



**Cloud Development, 2021 fall.
Lab 11**

Report

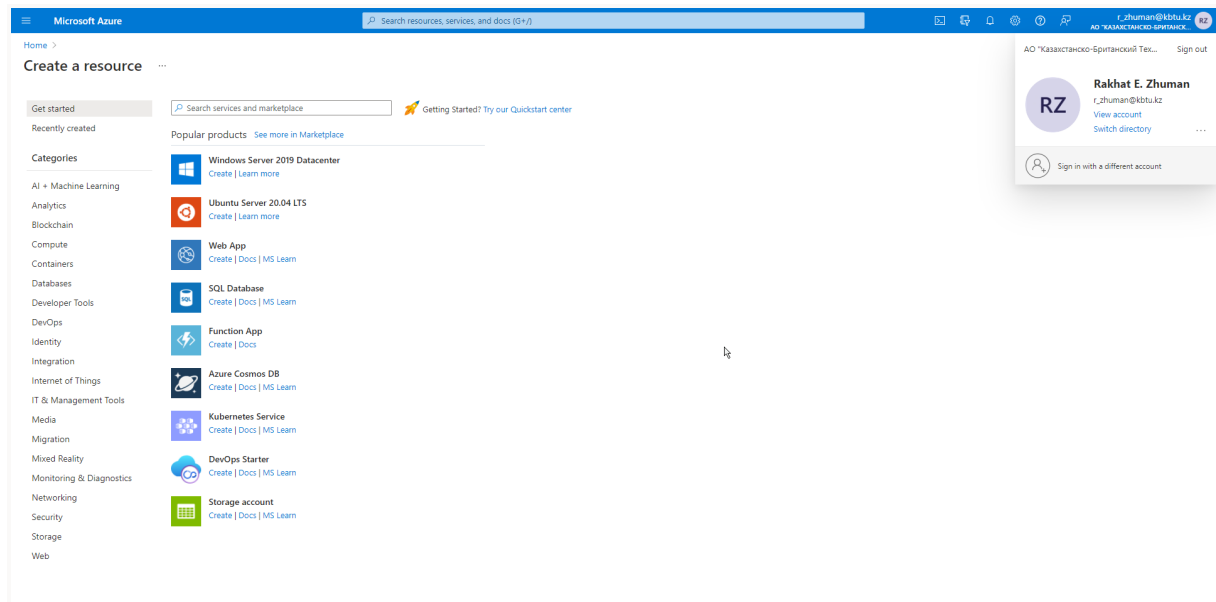
**Asynchronously processing messages by using Azure Queue
Storage**

