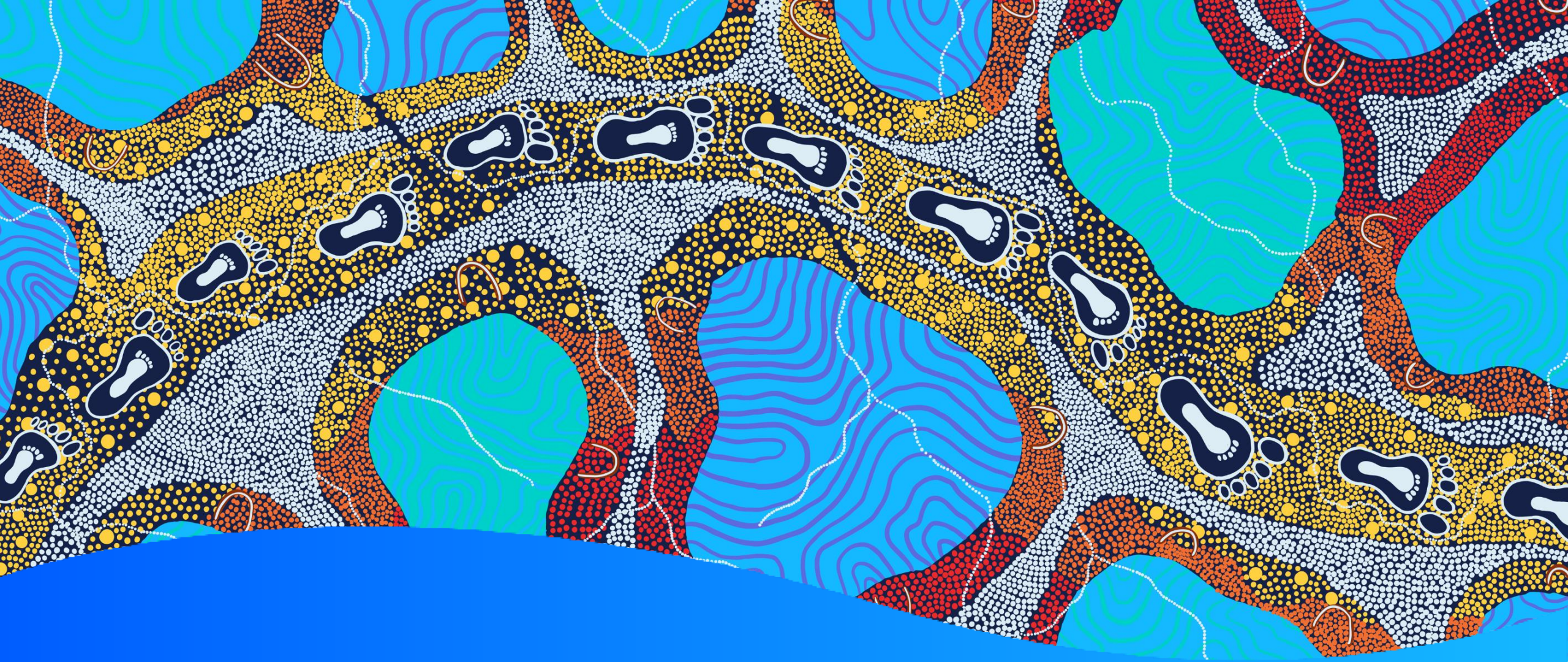


HSS Applications Azure Hands-on Session

Day 1 – Intro to Azure Portal & Azure Storage





HSS acknowledges the traditional custodians throughout Western Australia and their continuing connection to the land, waters and community. We pay our respects to all members of Aboriginal communities and their cultures, and acknowledge the wisdom of Elders both past and present.

Azure Portal – Where you start your Azure journey



HSS health
support
services

Azure Portal – What is it

1. The place where all your workloads are stored.
2. The place where you manage and govern your workloads.

The screenshot displays the Microsoft Azure Portal interface. The left sidebar contains navigation links for 'Create a resource', 'Home', 'Dashboard', 'All services', 'FAVORITES', 'All resources', 'Resource groups', 'App Services', 'SQL databases', 'Virtual machines', 'Load balancers', 'Virtual networks', 'Storage accounts', 'Azure Active Directory', 'Monitor', 'Advisor', 'Microsoft Defender for Cloud', 'Help + support', 'Cost Management + Billing', 'Function App', 'Static Web Apps', 'Azure Cosmos DB', 'Kubernetes services', and 'Container registries'. The main content area is titled 'Dashboard' and shows a 'Private dashboard' with a search bar and various action buttons like 'New dashboard', 'Refresh', 'Full screen', 'Edit', 'Share', 'Export', 'Clone', 'Assign tags', 'Delete', and 'Feedback'. Below this, there are two 'Resources' sections. The first section, 'hssappdemorg', lists resources like 'hssappdemostorage' (Storage account), 'testgetusername' (App Service), 'hssappdemoasp' (App Service plan), 'hssappsdemowebapp' (App Service), 'hssappdemofuncapp' (Function App), and 'hssappdemofuncapp' (Application Insights). The second section, 'patrickplaygroundrg', lists resources like 'patrickwebapp' (Storage account), 'WAFormConfiguration' (SQL database), 'patplaygroundsqlserver' (SQL server), 'TestTrackerPatrick' (SQL database), 'PatrickPlayWebApp' (App Service), 'PatrickPlaygroundSQL' (SQL database), and 'ASP-PatPlay-AzureFunction' (App Service plan). To the right of these sections is an 'Application map' for 'HSSAPPSDEMOWEBAPP - LAST 24 HOURS', which currently shows 'No data available' and a button to 'Get started with Application Map'. Below the application map is a 'Quickstarts + tutorials' section with links for 'Windows Virtual Machines', 'Linux Virtual Machines', 'App Service', and 'Functions'. A 'Marketplace' button is also visible.

