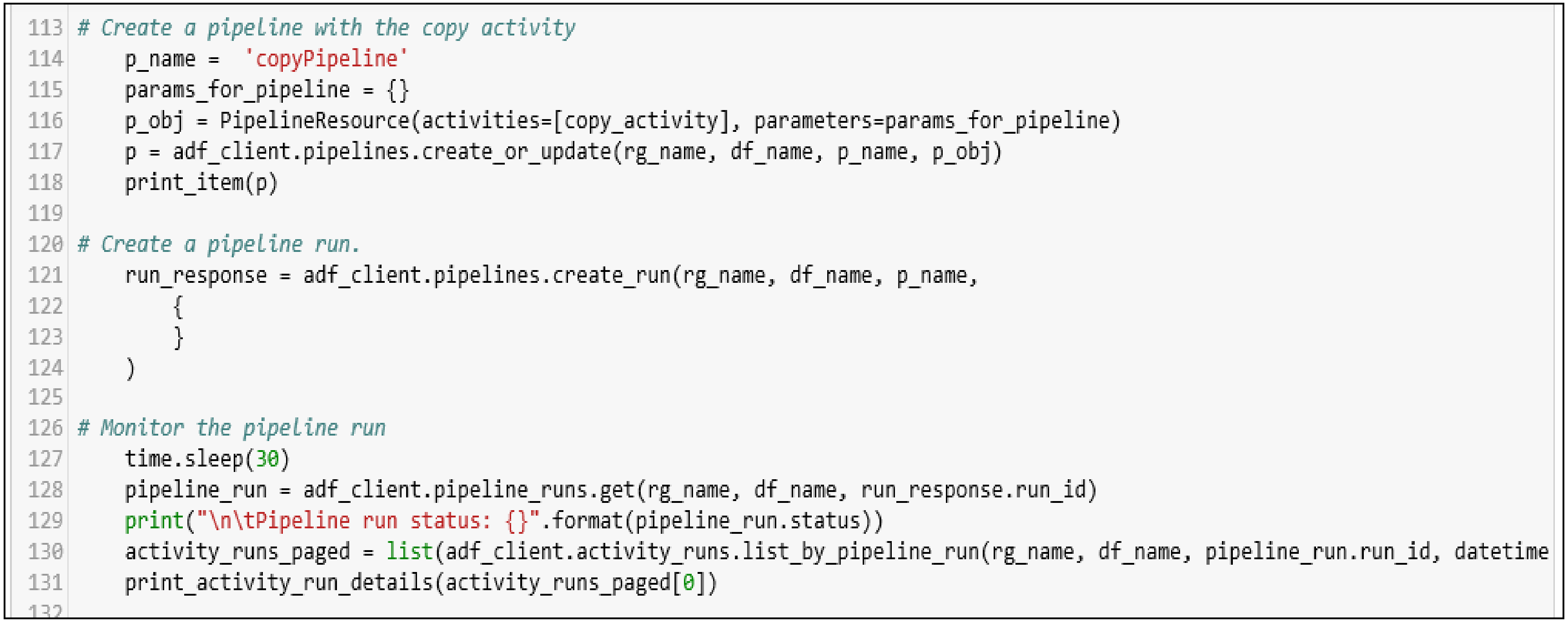
- Define Input Blob Data Source, Output Blob Sink, Copy Job (Activity)



```python
# Create an Azure blob dataset (input)
    ds_name = 'ds_in'
    ds_ls = LinkedServiceReference(ls_name)
    blob_path= 'adfv2tutorial/input'
    blob_filename = 'input.txt'
    ds_azure_blob= AzureBlobDataset(ds_ls, folder_path=blob_path, file_name = blob_filename)
    ds = adf_client.datasets.create_or_update(rg_name, df_name, ds_name, ds_azure_blob)
    print_item(ds)

# Create an Azure blob dataset (output)
    dsOut_name = 'ds_out'
    output_blobpath = 'adfv2tutorial/output'
    dsOut_azure_blob = AzureBlobDataset(ds_ls, folder_path=output_blobpath)
    dsOut = adf_client.datasets.create_or_update(rg_name, df_name, dsOut_name, dsOut_azure_blob)
    print_item(dsOut)

# Create a copy activity
    act_name =  'copyBlobtoBlob'
    blob_source = BlobSource()
    blob_sink = BlobSink()
    dsin_ref = DatasetReference(ds_name)
    dsOut_ref = DatasetReference(dsOut_name)
    copy_activity = CopyActivity(act_name,inputs=[dsin_ref], outputs=[dsOut_ref], source=blob_source, sink=blob_sink)
```