made by Zhuman Rakhat

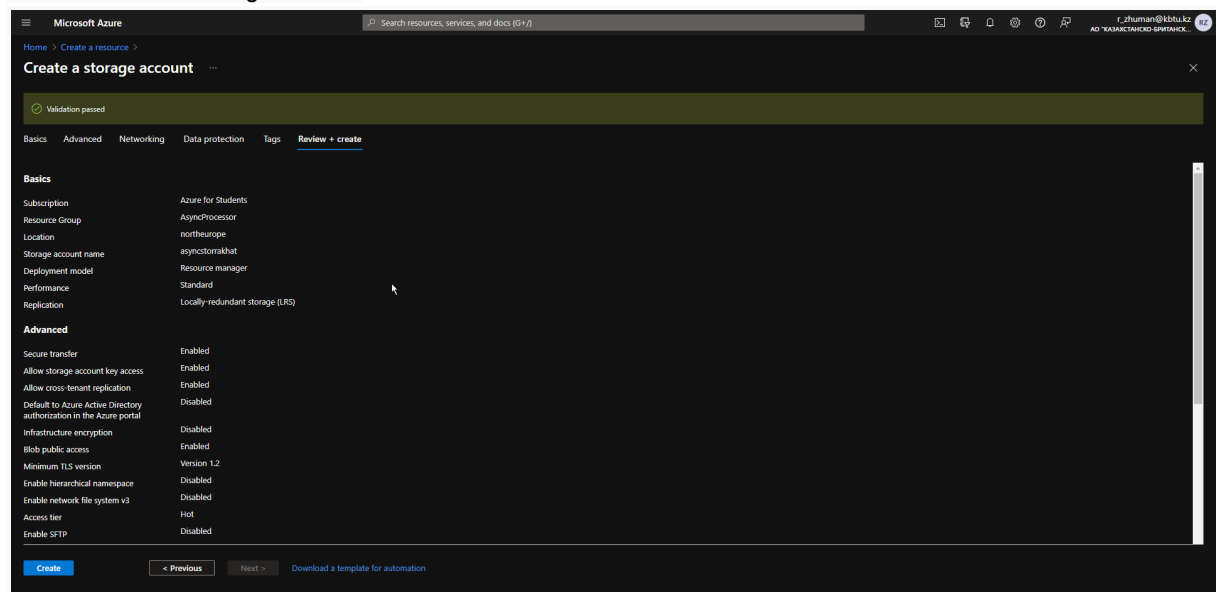
Almaty 2021

Exercise 1: Creating an Azure App Service resource by using a Docker container image