Azure Storage



HSS health
support
services

What is Azure Storage

1. A one-stop "service" for all things relating to storing any objects
2. Has 4 types of "storage"
 1. Blob Storage (aka Containers) – Think of this like a disk space for you to store any objects
 2. Queue – Store messages/work items that needs to be processed
 3. Table – a very simple DocumentDB type table storage to store some semi-structured data (think NoSQL)
 4. File Share – A UNC-type storage area which you can mount on any authorised workstations

Azure Storage - Redundancy

Redundancy:

- Protect your storage from data loss
- Allow you to store multiple copies of your data into

1. Local Redundancy (LRS) – Redundant copies of your data is stored and copied to different segmented zones in the same data centre
2. Zone Redundancy (ZRS) – Redundant copies of your data is stored and copied to different zones (data centres) in the same region
3. Geo Redundancy (GRS) – Redundant copies of your data is stored and copied to different regions (geographical regions)
 1. GRS – Data is copied three times in the primary region using LRS and then copied to secondary region
 2. GZRS - Data is copied three times in the primary region using ZRS and then copied to secondary region
 3. Storage/Data in the secondary region is not accessible until you perform a failover
 4. Additional option to allow read-only access to secondary region
 1. RA-GRS
 2. RA-GZRS

Azure Storage - Security

1. Secured using Azure AD
 1. Give users within your AD tenancy direct access to the Storage either on the Storage account level or Blob level
2. Secured using Shared Key
 1. A key is distributed to the consumer and consumer will use the Key together with the storage endpoint to access
3. Secured using Shared Access Signature (SAS)
 1. Usage of SAS through Azure Storage Explorer

Azure Storage – how do you look at your storage contents

Two ways:

1. Through Azure Portal – use the Storage Explorer
2. Download Azure Storage Explorer and access using either your login or SAS

Azure Storage Explorer - <https://aka.ms/portalfx/downloadstorageexplorer>

Azure Storage Emulator - Azurite

While building an application, you may want to work offline or don't want to start paying for storage you are not using. In this scenario, you can use "Azurite" which is an Azure Storage Emulator that runs on your local machine (or wherever you install it).

1. It is cross-platform – runs on Windows, Linux and MacOS
2. Doesn't emulate File Share – only Blob, Queue and Table

Azurite - <https://learn.microsoft.com/en-us/azure/storage/common/storage-use-azurite?tabs=visual-studio>

Fun time - Create an Azure Storage Account

1. Go to Azure Portal – <https://portal.azure.com>
2. Search for “Resource Group” and click on “Create”
 1. Give it an appropriate name
 2. Under region, select “Australia East”.
 3. Click “Next” to go the “Tags” page
 1. It’s good practice to always add tags for your resource group and resources.
 2. You can skip it for this demo but make it a good practice to add some tags
 3. Some example tags:
 1. ENV:{NON-PROD|UAT|PROD|STAGING}
 2. APPNAME:{NAME_OF_APP}
3. Wait for the resource group to be created and go to the Resource Group
4. Click on the “Create” button and search for “storage account”
5. Follow the prompt to create the storage account and I will guide you through it
6. Once the Storage account is created, create two blob containers called “original-image” and “small-image”

Fun time – Build an app to add a file to a blob container

1. Launch Visual Studio Pro/Ent/Code – your choice
2. If you are using Visual Studio Pro/Ent
 1. Create a console app and called it **demostorageapp**
 2. Install the Nuget Package **Azure.Storage.Blobs**
3. If you are using Visual Studio Code
 1. Launch a terminal
 2. Type this command to create a new console app: **dotnet new console –o demostorageapp**
 3. Add the Azure Blob Storage SDK: **dotnet add package Azure.Storage.Blobs**

Fun time – Build an app to add a file to a blob container – cont'd

NuGet Packages to manipulate Azure Storage:

1. Blob – Azure.Storage.Blobs
2. Queue – Azure.Storage.Queues
3. Table – Azure.Data.Tables

Additional work:

1. Resize the image in the “original-image” container smaller and save it into the “small-image” container
2. Read from the table and display message on screen
3. How do you get rid of the message from the queue?