CS643 Programming assignment-1

Dairui Zhang

README file

The ZIP file that submitted includes:

1. Car recognition project(name: awsTest)
2. Text recognition project(name:txtRek)
3. This README file.(includes all the code and steps’ screenshots)
4. A demo video

CODE

Code for car recognition

1. package com.example.recogniseImages;
2. import java.util.ArrayList;
3. import java.util.List;
4. import com.amazonaws.services.rekognition.AmazonRekognition;
5. import com.amazonaws.services.rekognition.AmazonRekognitionClientBuilder;
6. import com.amazonaws.services.rekognition.model.AmazonRekognitionException;
7. import com.amazonaws.services.rekognition.model.DetectLabelsRequest;
8. import com.amazonaws.services.rekognition.model.DetectLabelsResult;
9. import com.amazonaws.services.rekognition.model.Image;
10. import com.amazonaws.services.rekognition.model.S3Object;
11. import com.amazonaws.services.rekognition.model.Label;
12. import com.amazonaws.services.sqs.AmazonSQS;
13. import com.amazonaws.services.sqs.AmazonSQSClientBuilder;
14. import com.amazonaws.services.sqs.model.SendMessageRequest;
15. public class LabelImage {
16. public static void detect(AmazonRekognition rekognitionClient, String name, String s3Bucket) {
17. S3Object s3Obj = new S3Object();
18. s3Obj.withBucket(s3Bucket);
19. s3Obj.withName(name);
21. Image img = new Image();
22. img.withS3Object(s3Obj);
24. DetectLabelsRequest request = new DetectLabelsRequest();
25. request.withImage(img);
26. request.withMaxLabels(10);
27. request.withMinConfidence(90F);
29. *//Create SQS*
30. AmazonSQS sqs = AmazonSQSClientBuilder.defaultClient();

33. try {
35. DetectLabelsResult result = rekognitionClient.detectLabels(request);
36. List<Label> labels = result.getLabels();
37. System.out.println("========="+name+"==========");
38. for(Label label: labels) {
40. System.out.println("Label ::" + label.getName());
41. System.out.println("Confidence ::" + label.getConfidence());
42. if( label.getName().equals("Car")){
43. *//prepare the message send to sqs*
44. SendMessageRequest send\_msg\_request = new SendMessageRequest()
45. .withQueueUrl("https://sqs.us-east-1.amazonaws.com/333516920379/MyQueue1615875981231")
46. .withMessageBody(name)
47. .withDelaySeconds(5);
48. *//send*
49. sqs.sendMessage(send\_msg\_request);
50. }
51. }
52. System.out.println("");
53. }
54. catch(AmazonRekognitionException e){
55. e.printStackTrace();
56. }
57. }
58. public static void main(String[] args) {

61. AmazonRekognition rekognitionClient = AmazonRekognitionClientBuilder.defaultClient();
62. List<String> imageList = new ArrayList<String>();
63. for(int i = 1; i<11; i++) {
64. String a = i + ".jpg";
65. imageList.add(a);
66. }
68. String s3Bucket = "njit-cs-643";
70. for(String i : imageList) {
71. detect(rekognitionClient, i, s3Bucket); *//detection*
72. }
74. *//send -1*
75. *//prepare the message send to sqs*
76. AmazonSQS sqs = AmazonSQSClientBuilder.defaultClient();
77. SendMessageRequest send\_msg\_request = new SendMessageRequest()
78. .withQueueUrl("https://sqs.us-east-1.amazonaws.com/333516920379/MyQueue1615875981231")
79. .withMessageBody("-1")
80. .withDelaySeconds(5);
81. *//send*
82. sqs.sendMessage(send\_msg\_request);
83. }
84. }