Task 1: Open the Azure portal



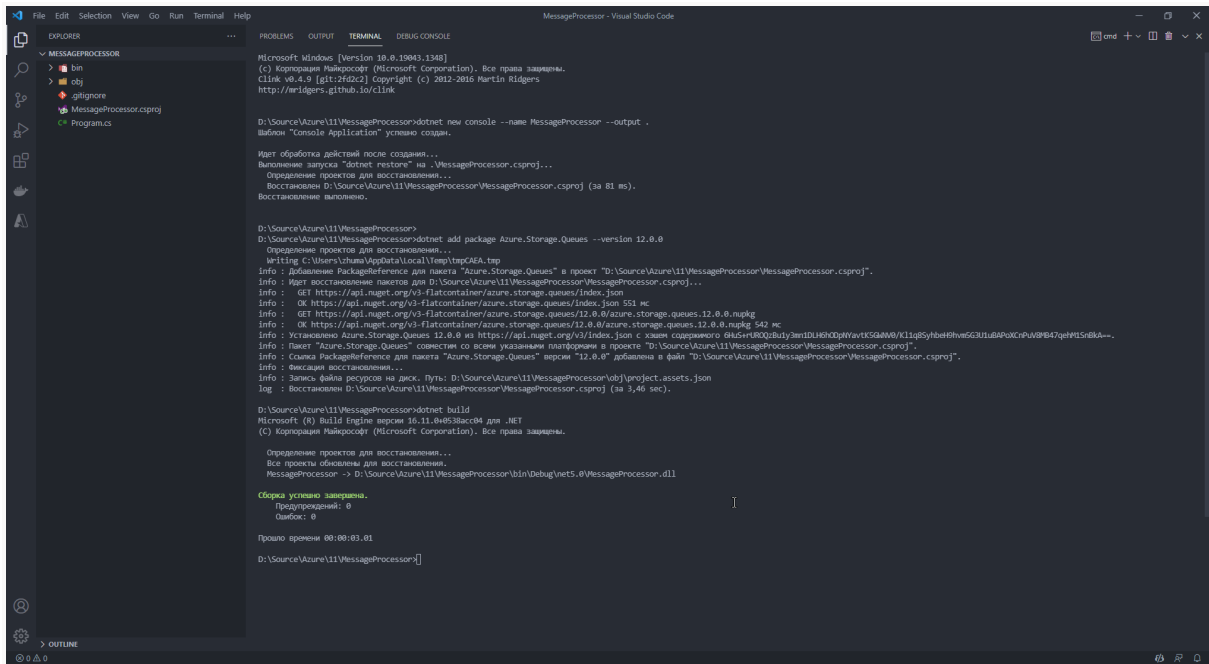
Task 2: Create a Storage account



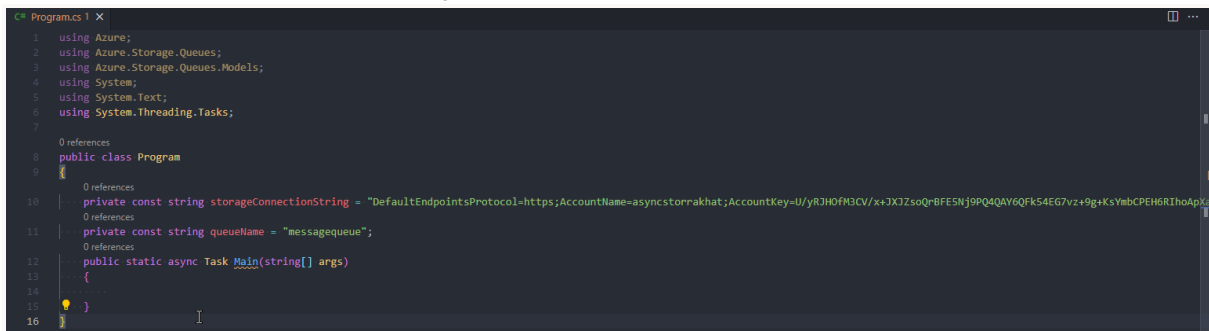
DefaultEndpointsProtocol=https;AccountName=asynctorrakhat;AccountKey=U/yRJHOfM3CV/x+JXJZsoQrBFE5Nj9PQ4QAY6QFk54EG7vz+9g+KsYmbCPEH6RIhoApXaP8NObejUMsPRI33w==;EndpointSuffix=core.windows.net

Exercise 2: Configure the Azure Storage SDK in a .NET project

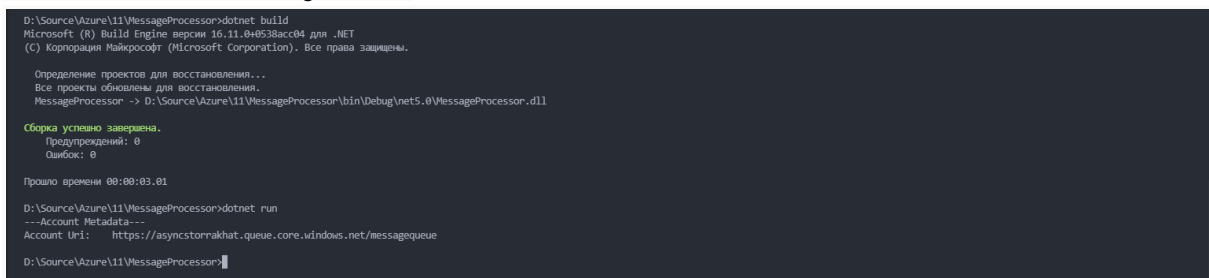
Task 1: Create a .NET project



Task 2: Write code to access Azure Storage

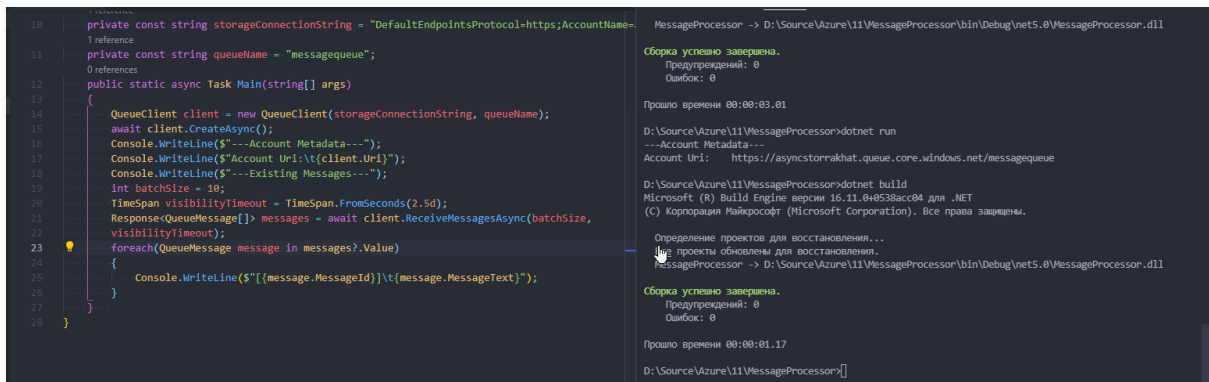


Task 3: Validate Azure Storage access



Exercise 3: Add messages to the queue

Task 1: Write code to access queue messages



Task 2: Test message queue access

```
D:\Source\Azure\11\MessageProcessor>
D:\Source\Azure\11\MessageProcessor>dotnet run
---Account Metadata---
Account Uri:   https://asyncstorakhat.queue.core.windows.net/messagequeue
---Existing Messages---

D:\Source\Azure\11\MessageProcessor>
```