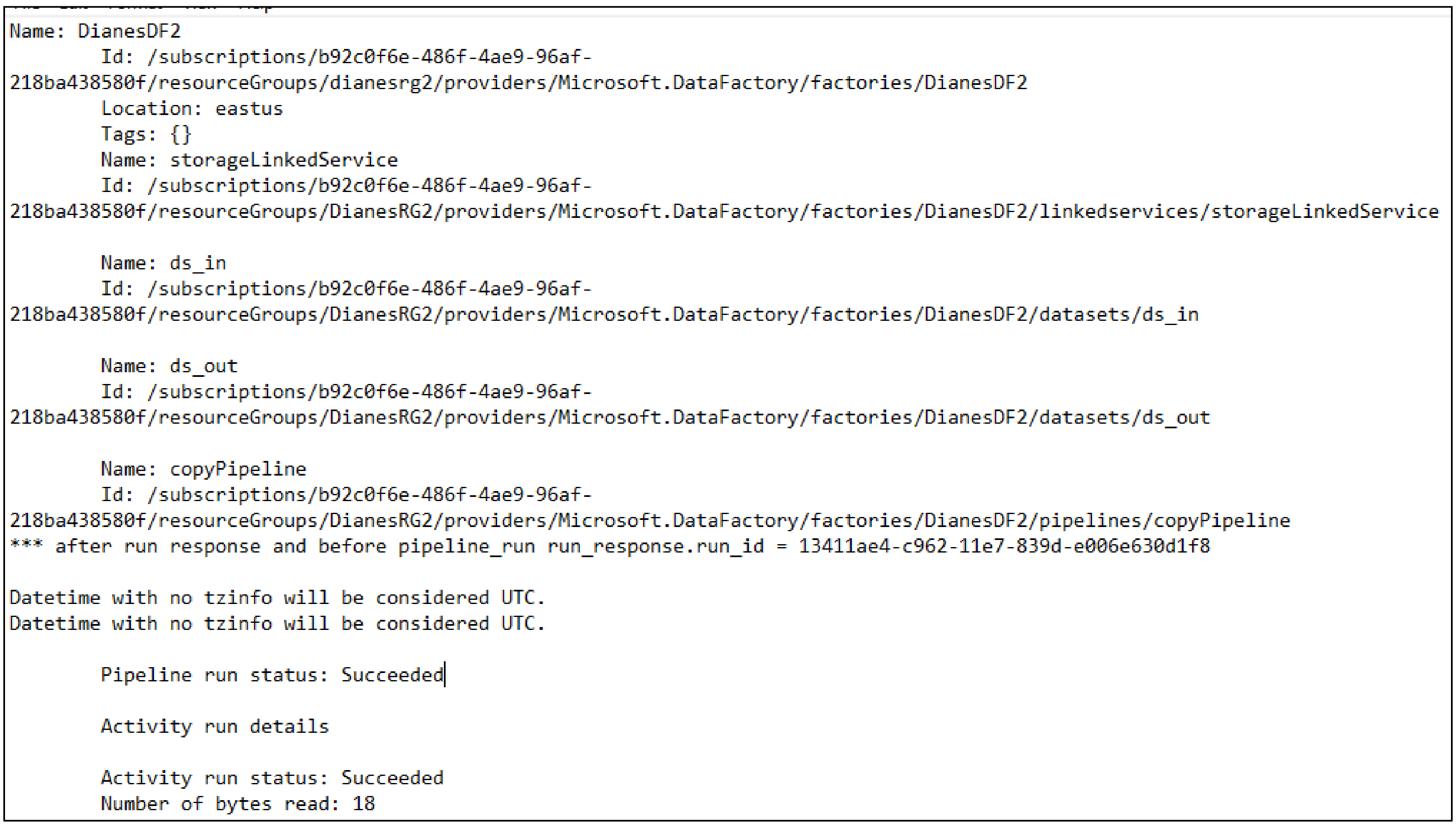
Main: Create AZ Resources

- Create the Pipeline for the Copy Activity, Pipeline Run, Set time delay for the Run, Monitor the Pipeline



```
# Create a pipeline with the copy activity
    p_name =  'copyPipeline'
    params_for_pipeline = {}
    p_obj = PipelineResource(activities=[copy_activity], parameters=params_for_pipeline)
    p = adf_client.pipelines.create_or_update(rg_name, df_name, p_name, p_obj)
    print_item(p)

# Create a pipeline run.
    run_response = adf_client.pipelines.create_run(rg_name, df_name, p_name,
        {
        }
    )

# Monitor the pipeline run
    time.sleep(30)
    pipeline_run = adf_client.pipeline_runs.get(rg_name, df_name, run_response.run_id)
    print("\n\tPipeline run status: {}".format(pipeline_run.status))
    activity_runs_paged = list(adf_client.activity_runs.list_by_pipeline_run(rg_name, df_name, pipeline_run.run_id, datetime
    print_activity_run_details(activity_runs_paged[0])
```

