

# CLOUD COMPUTING SYSTEMS

## Lab 2

Nuno Preguiça

(nuno.preguica\_at\_fct.unl.pt)

# GOAL

In the end of this lab you should be able to:

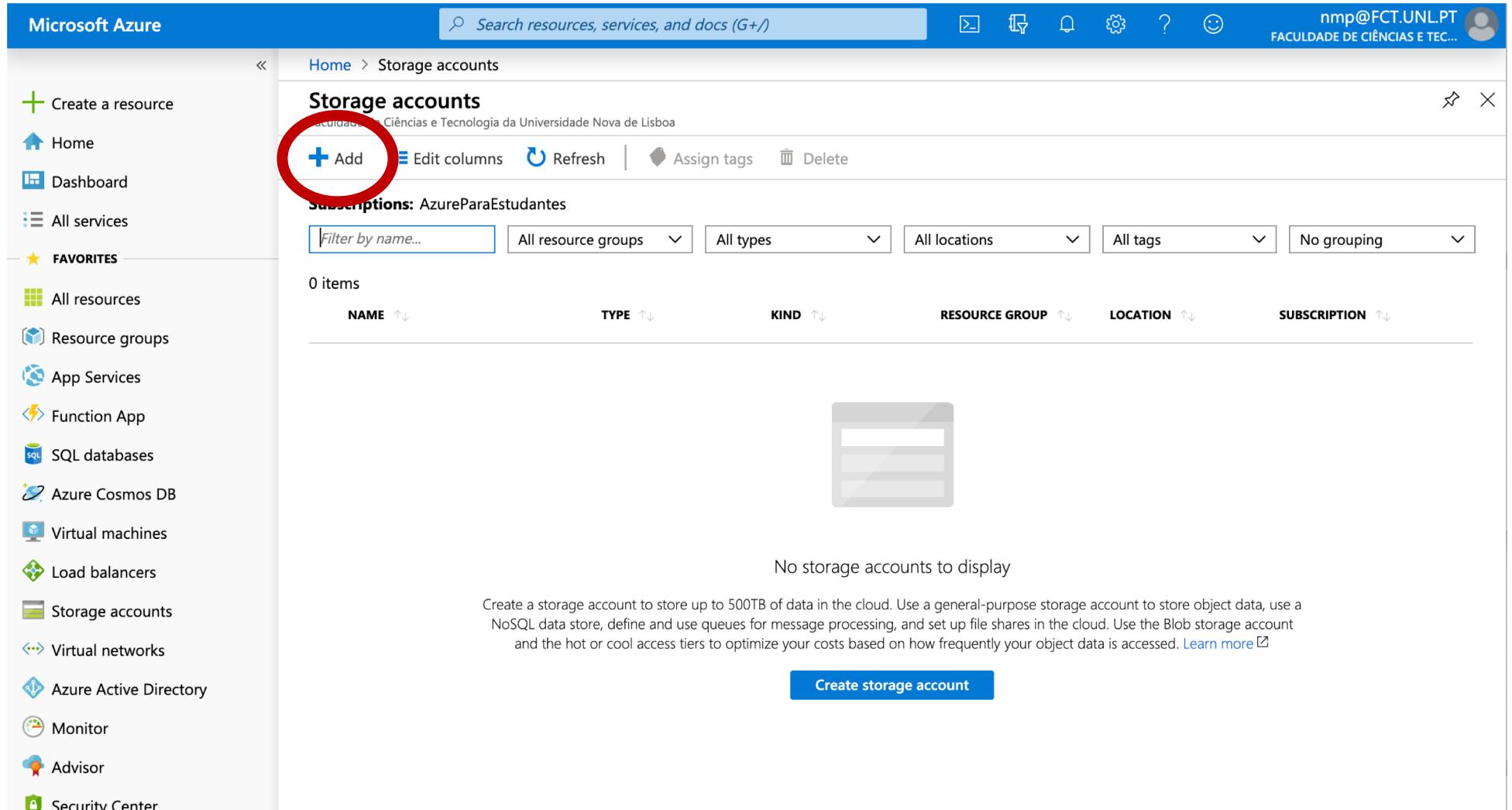
- Create a StorageAccount + Blob Container @ Azure;
- Complete the MediaResource, by storing data at Azure Blob Storage
- Next steps to do for the project...

