

Tutorial 1 – Azure storage & compute

Sette opp Tweetpubliserer

- Clone prosjektet GeekRetreatInit (https://ksitv.visualstudio.com/DefaultCollection/GeekRetreat2015/_git/GeekRetreatInit)
- Importer solution in VS og restore nuget pakker.
- Sett startup prosjekt til TweetPublishService og start debugging.
 - Åpne Azure Compute Emulator og se at Tweets blir publisert.
 - Åpne Cloud Explorer i VS og se at meldinger kommer inn i «tweetsqueue»

Håndter meldinger fra kø

- Lag nytt «Azure Cloud Service»-prosjekt. («TweetHandlerService»)
 - Legg til én «Worker Role» («TweetHandler»)
 - Implementer logikk for å lese melding fra køen. Kan f.eks. logge meldingene til konsoll med Trace.

Tips1: koble til development storage med følgende logikk. Dette må skiftes ut senere før man eventuelt publiserer til Azure.

```
CloudStorageAccount storageAccount= CloudStorageAccount.DevelopmentStorageAccount;  
CloudQueueClient queueClient = _storageAccount.CreateCloudQueueClient();  
CloudQueue cloudQueue = _queueClient.GetQueueReference("tweetsqueue");  
_cloudQueue.CreateIfNotExists();
```

Tips2: Les meldinger fra køen og husk å slette etter håndteringen er fullført

```
var msg = await _cloudQueue.GetMessageAsync(cancellationToken);  
await _cloudQueue.DeleteMessageAsync(msg, cancellationToken);
```

Last opp meldingene fra køen som blobs.

- Koble nå til en ny «blob container» («handledtweets»)
- Set to startup prosjekter - TweetHandlerService og TweetPublisherService.
- Debug systemet og se at meldingene blir håndtert og lastet opp som blobs.

Tips3: Bruk samme storage account som i forrige steg, men lag nå en blob client.

```
CloudBlobClient blobClient = _storageAccount.CreateCloudBlobClient();  
CloudBlobContainer blobContainer =  
    blobClient.GetContainerReference("handledtweets");  
blobContainer.CreateIfNotExists();
```

Tutorial 1 – Resource Manager Provisioning

Provisjoner Azure-resurser for hele systemet

- Sørg for å ha Azure Powershell modulene installert og klare til bruk.
(<https://azure.microsoft.com/en-us/documentation/articles/powershell-install-configure/>)
- Koble til «Azure subscription», bruk enten personlig konto med MSDN, eller «ITverket playground» subscriptionen. Bruk de 3 følgende commandletsene.

1. Logg inn med «work account» eller Microsoft account, vha. Add-AzureAccount (Azure AD)

```
λ Add-AzureAccount
```

Id	Type	Subscriptions
anders.kofoed@outlook.com	User	2b60a02f-805b-49e9-812a-04304091867b 8c006a3c-f5b0-4b23-a99c-bfc2fb741be1

2. List alle tilgjengelige Azure subscriptions med Get-AzureSubscription

```
λ Get-AzureSubscription | select SubscriptionId, SubscriptionName | Format-List
```

SubscriptionId : 8c006a3c-f5b0-4b23-a99c-bfc2fb741be1
SubscriptionName : Visual Studio Enterprise med MSDN

SubscriptionId : 2b60a02f-805b-49e9-812a-04304091867b
SubscriptionName : Windows Azure Cloud Essentials

SubscriptionId : dca72209-8a25-4ce3-83e3-51ec04e5ac2a
SubscriptionName : ITverket Drift Pay-As-You-Go

SubscriptionId : 00f3b613-a2d4-4828-97f4-a28d906969c0
SubscriptionName : ITverket Azure Playground Pay-As-You-Go

3. Velg en subscription med Select-AzureSubscription

```
λ Select-AzureSubscription -SubscriptionId 8c006a3c-f5b0-4b23-a99c-bfc2fb741be1
```

- Kjør scriptet <repos-dir>/GeekRetreatItnit/AzureManagement/Create-GRResourceGroup.ps1
- Velg et kort gruppenavn eller bruk gruppenummer (må være kort grunnet begrensninger på «storage account» navn)
- Følg med at alt går greit, og sjekk gjerne i portalen at resursene har blitt provisjonert som ønsket. ☺