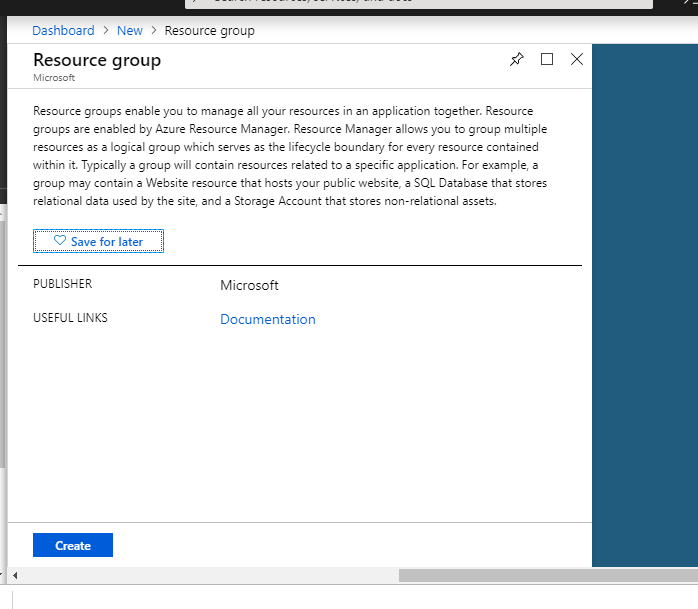
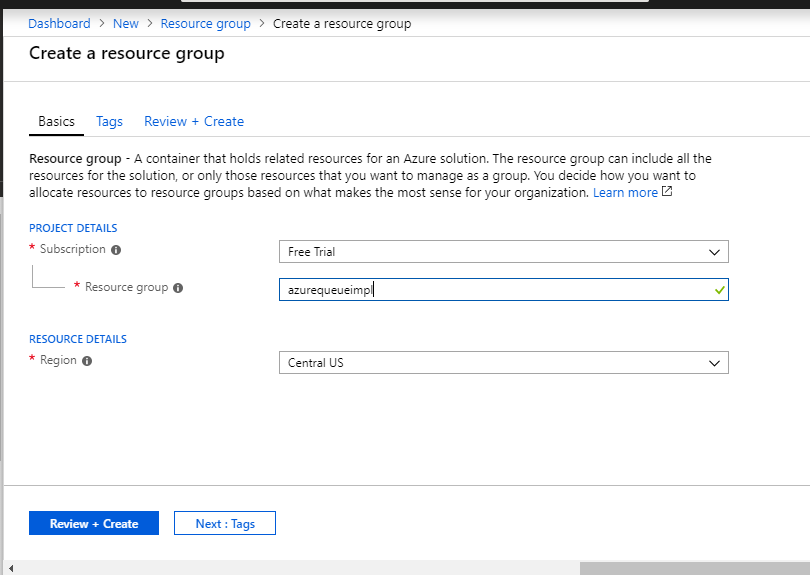
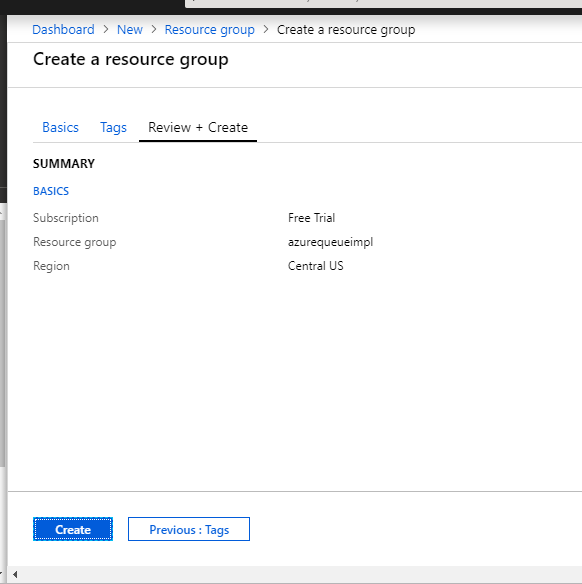
**Implementation of Azure Queue**