# GOAL

In the end of this lab you should be able to:

- **Create a StorageAccount + Blob Container @ Azure;**
- Complete the MediaResource, by storing data at Azure Blob Storage
- Next steps to do for the project...

# CREATE STORAGE ACCOUNT (1)



The screenshot shows the Microsoft Azure Storage accounts page. On the left, there's a sidebar with various service icons and a 'Create a resource' button. The main area has a search bar at the top. Below it, the 'Storage accounts' section is titled 'Storage accounts' and includes a note about the Universidade Nova de Lisboa. It features a red circle around the '+ Add' button. There are also 'Edit columns', 'Refresh', 'Assign tags', and 'Delete' buttons. A filter bar below the title allows filtering by name, resource group, type, location, tags, and grouping. The table below shows 0 items, with columns for NAME, TYPE, KIND, RESOURCE GROUP, LOCATION, and SUBSCRIPTION. A large placeholder icon for a storage account is shown, along with the message 'No storage accounts to display'. At the bottom, there's a descriptive text about creating storage accounts and a 'Create storage account' button.

Microsoft Azure

Search resources, services, and docs (G+)

nmp@FCT.UNL.PT  
FACULDADE DE CIÉNCIAS E TEC...

Create a resource

Home

Dashboard

All services

FAVORITES

All resources

Resource groups

App Services

Function App

SQL databases

Azure Cosmos DB

Virtual machines

Load balancers

Storage accounts

Virtual networks

Azure Active Directory

Monitor

Advisor

Security Center

Storage accounts

+ Add

Edit columns

Refresh

Assign tags

Delete

Subscriptions: AzureParaEstudantes

Filter by name...

All resource groups

All types

All locations

All tags

No grouping

NAME ↑↓

TYPE ↑↓

KIND ↑↓

RESOURCE GROUP ↑↓

LOCATION ↑↓

SUBSCRIPTION ↑↓

No storage accounts to display

Create a storage account to store up to 500TB of data in the cloud. Use a general-purpose storage account to store object data, use a NoSQL data store, define and use queues for message processing, and set up file shares in the cloud. Use the Blob storage account and the hot or cool access tiers to optimize your costs based on how frequently your object data is accessed. [Learn more](#)

Create storage account

# CREATE STORAGE ACCOUNT (2)

The screenshot shows the 'Create storage account' wizard in the Microsoft Azure portal. The left sidebar contains a navigation menu with various service icons. The main page title is 'Create storage account' under 'Storage accounts'. A brief description of Azure Storage is provided, mentioning its features like high availability, security, durability, and scalability. The 'Project details' section allows selecting a subscription and a resource group. The 'Instance details' section includes fields for the storage account name ('scc192'), location ('(Europe) West Europe'), performance tier ('Standard'), account kind ('StorageV2 (general purpose v2)'), replication ('Read-access geo-redundant storage (RA-GRS)'), and access tier ('Hot'). At the bottom, there are 'Review + create' and 'Next : Networking >' buttons.

Microsoft Azure

Search resources, services, and docs (G/)

Home > Storage accounts > Create storage account

Create storage account

Azure Storage is a Microsoft-managed service providing cloud storage that is highly available, secure, durable, scalable, and redundant. Azure Storage includes Azure Blobs (objects), Azure Data Lake Storage Gen2, Azure Files, Azure Queues, and Azure Tables. The cost of your storage account depends on the usage and the options you choose below. [Learn more](#)

**Project details**

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

\* Subscription: AzureParaEstudantes

\* Resource group: scc-backend-rg-4204

[Create new](#)

**Instance details**

The default deployment model is Resource Manager, which supports the latest Azure features. You may choose to deploy using the classic deployment model instead. [Choose classic deployment model](#)

\* Storage account name: scc192

\* Location: (Europe) West Europe

Performance: Standard

Account kind: StorageV2 (general purpose v2)

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

Access tier (default): Hot

Review + create

< Previous

Next : Networking >

Check lecture 2 for info on this options.