Code for text recognition

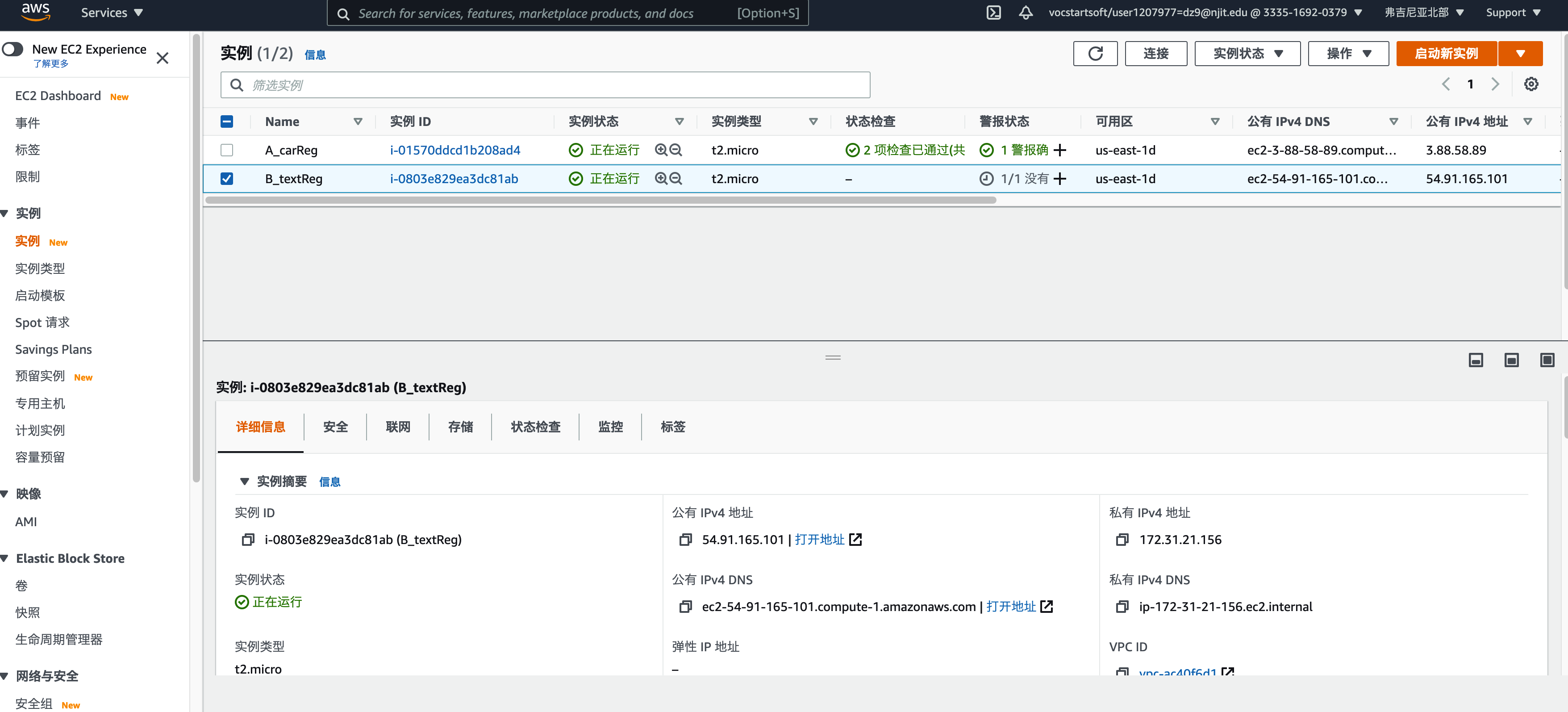
1. package aws.example.rekognition.image;
2. import java.io.File;
3. import java.io.FileNotFoundException;
4. import java.io.PrintStream;
5. import java.util.ArrayList;
6. import java.util.List;
7. import com.amazonaws.services.rekognition.AmazonRekognition;
8. import com.amazonaws.services.rekognition.AmazonRekognitionClientBuilder;
9. import com.amazonaws.services.rekognition.model.AmazonRekognitionException;
10. import com.amazonaws.services.rekognition.model.DetectTextRequest;
11. import com.amazonaws.services.rekognition.model.DetectTextResult;
12. import com.amazonaws.services.rekognition.model.Image;
13. import com.amazonaws.services.rekognition.model.S3Object;
14. import com.amazonaws.services.rekognition.model.TextDetection;
15. import com.amazonaws.services.sqs.AmazonSQS;
16. import com.amazonaws.services.sqs.AmazonSQSClientBuilder;
17. import com.amazonaws.services.sqs.model.Message;
18. public class DetectText {
19. public static List<String> detect(AmazonRekognition rekognitionClient, String name, String s3Bucket) throws FileNotFoundException {
20. S3Object s3Obj = new S3Object();
21. s3Obj.withBucket(s3Bucket);
22. s3Obj.withName(name);
24. Image img = new Image();
25. img.withS3Object(s3Obj);
27. DetectTextRequest request = new DetectTextRequest();
28. request.withImage(img);
30. List<String> resultList = new ArrayList<String>();
32. try {
34. DetectTextResult result = rekognitionClient.detectText(request);
36. List<TextDetection> textDetections = result.getTextDetections();
37. if(!textDetections.equals(null)) {
38. resultList.add("========="+name+"==========");
39. for(TextDetection text: textDetections) {
40. resultList.add("Detected: " + text.getDetectedText());
41. resultList.add("Type:" + text.getType());}
42. resultList.add(" ");
43. }
44. resultList.add("------");
45. return resultList;
46. }
47. catch(AmazonRekognitionException e){
48. e.printStackTrace();
49. }
50. return resultList;
51. }
52. public static void main(String[] args) throws FileNotFoundException {
53. *//Prepare a list to save images*
54. List<String> imageList = new ArrayList<String>();
55. *//Prepare a status to know whether we should continue to get the message from sqs*
56. String Astatus = "stillRun"; *//when it becomes "-1", the loop will end.*
58. *//create a sqs*
59. AmazonSQS sqs = AmazonSQSClientBuilder.defaultClient();
61. *//get the message from sqs*
62. while(!Astatus.equals("-1")) {
64. List<Message> messages = sqs.receiveMessage("https://sqs.us-east-1.amazonaws.com/333516920379/MyQueue1615875981231").getMessages();
65. System.out.println(messages);
67. for (Message picIndex : messages) {
68. String pic = picIndex.getBody();*//get all the message body*
69. System.out.println(pic);
70. if(pic.equals("-1")) {
71. Astatus = "-1";
72. break;
73. }else {
74. imageList.add(pic);*//save the images}*
75. }
76. }
77. }
79. AmazonRekognition rekognitionClient = AmazonRekognitionClientBuilder.defaultClient();
80. String s3Bucket = "njit-cs-643";
82. *//make a txt*
83. String path = "/Users/zhangdairui/Desktop/";
84. //for instance: String path = "/home/ec2-user/";
85. String filepath = path + "output.txt";
86. File file1 = new File(filepath);
87. PrintStream ps = new PrintStream(file1);
88. System.setOut(ps);
90. for(String i : imageList) {
91. try {
92. List<String> result = detect(rekognitionClient, i, s3Bucket); *//text recognition*
93. for(String r: result) {
94. ps.println(r);
95. }
96. } catch (FileNotFoundException e) {
97. e.printStackTrace();
98. }
99. }
100. }
101. }