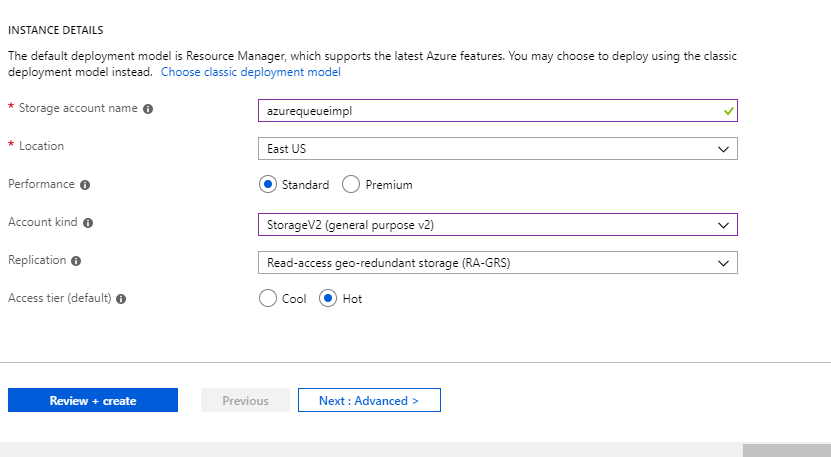
Step 1 : Creation of Resource Group

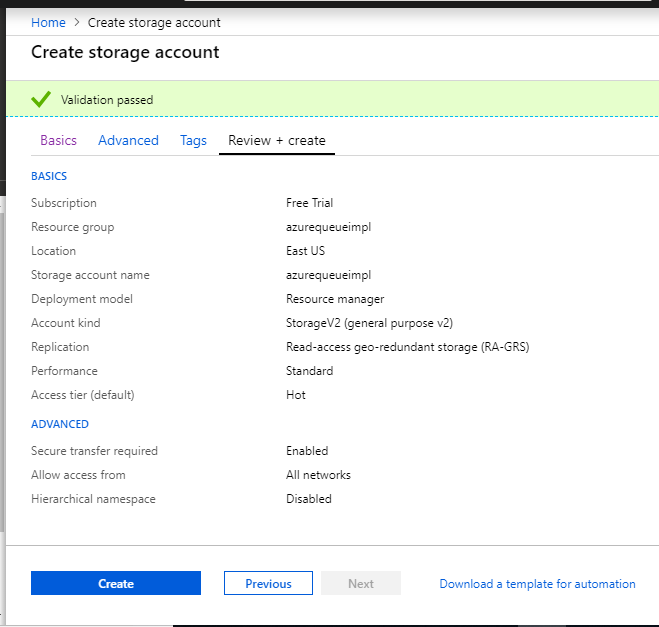






Step 2 : Creation of Storage account



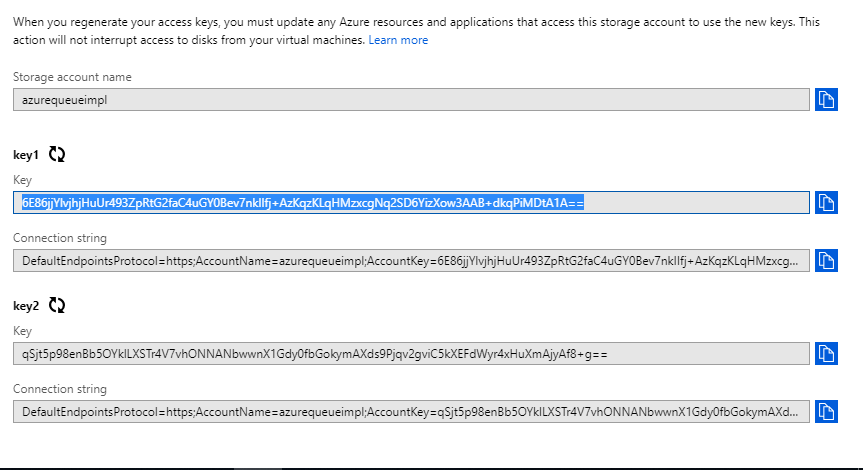


**Details for storage account:**

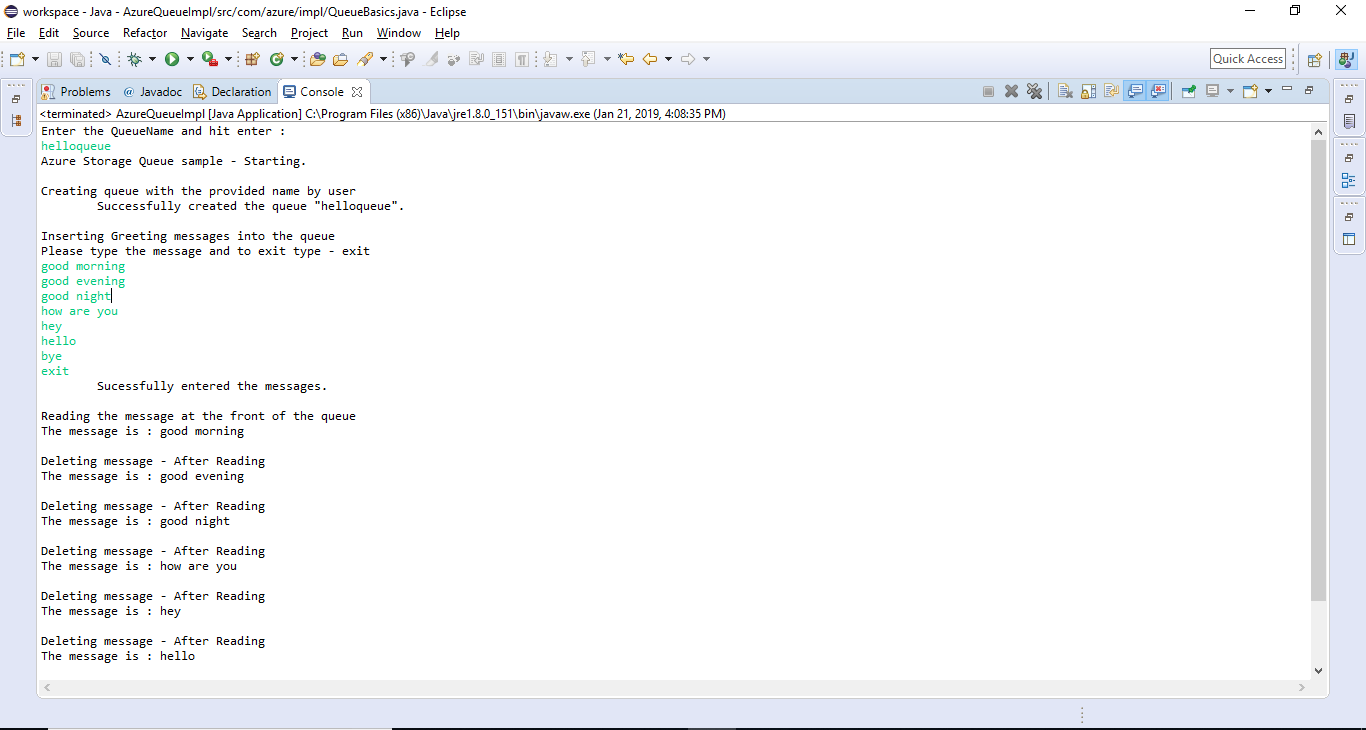
Storage Account Name: azurequeueimpl

Key1 : 6E86jjYlvjhjHuUr493ZpRtG2faC4uGY0Bev7nkIIfj+AzKqzKLqHMzxcgNq2SD6YizXow3AAB+dkqPiMDtA1A==

6E86jjYlvjhjHuUr493ZpRtG2faC4uGY0Bev7nkIIfj+AzKqzKLqHMzxcgNq2SD6YizXow3AAB+dkqPiMDtA1A=