# CREATE STORAGE ACCOUNT (3)

The screenshot shows the Microsoft Azure Storage Account Overview page for a deployment named "Microsoft.StorageAccount-20190925185556".

**Deployment Summary:**

- Deployment name: Microsoft.StorageAccount-20190925185556
- Subscription: AzureParaEstudantes
- Resource group: scc-backend-rg
- Start time: 9/25/2019, 7:02:00 PM
- Correlation ID: d4129cd7-0c02-47fc-ae1b-e006e

**Deployment Details:** Deployment is complete.

RESOURCE	TYPE	STATUS	OPERATION DETAILS
scc1920	Microsoft.Storage/stora...	OK	<a href="#">Operation details</a>

**Next steps:**

[Go to resource](#)

**Left sidebar:**

- Create a resource
- Home
- Dashboard
- All services
- FAVORITES**
- All resources
- Resource groups
- App Services
- Function App
- SQL databases
- Azure Cosmos DB
- Virtual machines
- Load balancers
- Storage accounts
- Virtual networks
- Azure Active Directory
- Monitor

# STORAGE ACCOUNT: ACCESS KEYS (FOR CODE)

The screenshot shows the Microsoft Azure portal interface for managing storage account access keys. The left sidebar contains navigation links for creating resources, home, dashboard, all services, and various favorites like all resources, resource groups, app services, function apps, SQL databases, Azure Cosmos DB, virtual machines, load balancers, storage accounts, virtual networks, Azure Active Directory, monitor, advisor, and security center. The main content area is titled 'scc1920 - Access keys' for a storage account named 'scc1920'. It includes a search bar, a list of management options (activity log, access control, tags, diagnose, data transfer, events, storage explorer), and a 'Settings' section with links to access keys, geo-replication, CORS, configuration, encryption, shared access signature, firewalls, and advanced security. The 'Access keys' section displays two keys: 'key1' and 'key2'. The 'key1' connection string is highlighted with a red oval: `DefaultEndpointsProtocol=https;AccountName=scc1920;AccountKey=suTNjWBGcTDCB60EJ6YEngRlefzRbC9BbkLXI+yAzgWOYCnDuutwgvWWArdppu8e`. The 'key2' connection string is also shown: `DefaultEndpointsProtocol=https;AccountName=scc1920;AccountKey=9bHc07gC/dUT2K4+hvlhCad/miLr4Nb8OvrOAJ+d17Lqn8/U/q9WN5YTKEc3X3ajl`.

# CREATE BLOB STORAGE CONTAINER (1)

The screenshot shows the Microsoft Azure Storage Account Overview page for the account 'scc1920'. The left sidebar contains various navigation links. The main content area displays the storage account details under the 'Overview' tab. A large red circle highlights the 'Blobs' service section, which is described as 'REST-based object storage for unstructured data'. Other services shown include 'Files', 'Tables', and 'Queues'.

**scc1920**  
Storage account

Search (Cmd+/)

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Data transfer

Events

Storage Explorer (preview)

Settings

Access keys

Geo-replication

CORS

Configuration

Encryption

Shared access signature

Firewalls and virtual networks

Advanced security

Static website

Resource group (change)  
scc-backend-rg

Status  
Primary: Available, Secondary: Available

Location  
West Europe, North Europe

Subscription (change)  
AzureParaEstudiantes

Subscription ID  
b0850bd9-05dd-4f04-9200-88e96d65f3bd

Tags (change)  
Click here to add tags

**Blobs**  
REST-based object storage for unstructured data

Learn more

**Files**  
File shares that use the standard SMB 3.0 protocol

Learn more

**Tables**  
Tabular data storage

Learn more

**Queues**  
Effectively scale apps according to traffic

Learn more

# CREATE BLOB STORAGE CONTAINER (2)

The screenshot shows the Microsoft Azure Storage Account Overview page for the account 'scc1920'. The left sidebar contains various navigation links. The main content area displays the storage account details under the 'Overview' tab. A large red circle highlights the 'Blobs' service section, which is described as 'REST-based object storage for unstructured data'. Other services shown include 'Files', 'Tables', and 'Queues'.