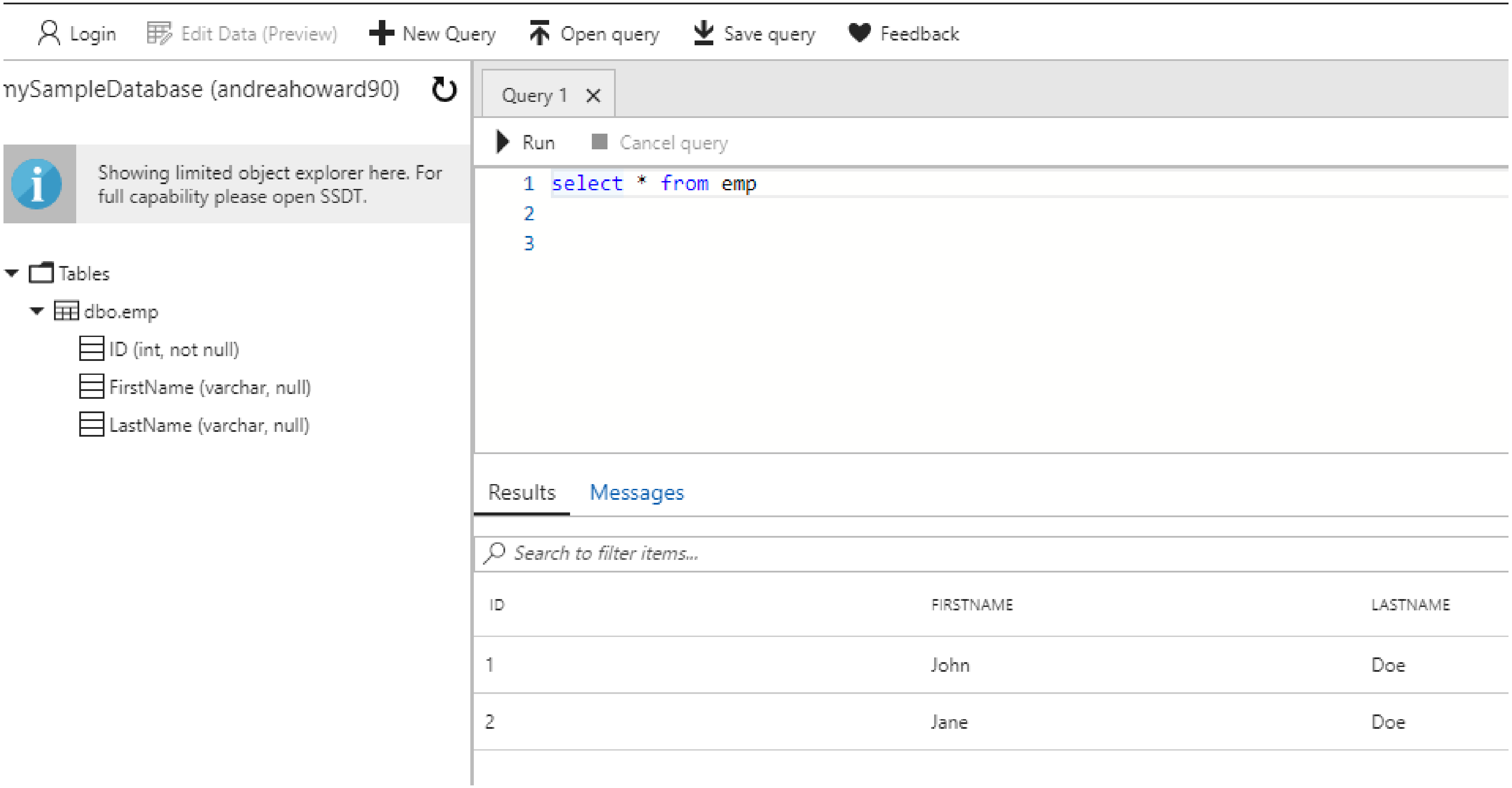
Python Output



```
Name: DianesDF2
        Id: /subscriptions/b92c0f6e-486f-4ae9-96af-
218ba438580f/resourceGroups/dianesrg2/providers/Microsoft.DataFactory/factories/DianesDF2
        Location: eastus
        Tags: {}
        Name: storageLinkedService
        Id: /subscriptions/b92c0f6e-486f-4ae9-96af-
218ba438580f/resourceGroups/DianesRG2/providers/Microsoft.DataFactory/factories/DianesDF2/linkedservices/storageLinkedService

        Name: ds_in
        Id: /subscriptions/b92c0f6e-486f-4ae9-96af-
218ba438580f/resourceGroups/DianesRG2/providers/Microsoft.DataFactory/factories/DianesDF2/datasets/ds_in

        Name: ds_out
        Id: /subscriptions/b92c0f6e-486f-4ae9-96af-
218ba438580f/resourceGroups/DianesRG2/providers/Microsoft.DataFactory/factories/DianesDF2/datasets/ds_out

        Name: copyPipeline
        Id: /subscriptions/b92c0f6e-486f-4ae9-96af-
218ba438580f/resourceGroups/DianesRG2/providers/Microsoft.DataFactory/factories/DianesDF2/pipelines/copyPipeline
*** after run response and before pipeline_run run_response.run_id = 13411ae4-c962-11e7-839d-e006e630d1f8

Datetime with no tzinfo will be considered UTC.
Datetime with no tzinfo will be considered UTC.

        Pipeline run status: Succeeded

        Activity run details

        Activity run status: Succeeded
        Number of bytes read: 18
```

Visual Studio output from ADF run

- Within the portal in SQL databases select your database that you just created.
- Select Tools editor to query your table



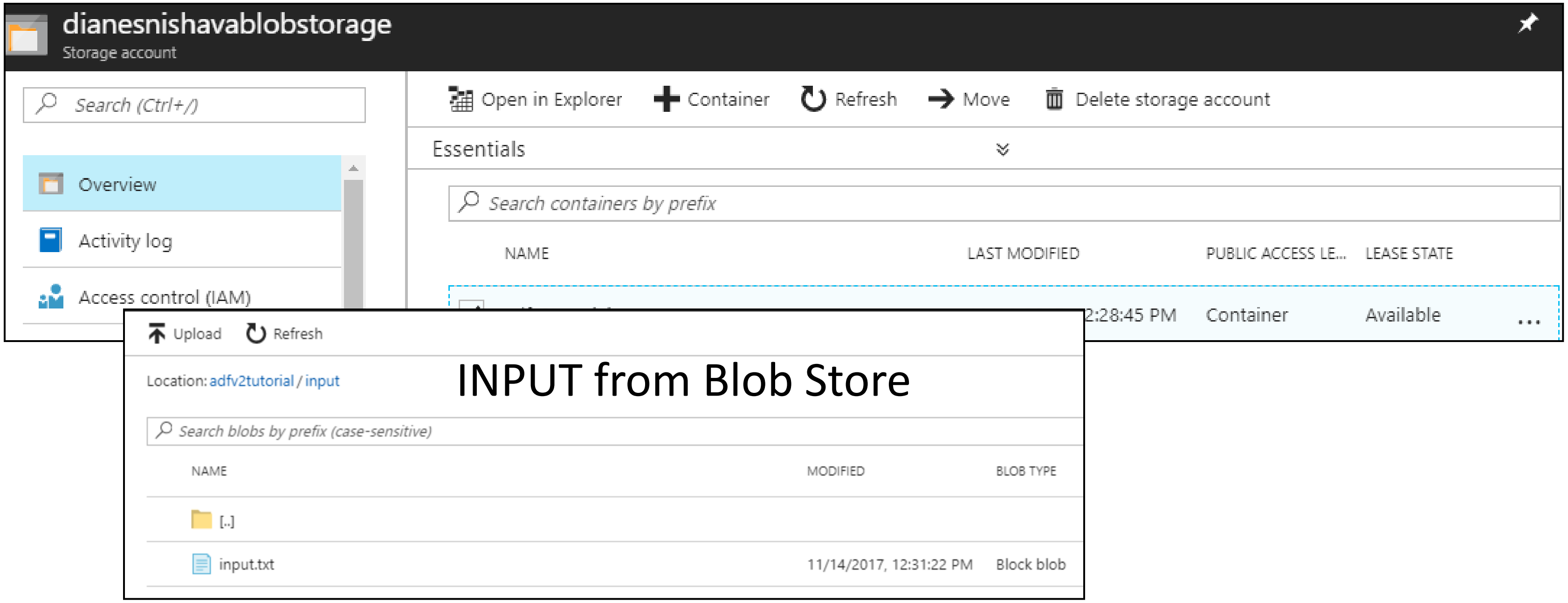
Showing limited object explorer here. For full capability please open SSDT.

```
select * from emp
```