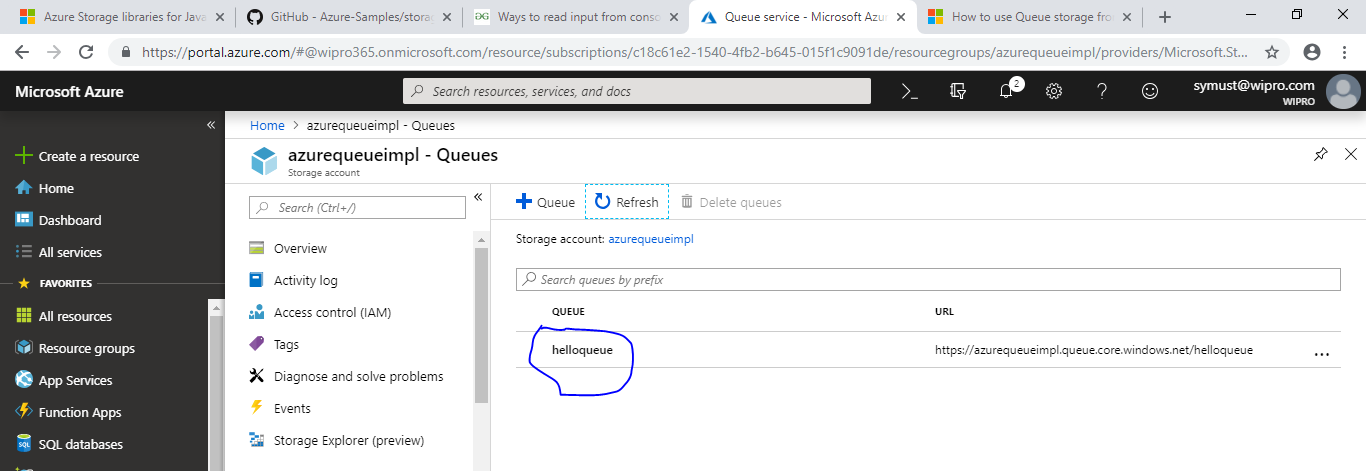


Program execution and Outputs :

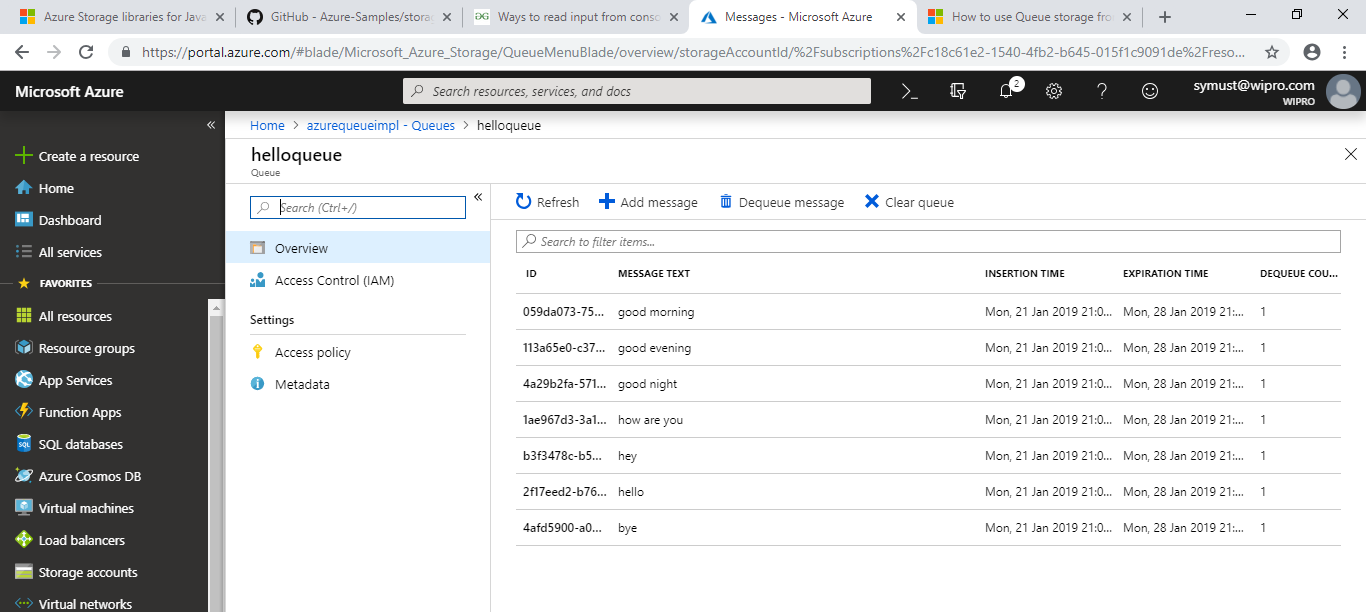


Verification from the Portal

1. Creation of Queue with exact same name



Messages inside the queue :



Here is the Source code :

AzureQueueImpl.java

---------------------------

**import** com.azure.impl.QueueBasics;

**import** java.io.BufferedReader;

**import** java.io.IOException;

**import** java.io.InputStreamReader;

**public** **class** AzureQueueImpl {

**public** **static** **void** main(String[] args) {

// **TODO** Auto-generated method stub

BufferedReader reader =

**new** BufferedReader(**new** InputStreamReader(System.***in***));

System.***out***.println("Enter the QueueName and hit enter :");

**try**{

String name = reader.readLine();

QueueBasics.*runSamples*(name);

}

**catch**(Exception e)

{

e.printStackTrace();