**scc1920**  
Storage account

Search (Cmd+/)

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Data transfer

Events

Storage Explorer (preview)

Settings

Access keys

Geo-replication

CORS

Configuration

Encryption

Shared access signature

Firewalls and virtual networks

Advanced security

Static website

Resource group (change)  
scc-backend-rg

Status  
Primary: Available, Secondary: Available

Location  
West Europe, North Europe

Subscription (change)  
AzureParaEstudiantes

Subscription ID  
b0850bd9-05dd-4f04-9200-88e96d65f3bd

Tags (change)  
Click here to add tags

**Blobs**  
REST-based object storage for unstructured data

Learn more

**Files**  
File shares that use the standard SMB 3.0 protocol

Learn more

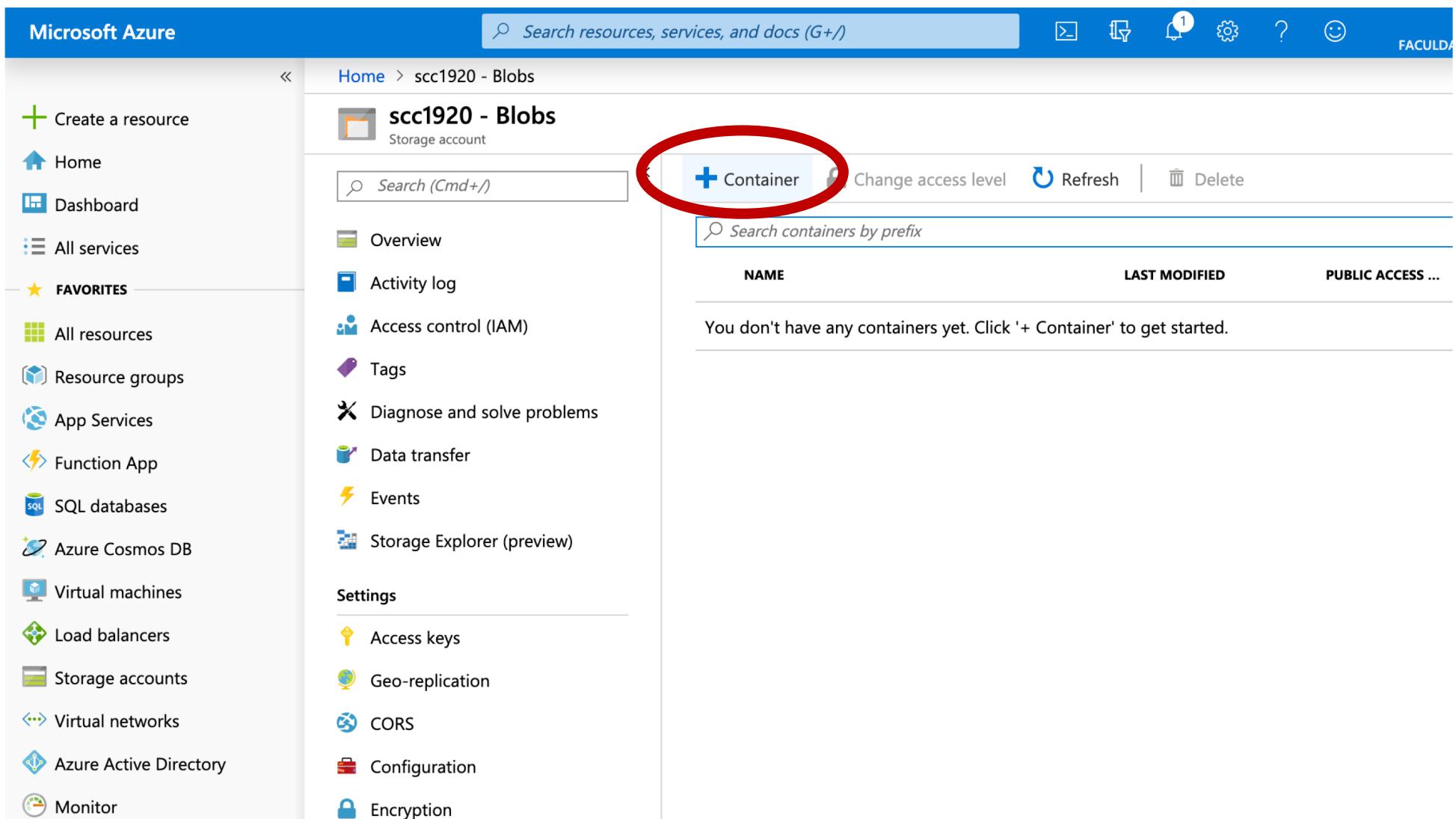
**Tables**  
Tabular data storage

Learn more

**Queues**  
Effectively scale apps according to traffic

Learn more

# CREATE BLOB STORAGE CONTAINER (2)



The screenshot shows the Microsoft Azure portal interface for managing blob storage containers. The left sidebar contains navigation links for creating resources, home, dashboard, and various services like App Services and SQL databases. The main content area shows the 'scc1920 - Blobs' storage account. A red circle highlights the '+ Container' button in the top right of the main panel. Below it is a search bar for containers by prefix. A message indicates there are no containers yet, with a note to click the button to get started. The main table lists columns for NAME, LAST MODIFIED, and PUBLIC ACCESS ...

NAME	LAST MODIFIED	PUBLIC ACCESS ...
You don't have any containers yet. Click '+ Container' to get started.		