Идентификатор	Текст сообщения	Время вставки	Время окончания срока действия	Число выводов из очереди	Размер
1e20b0e2-7447-4efa-b8c5-341cbbd3cc71	Hello World	2021-12-04T06:19:12.000Z	2021-12-04T18:19:12.000Z	0	11 B

```
D:\Source\Azure\11\MessageProcessor>dotnet run
---Account Metadata---
Account Uri:   https://asyncstorakhat.queue.core.windows.net/messagequeue
---Existing Messages---
[1e20b0e2-7447-4efa-b8c5-341cbbd3cc71] Hello World

D:\Source\Azure\11\MessageProcessor>
```

Task 3: Delete queued messages

```
D:\Source\Azure\11\MessageProcessor>dotnet run
---Account Metadata---
Account Uri:   https://asyncstorakhat.queue.core.windows.net/messagequeue
---Existing Messages---
[1e20b0e2-7447-4efa-b8c5-341cbbd3cc71] Hello World

D:\Source\Azure\11\MessageProcessor>
```

Идентификатор	Текст сообщения	Время вставки	Время окончания срока действия	Число выводов из очереди	Размер
No data available in table					

Exercise 4: Queue new messages by using .NET

Task 1: Write code to create queue messages

```
12 public static async Task Main(string[] args)
13 {
14     QueueClient client = new QueueClient(storageConnectionString, queueName);
15     await client.CreateAsync();
16     Console.WriteLine($"---Account Metadata---");
17     Console.WriteLine($"Account Uri: {client.Uri}");
18     Console.WriteLine($"---Existing Messages---");
19     int batchSize = 10;
20     TimeSpan visibilityTimeout = TimeSpan.FromSeconds(2.5d);
21     Response<QueueMessage[]> messages = await client.ReceiveMessagesAsync(batchSize,
22     visibilityTimeout);
23     foreach(QueueMessage message in messages.Value)
24     {
25         Console.WriteLine($"[{message.MessageId}] {message.MessageText}");
26         await client.DeleteMessageAsync(message.MessageId, message.PopReceipt);
27     }
28     Console.WriteLine($"---New Messages---");
29     string greeting = "Hi, Developer!";
30     await client.SendMessageAsync(Convert.ToBase64String(Encoding.UTF8.GetBytes(greeting)));
31     Console.WriteLine($"Sent Message: {greeting}");
32 }
33 }
```

```
Account Uri: https://asyncstorakhat.queue.core.windows.net/messagequeue
---Existing Messages---

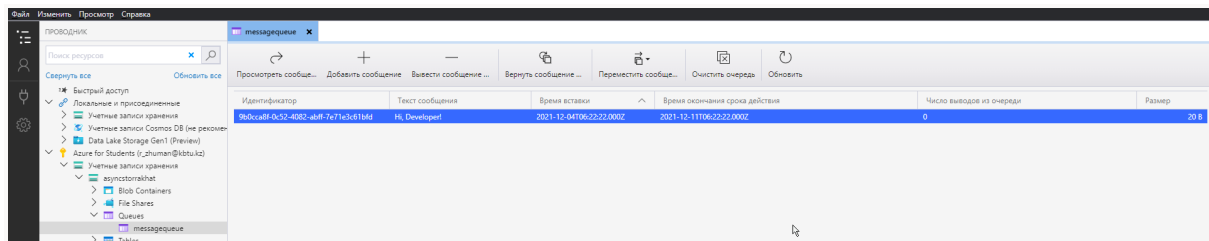
D:\Source\Azure\11\MessageProcessor>dotnet run
---Account Metadata---
Account Uri: https://asyncstorakhat.queue.core.windows.net/messagequeue
---Existing Messages---
[1e20b0e2-7447-4efa-b8c5-341cbbd3cc71] Hello World

D:\Source\Azure\11\MessageProcessor>dotnet run
---Account Metadata---
Account Uri: https://asyncstorakhat.queue.core.windows.net/messagequeue
---Existing Messages---
[1e20b0e2-7447-4efa-b8c5-341cbbd3cc71] Hello World

D:\Source\Azure\11\MessageProcessor>dotnet run
---Account Metadata---
Account Uri: https://asyncstorakhat.queue.core.windows.net/messagequeue
---Existing Messages---
---New Messages---
Sent Message: Hi, Developer!

D:\Source\Azure\11\MessageProcessor>
D:\Source\Azure\11\MessageProcessor>
D:\Source\Azure\11\MessageProcessor>
```

Task 2: View queued messages by using Storage Explorer



Exercise 5: Clean up your subscription

Task 2: Delete a resource group