}

}

}

QueueBasics.java

-----------------------------------

**package** com.azure.impl;

**import** java.io.IOException;

**import** java.util.Scanner;

**import** java.io.InputStream;

**import** java.io.PrintWriter;

**import** java.io.StringWriter;

**import** java.net.URISyntaxException;

**import** java.security.InvalidKeyException;

**import** java.util.EnumSet;

**import** java.util.Properties;

**import** java.util.Scanner;

**import** java.util.UUID;

**import** com.microsoft.azure.storage.CloudStorageAccount;

**import** com.microsoft.azure.storage.StorageException;

**import** com.microsoft.azure.storage.queue.CloudQueue;

**import** com.microsoft.azure.storage.queue.CloudQueueClient;

**import** com.microsoft.azure.storage.queue.CloudQueueMessage;

**import** com.microsoft.azure.storage.queue.MessageUpdateFields;

**public** **class** QueueBasics {

**public** **static** **void** runSamples(String quename) **throws** Exception

{

System.***out***.println("Azure Storage Queue sample - Starting.");

CloudQueueClient queueClient;

CloudQueue queue1 = **null**;

CloudQueue queue2 = **null**;

**try** {

// Create a queue client for interacting with the queue service

queueClient = *getQueueClientReference*();

// Create new queues with randomized names

System.***out***.println("\nCreating queue with the provided name by user");

// queue1 = createQueue(queueClient, DataGenerator.createRandomName("queuebasics-"));

queue1 = *createQueue*(queueClient, quename);

System.***out***.println(String.*format*("\tSuccessfully created the queue \"%s\".", queue1.getName()));

// Insert a message into the queue

System.***out***.println("\nInserting Greeting messages into the queue");

System.***out***.println("Please type the message and to exit type - exit ");

Scanner in=**null**;

String s=**null**;

**while**(**true**){

in = **new** Scanner(System.***in***);

s = in.nextLine();

**if**(s.equalsIgnoreCase("exit"))

**break**;

queue1.addMessage(**new** CloudQueueMessage(s));

}

System.***out***.println("\tSucessfully entered the messages.");

System.***out***.println("\nReading the message at the front of the queue");

**while**(**true**){

CloudQueueMessage peekedMessage = queue1.peekMessage();

**if** (peekedMessage != **null**) {

System.***out***.println("The message is : " + peekedMessage.getMessageContentAsString());

CloudQueueMessage message = queue1.retrieveMessage();

System.***out***.println("\nDeleting message - After Reading");

}

**else**{

**break**;

}

}// end of while loop

}

**catch**(Exception e){

System.***out***.println("Exception catch is :"+e);

e.printStackTrace();

}

}

**private** **static** CloudQueueClient getQueueClientReference() **throws** RuntimeException, IOException, IllegalArgumentException, URISyntaxException, InvalidKeyException {

CloudStorageAccount storageAccount;

String storageConnectionString =

"DefaultEndpointsProtocol=https;" +

"AccountName=azurequeueimpl;" +

"AccountKey=6E86jjYlvjhjHuUr493ZpRtG2faC4uGY0Bev7nkIIfj+AzKqzKLqHMzxcgNq2SD6YizXow3AAB+dkqPiMDtA1A==";

**try** {

//storageAccount = CloudStorageAccount.parse(prop.getProperty("StorageConnectionString"));

storageAccount = CloudStorageAccount.*parse*(storageConnectionString);

}

**catch** (IllegalArgumentException|URISyntaxException e) {

System.***out***.println("\nConnection string specifies an invalid URI.");

System.***out***.println("Please confirm the connection string is in the Azure connection string format.");

**throw** e;

}

**catch** (InvalidKeyException e) {

System.***out***.println("\nConnection string specifies an invalid key.");

System.***out***.println("Please confirm the AccountName and AccountKey in the connection string are valid.");

**throw** e;

}

**return** storageAccount.createCloudQueueClient();

}

**private** **static** CloudQueue createQueue(CloudQueueClient queueClient, String queueName) **throws** StorageException, RuntimeException, IOException, InvalidKeyException, IllegalArgumentException, URISyntaxException, IllegalStateException {

// Create a new queue

CloudQueue queue = queueClient.getQueueReference(queueName);

**try** {

**if** (queue.createIfNotExists() == **false**) {

**throw** **new** IllegalStateException(String.*format*("Queue with name \"%s\" already exists.", queueName));

}

}

**catch** (StorageException s) {

**if** (s.getCause() **instanceof** java.net.ConnectException) {

System.***out***.println("Caught connection exception from the client. If running with the default configuration please make sure you have started the storage emulator.");

}

**throw** s;

}

**return** queue;

}

}