# CREATE BLOB STORAGE CONTAINER (3)

The screenshot shows the Microsoft Azure portal interface. On the left, there's a sidebar with various service icons and a search bar at the top. The main area shows a storage account named "scc1920 - Blobs". A modal dialog is open for creating a new container. The dialog has fields for "Name" (set to "images") and "Public access level" (set to "Blob (anonymous read access for blobs only)". A warning message states: "Blobs within the container can be read by anonymous request, but container data is not available. Anonymous clients cannot enumerate the blobs within the container." At the bottom of the dialog are "OK" and "Cancel" buttons. A large callout box at the bottom right contains the text: "Set anonymous read access for blobs if you plan to allow applications to retrieve blobs directly from Blob Storage."

Microsoft Azure

Search resources, services, and docs (G+/)

Home > scc1920 - Blobs

scc1920 - Blobs

Create a resource

Home

Dashboard

All services

FAVORITES

All resources

Resource groups

App Services

Function App

SQL databases

Azure Cosmos DB

Virtual machines

Load balancers

Storage accounts

Virtual networks

Azure Active Directory

Monitor

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Data transfer

Events

Storage Explorer (preview)

Settings

Access keys

Geo-replication

CORS

Configuration

Encryption

+ Container

Change access level

Refresh

Delete

New container

\* Name: images

Public access level: Blob (anonymous read access for blobs only)

Blobs within the container can be read by anonymous request, but container data is not available. Anonymous clients cannot enumerate the blobs within the container.

OK Cancel

Set anonymous read access for blobs if you plan to allow applications to retrieve blobs directly from Blob Storage.

# CONTAINER BASE URL

The base URL allows direct access to the blob by clients.

Try uploading a file (in Azure interface) and downloading it in the browser using this URL.

