The first thing I did, was exploring Amazon Web services and creating a new user
After this, I created an IAM user and added this to group(This was important so we can add policies and permissions for the user)
After the user was created and assigned to the group, I could now get my Aws_access and secret key, and also specify which region my AWS user is connected to, which mostly in this case was ‘Us-east-2’

Picture below:

```
import boto3

AWS_ID_VLORJAN = '........'
AWS_SECRETID_VLORJAN = '.......'

ConnectTosqs = boto3.resource('sqs',
aws_access_key_id=AWS_ID_VLORJAN,
    aws_secret_access_key=AWS_SECRETID_VLORJAN,
    region_name='......'
)
```

Now that I had they code to connect, I could start coding, the question i asked was do i need to use boto or boto3? After searching around web, boto3 looked as the best option as i needed to create a FIFO queue to process the first line from the text document in and in order.

Code example of how I created FIFO queue:

```
NameOfQueue = ('Vlorjan_Queue.fifo')

response = ConnectTosqs.create_queue(QueueName=NameOfQueue,
Attributes={'FifoQueue': 'true',
'ContentBasedDeduplication': 'true'})

queueResponse =
ConnectTosqs.get_queue_by_name(QueueName='Vlorjan_Queue.fifo')
```

Now I needed to to first read the “read.txt” file and then send to the SQS. What I learnt here is that i could use “with open” to open the text file, which means I dont need to close the textfile after the for each loop which is very great.

Code example:

```
#Opens the text file "Messages.txt".
with open("Messages.txt",'r') as textFile:
```

```
    '''For each loop that iterates through the file and
    sends the different lines in to the queue,
    these messages are stored together in "group1"'''
    for lines in textFile:
        queueResponse.send_message(MessageBody=lines,
MessageGroupId='Group1')
```

Task 2:

Task 2 has shorter line of code than task 1, but was a little bit more complicated to do. The first thing i managed to do was getting only 1 random message from the list. So i needed to create a list, and could now collect all messages from SQS to that list. But when writing to the file, a problem was that it didn't write all the lines from the list to the file. I was struggling a little bit on this part. So after searching around the web i found about this “a” :

```
with open("empty_text_file.txt", 'a')
```

which instead of “writing a new file and deleting the old one”, instead goes into the same file and simply adds new file.

full code of task 2 below:

```
import boto3

#connect to AWS SQS
AWS_ID_VLORJAN = '........'
AWS_SECRETID_VLORJAN = '........'

ConnectToSqs = boto3.resource('sqs',
aws_access_key_id=AWS_ID_VLORJAN,
    aws_secret_access_key=AWS_SECRETID_VLORJAN,
    region_name='......'
)

#checks and verifies that the name of the queue below is the same
as the on inside AWS SQS
GetNameOfQueue = 'Vlorjan_Queue.fifo'
collect = ConnectToSqs.get_queue_by_name(QueueName=GetNameOfQueue)

#a for each loop which says that we iterate over the message list
once.
for i in range(0, 1):
```

```
    message_list = collect.receive_messages(VisibilityTimeout=1,
MaxNumberOfMessages=10, WaitTimeSeconds=5)
    for messages in message_list:
        #now we open the empty_text_file.txt and write into it.
        with open("empty_text_file.txt", 'a') as newTextFile:
            newTextFile.write(format(messages.body))
```