# Azure MySQL Developer Guide Topics

## SOW Summary

### Background

Microsoft needs help from Solliance to build a typical developer journey for MySQL developers who build and deploy applications for Azure. Microsoft has a MySQL engineering team with a new PM focusing on MySQL Developer content. In addition to that the OSS Engineering has another PM focusing on developer audience. The ask for Solliance is to collaborate and brainstorm with Microsoft engineering and marketing stakeholders on how to drive more MySQL applications to Azure.

Microsoft has a MySQL engineering team with a new PM focusing on MySQL Developer content. In addition to that the OSS Engineering has another PM focusing on developer audience. The ask for Solliance is to collaborate and brainstorm with Microsoft engineering and marketing stakeholders on how to drive more MySQL applications to Azure.

### SOW topic suggestions

• We should keep “What is Azure Database for MySQL single and flexible server” and the features very light because we already have tons of documentation about it

• The journey typically starts with creating a server and connecting to it, setting up local environment, informing the developer what tools they need to set up their environment

• How to deploy a database using GitHub Actions, supported versions, limitations

• App development best practices, operational and performance best practices

• Web applications using App Service

• When to use Spring Cloud, when to use AKS

• Cloud Native Apps, microservices architecture. The big picture

• Building secure applications (include things like vnet, private link, ATP, specific features of MySQL etc.)

• Maybe talk a little bit about RESTFul API with Azure Functions, DB caching with Redis. No need to go into details but we should open the developer’s mind to understand what additional value Azure can give to them

• Developers love checklists or cheat sheet like of structure as well

## Guide strategy

1. Shoot for a guide length of 100 pages. Need to prioritize topics.
2. The guide is meant to address the needs of L200-L300 developer. Focus on L300. Code example. Assume the developer is a senior developer or architect.
3. Focus on Azure infrastructure and services that would functionally match other hosting vendors. Explain unique benefits.
4. Azure specific. Avoid explanations of common non-MS development tools and services that are found in all vendor cloud-hosted solutions. e.g. GitHub repo. Assume the reader knows how and why to use those tools.
5. The guide should start with a simple solution and move to advanced enterprise architecture topics. The beginning of the document should have the basics and move to the advance topics further in the document.
6. Developers are usually happy with the development tools, unless the new development tool offers substantial productivity improvements given the environment.

## Guide scenarios

Scenario 1 – Start up

Scenario 2 - Enterprise

Contoso has used on-premises and AWS infrastructure resources to host applications and systems. Recently, Contoso has been exploring Azure cloud services. Mary, a senior developer/architect, has just received guidance from her management to explore hosting the next PHP/Java/MySQL project in their new corporate Azure environment. Management expects a hosting environment comparison report and Getting Started guide at the end of the project. If all goes well with the new project deployment, management would like to migrate several enterprise applications to Azure cloud. They are looking for guidance from Mary on how to make sure their applications are enterprise ready in Azure.

## Outstanding Questions

1. How much of the document should be devoted to Azure fundamentals?

## Existing MySQL developer content

This section will contain links to existing tools and content.

- https://azure.microsoft.com/mediahandler/files/resourcefiles/developer-s-guide-to-getting-started-with-microsoft-azure-database-for-mysql/Azure%20Database%20for%20MySQL%20e-book.pdf

- https://docs.microsoft.com/en-us/azure/developer/java/spring-framework/configure-spring-data-r2dbc-with-azure-mysql

- https://docs.microsoft.com/en-us/azure/app-service/tutorial-php-mysql-app?pivots=platform-linux

- https://docs.microsoft.com/en-us/azure/mysql/connect-python

- <https://www.amazon.com/s?k=mysql+developers&crid=67S1YLX11AIS&sprefix=mysql+developer%2Caps%2C401&ref=nb_sb_ss_ts-doa-p_1_15>

<https://docs.microsoft.com/en-us/azure/mysql/concepts-limits>

<https://dev.mysql.com/doc/mysql-reslimits-excerpt/5.6/en/limits.html>

<https://docs.microsoft.com/en-us/azure/mysql/concepts-pricing-tiers#storage>

<https://docs.microsoft.com/en-us/azure/mysql/concept-reserved-pricing>

<https://docs.microsoft.com/en-us/azure/mysql/flexible-server/tutorial-php-database-app>

## Sample application ideas

1. Gaming

# TOC

## Getting started

## Azure Subscriptions

## Azure Portal basics

### Azure Resource Groups

### Azure Regions and Availability Zones

## Managing certificates in Azure

## Azure VMs

### Azure VM Scale Sets

### Azure Batch

## Setting up an development environment

### Windows

#### Visual Studio

#### Visual Studio Code IntelliJ

### Linux

### MacOS

#### Visual Studio Code

## Azure Language Support

C#, Java, JavaScript, Python, and PowerShell.

## Connection SDKs

### NodeJS

https://linuxhint.com/connect\_mysql\_nodejs/

### Java

### Azure JDK support

### Maven

### Windows / .NET

## Azure data storage

### Blob Storage

### Queues

### File Storage

### Managed Disks

### Tables

### Azure Data Lake Storage

## Azure continuous deployment tools

### ARM templates

### Azure Blueprints

### PowerShell

### Azure DevOps

### GitHub Actions

### Terraform

### Azure CLI

## Unsupported features

### Azure service tiers

## Azure Deployment Targets

### App Service

### Linux Support

### Micro Services

### Containers

### Best Practices (slots / app service; load balancer)

## Azure Database for MySQL

### General limitations

### Azure Database for MySQL limitations

#### Supported Versions

### Configure SSL

### Configure server admin user

### Single Server

### Flexible Server

## Types of MySQL Applications

### Web app

### Function app

### Microservices

### Services

### Batch Jobs

## Network configuration basics

### VNet/Subnet

### Firewall

### NSG

### Routing

### Load balancing

## Managing your MySQL environment

### App insights / Logging

### Debugging

### AZ-400 topics

### Back and restore your database

## Testing applications

### Azure DevTest Labs

### Azure Lab Services

## Securing and protecting applications

### Managed Service Identities

### Key Vault

### Security Baseline

### Azure Security Center

### Azure Defender

### Azure Sentinel

### Log Analytics

### Data encryption

## Application monitoring

### Azure Monitor

### Azure Service Health

### Azure Logic App

## Azure Kubernetes Service

## Azure Container Registry

## Azure Container Instances

## Scaling and run-time performance

### Caching (deterministic methods, result sets)

### Azure Cache for Redis

### Scaling

### Data replication

### Traffic Manager

### Azure Front Door

### Azure Content Delivery Network

### Database performance tuning

### Connection pooling

## Source code control and versioning

### Code

### Database

## Architecture Patterns

### Multi-node/instance

### Geo-routing (Front door)

## Azure pricing and cost containment

### Pay as you go

### Pricing calculator

### Total Cost of Ownership

## Azure migrations

DB Migration