The screenshot shows the Microsoft Azure portal interface. The left sidebar lists various services: Create a resource, Home, Dashboard, All services, Favorites (All resources, Resource groups, App Services, Function App, SQL databases, Azure Cosmos DB, Virtual machines, Load balancers, Storage accounts, Virtual networks, Azure Active Directory, Monitor, Advisor, Security Center, Cost Management + Billing). The main content area shows the 'Microsoft.StorageAccount-20190925224558 - Overview' page for a storage account named 'scc1920'. The 'images - Properties' section is selected. The URL for the container is displayed as <https://scc1920.blob.core.windows.net/images>. Other properties shown include Name (images), Last Modified (9/25/2019, 10:51:29 PM), ETAG (0x8D74202859C7266), Lease Status (Unlocked), Lease State (Available), and Lease Duration (-). A 'Calculate size' button is at the bottom.

# GOAL

In the end of this lab you should be able to:

- Create a StorageAccount + Blob Container @ Azure;
- **Complete the MediaResource, by storing data at Azure Blob Storage**
- Next steps to do for the project...

# ACCESSING AZURE (BLOB) STORAGE

We will use the library provided by Microsoft.

Java Docs available at:

<https://javadoc.io/doc/com.microsoft.azure/azure-storage/8.4.0>

Example available at:

<https://docs.microsoft.com/en-us/azure/storage/blobs/storage-quickstart-blobs-java>

# MAVEN DEPENDENCIES

```
<dependency>
    <groupId>com.microsoft.azure</groupId>
    <artifactId>azure-storage</artifactId>
    <version>8.4.0</version>
</dependency>
```

## STEP 1: CREATE CLIENT TO BLOB STORAGE

```
// Get connection string in the storage access keys page
String storageConnectionString = ...  
  
CloudStorageAccount storageAccount =
    CloudStorageAccount.parse(storageConnectionString);
CloudBlobClient blobClient =
    storageAccount.createCloudBlobClient();
  
  
CloudBlobContainer container =
    blobClient.getContainerReference("images");
```

## STEP 2: UPLOAD BYTE ARRAY

```
// Get reference to blob  
CloudBlob blob = container.getBlockBlobReference( name);  
  
// Upload contents from byte array (check documentation  
for other alternatives)  
blob.uploadFromByteArray(contents, 0, contents.length);
```

## STEP 3: DOWNLOAD TO BYTE ARRAY

```
// Get reference to blob
CloudBlob blob = container.getBlobReferenceFromServer(
    name);

ByteArrayOutputStream out = new ByteArrayOutputStream();
blob.download(out);
out.close();

byte[] contents = out.toByteArray();
```

## SOME NOTES

- Is it possible to create containers, etc. from the code?

Yes. Check documentation.

# GOAL

In the end of this lab you should be able to:

- Create a StorageAccount + Blob Container @ Azure;
- Complete the MediaResource, by storing data at Azure Blob Storage
- **Next steps to do for the project...**

# PROJECT

Do the backbone of a system similar to Reddit.

No strict specification, but some features are mandatory.

# MULTIMEDIA OBJECT

For each multimedia object, there will be a unique identifier.

# Post

For each post, the system must maintain the following information:

- The community;
- The creator;
- Time of creation;
- Text message;
- One link to a multimedia object (optional);
- Reference to the parent post (optional)
  - NOTE: posts without a parent are base posts;
- Number of likes.

# USER

For each post, the system must maintain the following information:

- Name.

NOTE: for now, we will ignore authentication.

# COMMUNITY

For each community, the system must maintain the following information:

- Name.

# PAGES WITH MULTIPLE INFORMATION

The system will have pages that provide multiple data:

- Conversation page: includes the initial post and replies to the post;
- Initial page: includes a selected list of conversations.

# TODO

Extend the server to support these new resources.

- Create the necessary endpoint;
  - Base endpoints for multimedia object, post, user and community will be provided.
- Create the objects to represent the resources;
- Start thinking on how to store info on CosmosDB (lecture 3 will introduce the model and lab 3 for code);
- Start thinking on how to provide information for pages with multiple information.