ID	FIRSTNAME	LASTNAME
1	John	Doe
2	Jane	Doe

@Andrea Hatch, Nishava Inc

Successful copy of our Workflow



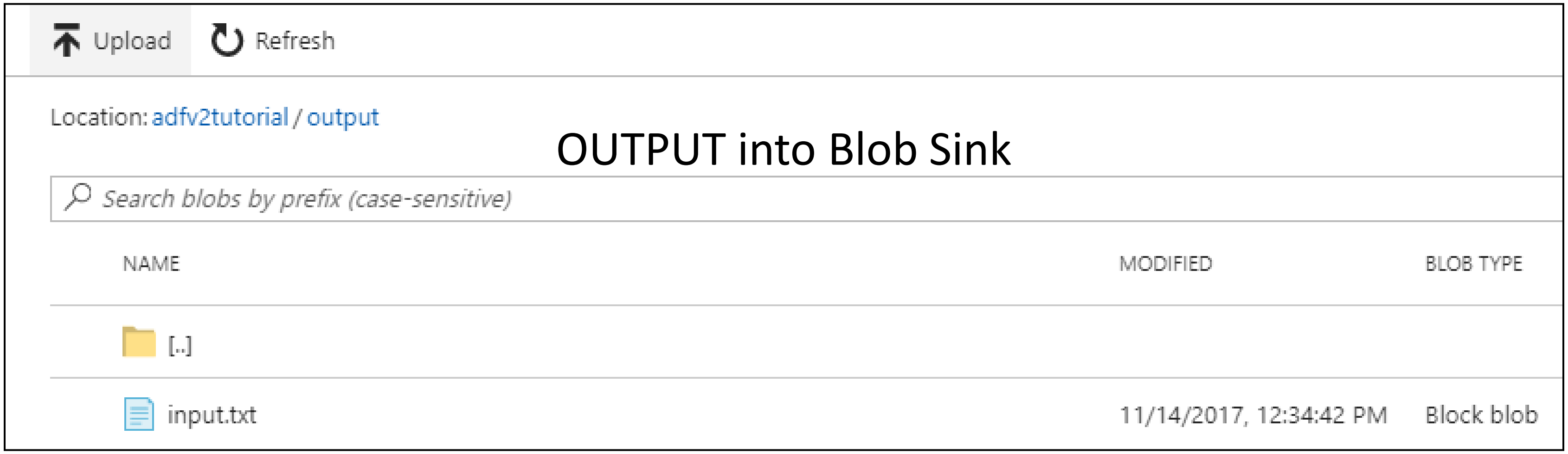
dianesnishavablobstorage
Storage account

NAME	LAST MODIFIED	PUBLIC ACCESS LE...	LEASE STATE
	2:28:45 PM	Container	Available

Location: adfv2tutorial / input

INPUT from Blob Store

NAME	MODIFIED	BLOB TYPE
[..]		
input.txt	11/14/2017, 12:31:22 PM	Block blob

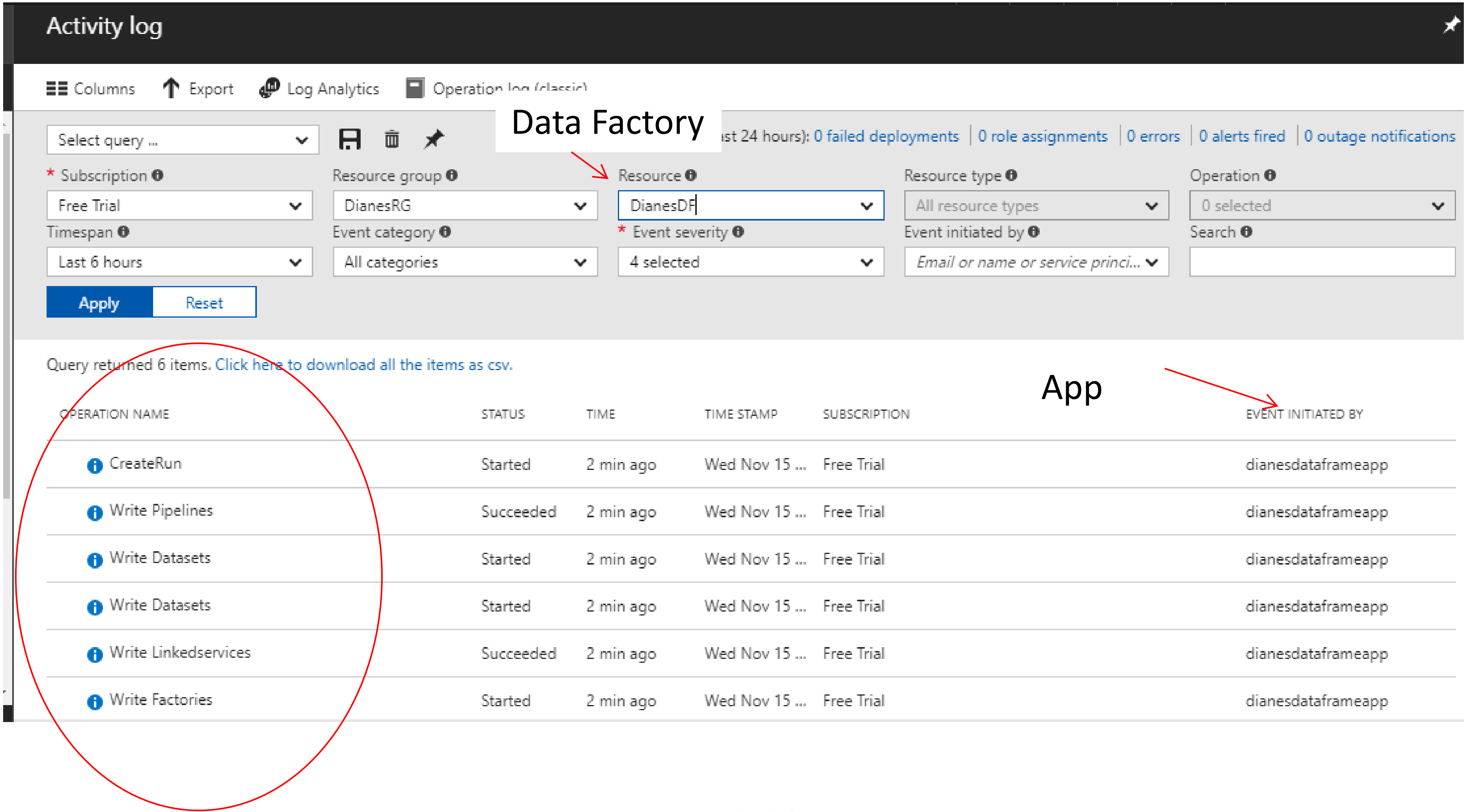


Location: adfv2tutorial / output

OUTPUT into Blob Sink

NAME	MODIFIED	BLOB TYPE
[..]		
input.txt	11/14/2017, 12:34:42 PM	Block blob

Log Activity from Azure Dashboard



Data Factory

App

Activity log

Select query ...

...st 24 hours): 0 failed deployments | 0 role assignments | 0 errors | 0 alerts fired | 0 outage notifications

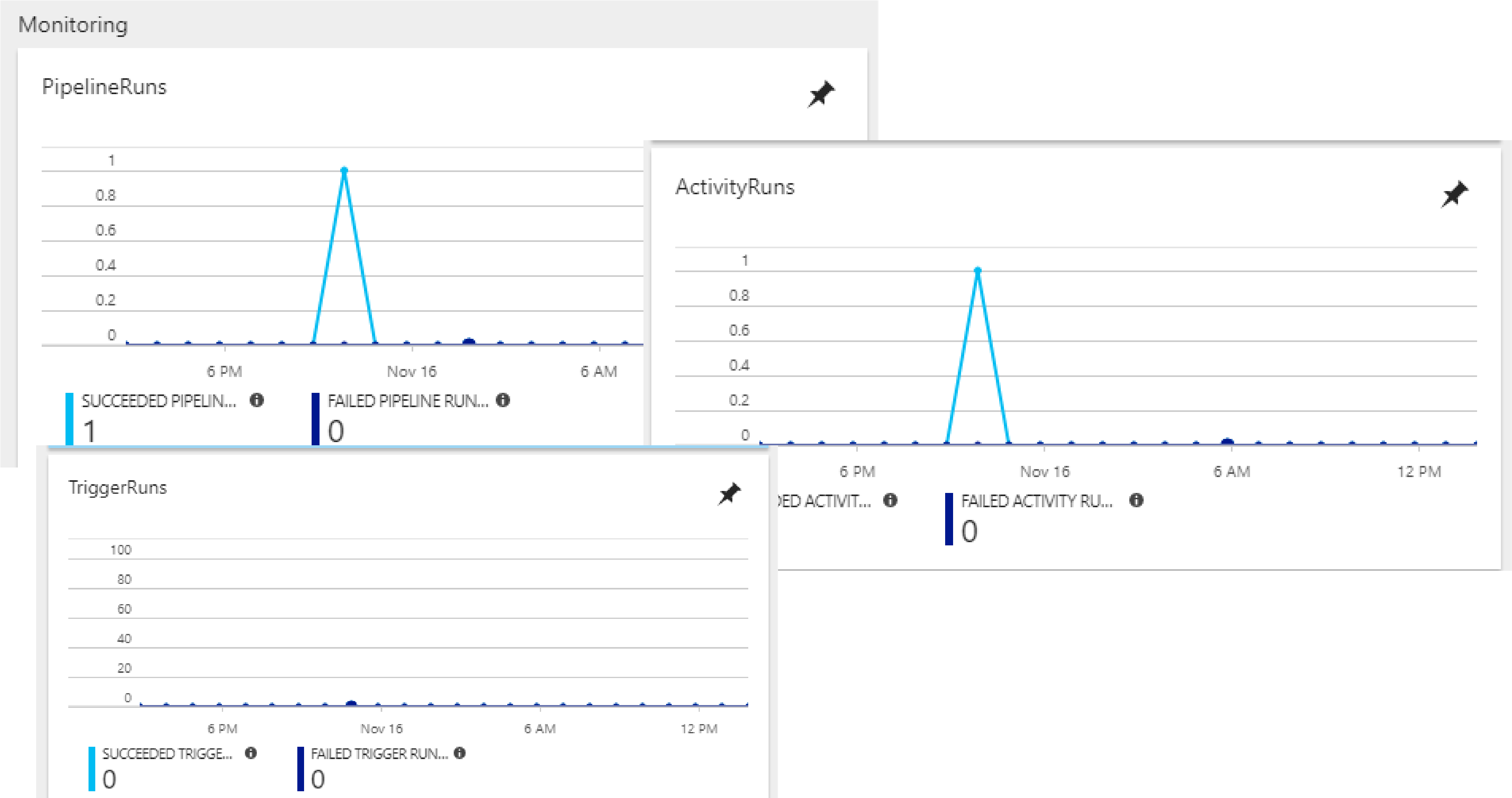
* Subscription	Resource group	Resource	Resource type	Operation
Free Trial	DianesRG	DianesDF	All resource types	0 selected
Timespan	Event category	* Event severity	Event initiated by	Search
Last 6 hours	All categories	4 selected	Email or name or service princi...	

Query returned 6 items. Click here to download all the items as csv.

OPERATION NAME	STATUS	TIME	TIME STAMP	SUBSCRIPTION	EVENT INITIATED BY
CreateRun	Started	2 min ago	Wed Nov 15 ...	Free Trial	dianesdataframeapp
Write Pipelines	Succeeded	2 min ago	Wed Nov 15 ...	Free Trial	dianesdataframeapp
Write Datasets	Started	2 min ago	Wed Nov 15 ...	Free Trial	dianesdataframeapp
Write Datasets	Started	2 min ago	Wed Nov 15 ...	Free Trial	dianesdataframeapp
Write Linkedservices	Succeeded	2 min ago	Wed Nov 15 ...	Free Trial	dianesdataframeapp
Write Factories	Started	2 min ago	Wed Nov 15 ...	Free Trial	dianesdataframeapp

Examination of Activities in Data Factory

- In your Portal you can view the workflow activities (Pipeline runs, Activity Runs, Trigger Runs.



Monitoring

PipelineRuns

SUCCEEDED PIPELIN... 1

FAILED PIPELINE RUN... 0

ActivityRuns

ED ACTIVIT...

FAILED ACTIVITY RU... 0

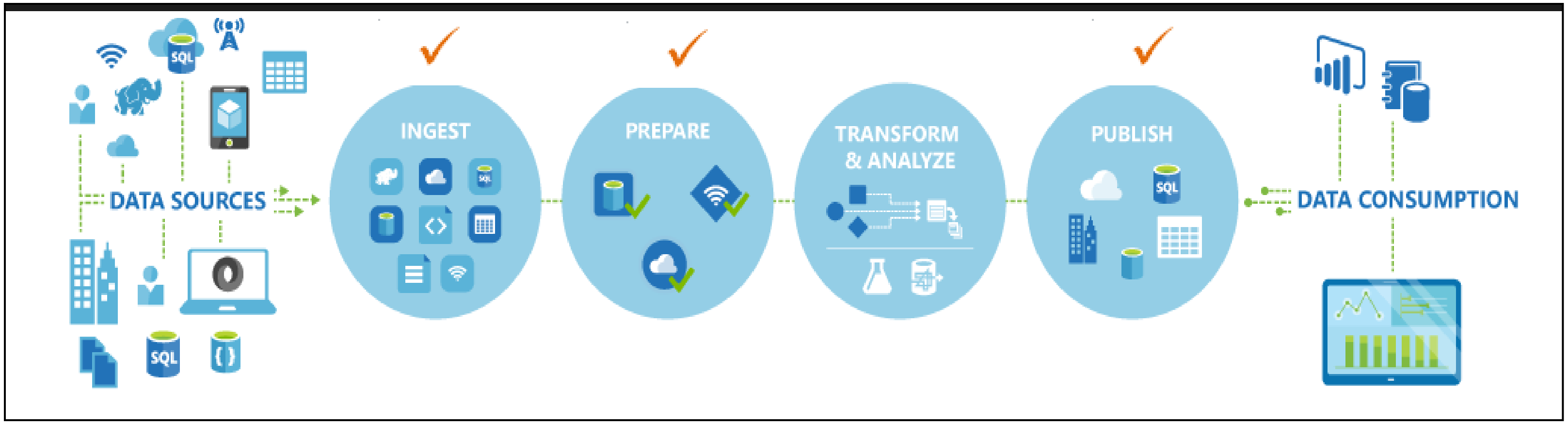
TriggerRuns

SUCCEEDED TRIGGE... 0

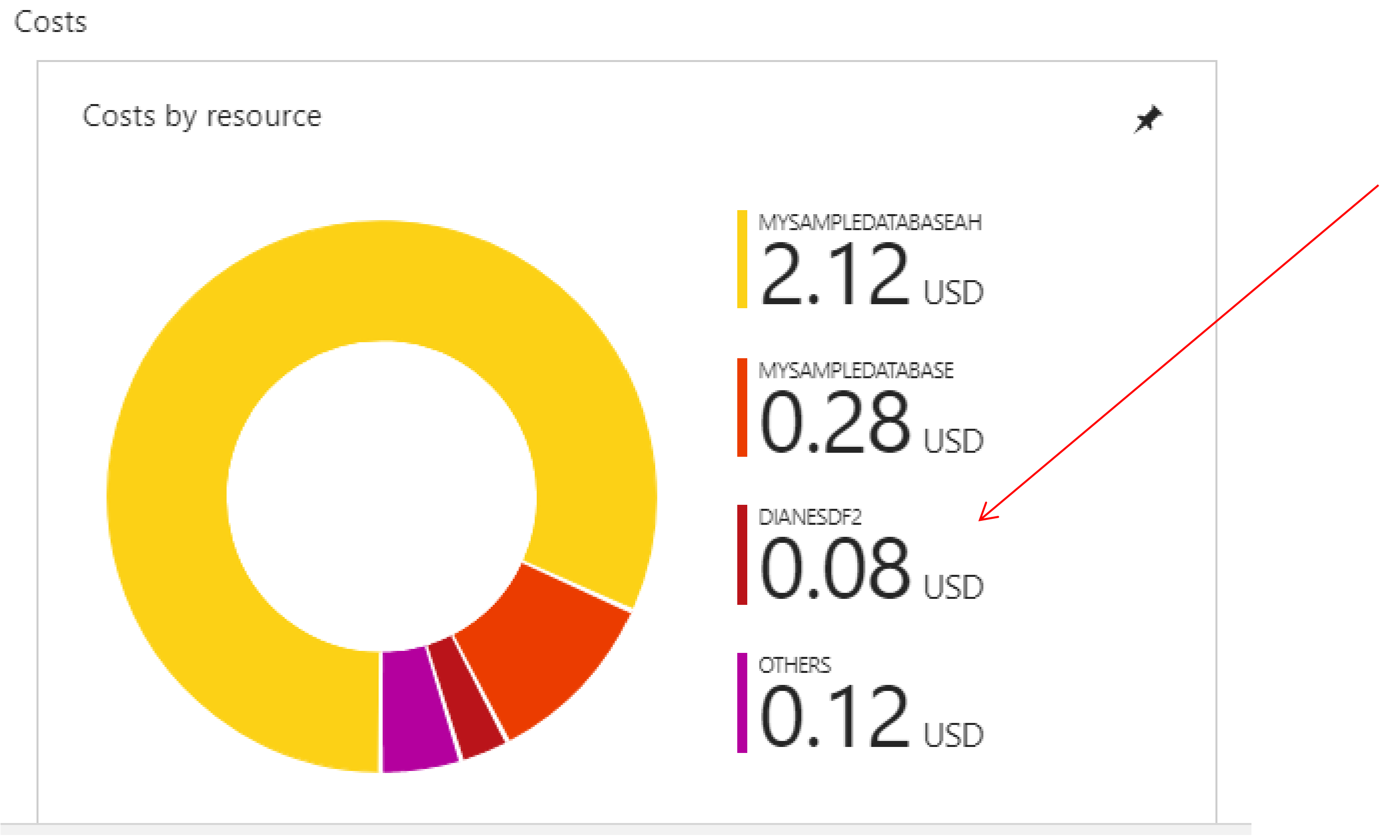
FAILED TRIGGER RUN... 0

Summary

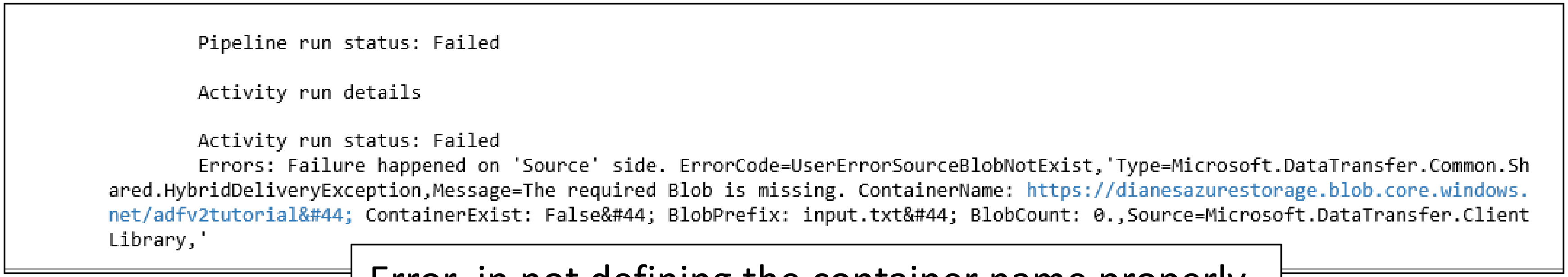
- We created a simple Data Factory in Python for a small data file which was uploaded to a Blob Store and watched the progress of the job: *Copy the data file to our Blob Sink*.
- AZ Resources Needed:
  - Subscription Info (ID)
  - Active Directory (Client ID, Client Secret Key) Storage Container (Name, ID)
  - Resource Group (Name)
- We did not perform any manipulation and analytics analysis of our data which is the heart of using Data Factory.



My Cost to Run the Data Factory Demo



Errors from run

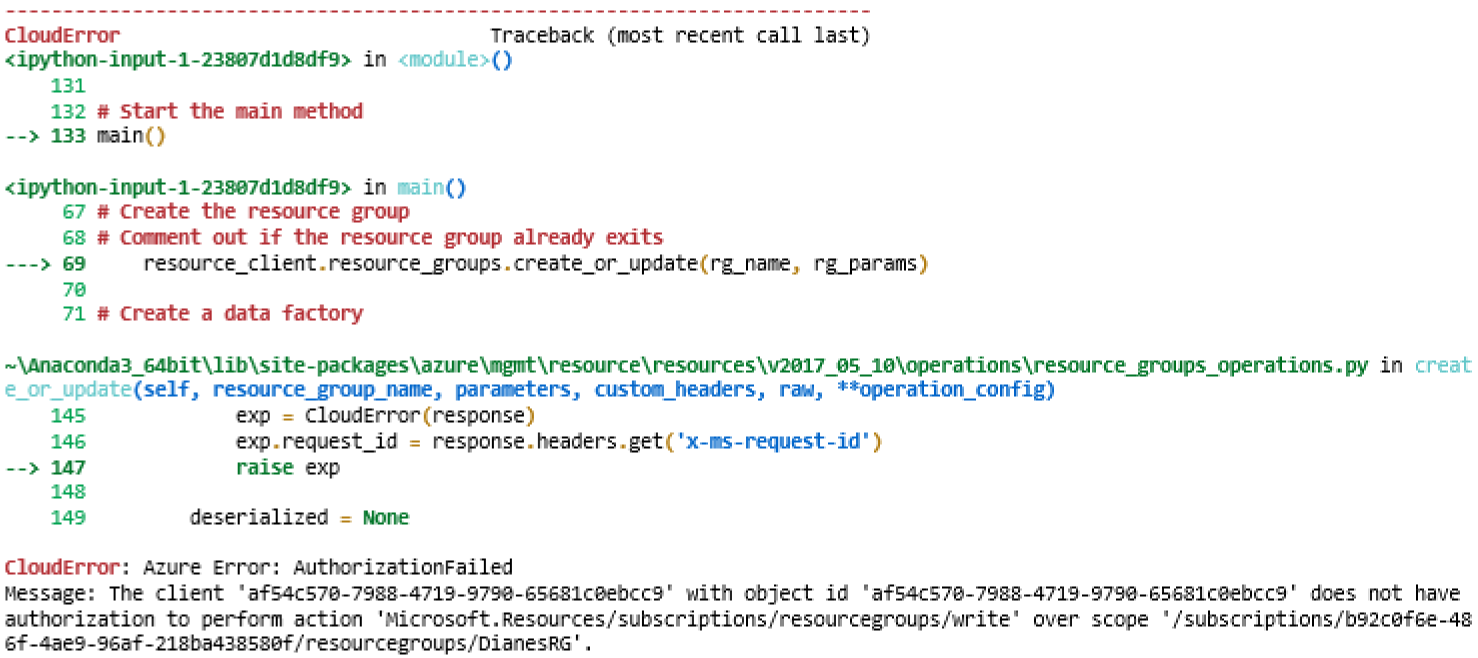


```
        Pipeline run status: Failed

        Activity run details

        Activity run status: Failed
        Errors: Failure happened on 'Source' side. ErrorCode=UserErrorSourceBlobNotExist,'Type=Microsoft.DataTransfer.Common.Sh
ared.HybridDeliveryException,Message=The required Blob is missing. ContainerName: https://dianesazurestorage.blob.core.windows.
net/adfv2tutorial&#44; ContainerExist: False&#44; BlobPrefix: input.txt&#44; BlobCount: 0.,Source=Microsoft.DataTransfer.Client
Library,'
```

Error in not defining the container name properly.

- Recommendations: Check how you defined your container vs. folder.

Issue with Credentials



```
---------------------------------------------------------------------------
CloudError                                Traceback (most recent call last)
<ipython-input-1-23807d1d8df9> in <module>()
    131 
    132 # Start the main method
--> 133 main()

<ipython-input-1-23807d1d8df9> in main()
     67 # Create the resource group
     68 # Comment out if the resource group already exits
---> 69     resource_client.resource_groups.create_or_update(rg_name, rg_params)
     70 
     71 # Create a data factory

~\Anaconda3_64bit\lib\site-packages\azure\mgmt\resource\resources\v2017_05_10\operations\resource_groups_operations.py in creat
e_or_update(self, resource_group_name, parameters, custom_headers, raw, **operation_config)
    145             exp = CloudError(response)
    146             exp.request_id = response.headers.get('x-ms-request-id')
--> 147             raise exp
    148 
    149         deserialized = None

CloudError: Azure Error: AuthorizationFailed
Message: The client 'af54c570-7988-4719-9790-65681c0ebcc9' with object id 'af54c570-7988-4719-9790-65681c0ebcc9' does not have
authorization to perform action 'Microsoft.Resources/subscriptions/resourcegroups/write' over scope '/subscriptions/b92c0f6e-48
6f-4ae9-96af-218ba438580f/resourcegroups/DianesRG'.
```

- Check your container ID – was it copied correctly?

Can’t run Pip

- Check your directory where python is installed

```
>where python
C:\Users\dhoward\anaconda3_64bit\python.exe
```

- Go to the directory where python is installed

```
>cd c:\users\dhoward\anaconda3_64bit
```

- Go to the Scripts directory where pip resides

```
>cd Scripts
c:\Users\dhoward\Anaconda3_64bit\Scripts>dir pip*
 Volume in drive C is OS
 Volume Serial Number is 2426-663B
 Directory of c:\Users\dhoward\Anaconda3_64bit\Scripts
09/25/2017  03:52 PM               197 pip-script.py
09/19/2017  08:10 AM            40,960 pip.exe
               2 File(s)         41,157 bytes
               0 Dir(s)  84,455,149,568 bytes free
```

- Run PIP from this directory (Scripts) to install your packages

```
>pip install azure-mgmt-resource
Requirement already satisfied: azure-mgmt-resource in
c:\users\dhoward\anaconda3_64bit\lib\site-packages
```