The output file



Steps

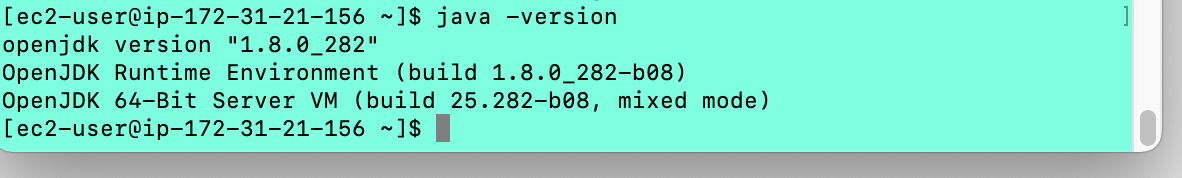
1. Prepare EC2 instances



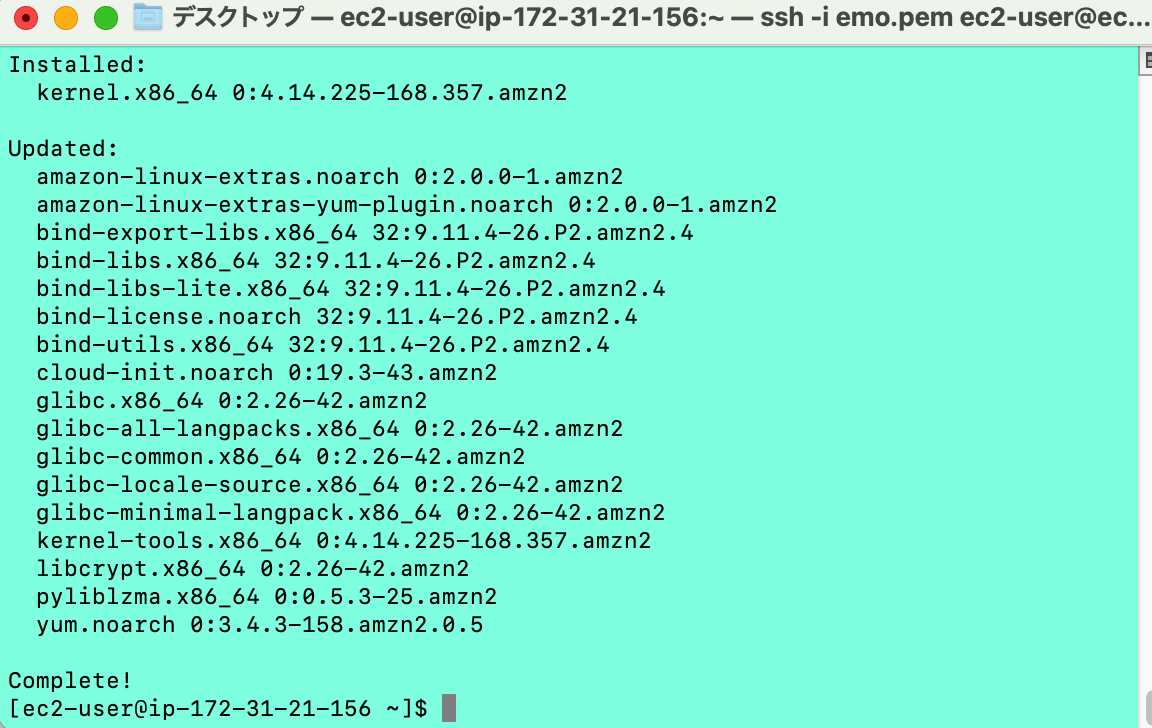
1. Connect to EC2 instance:



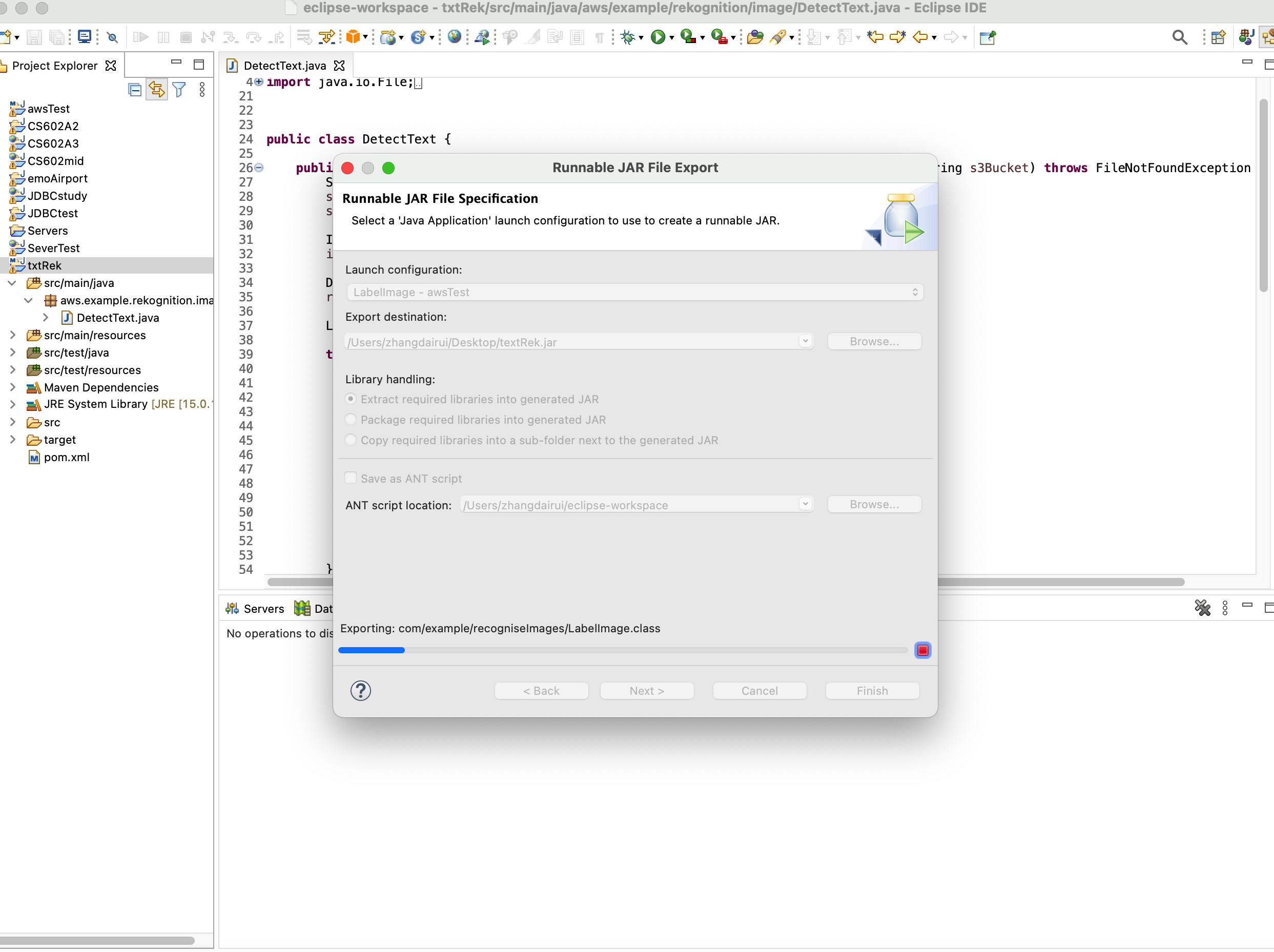
Install JDK



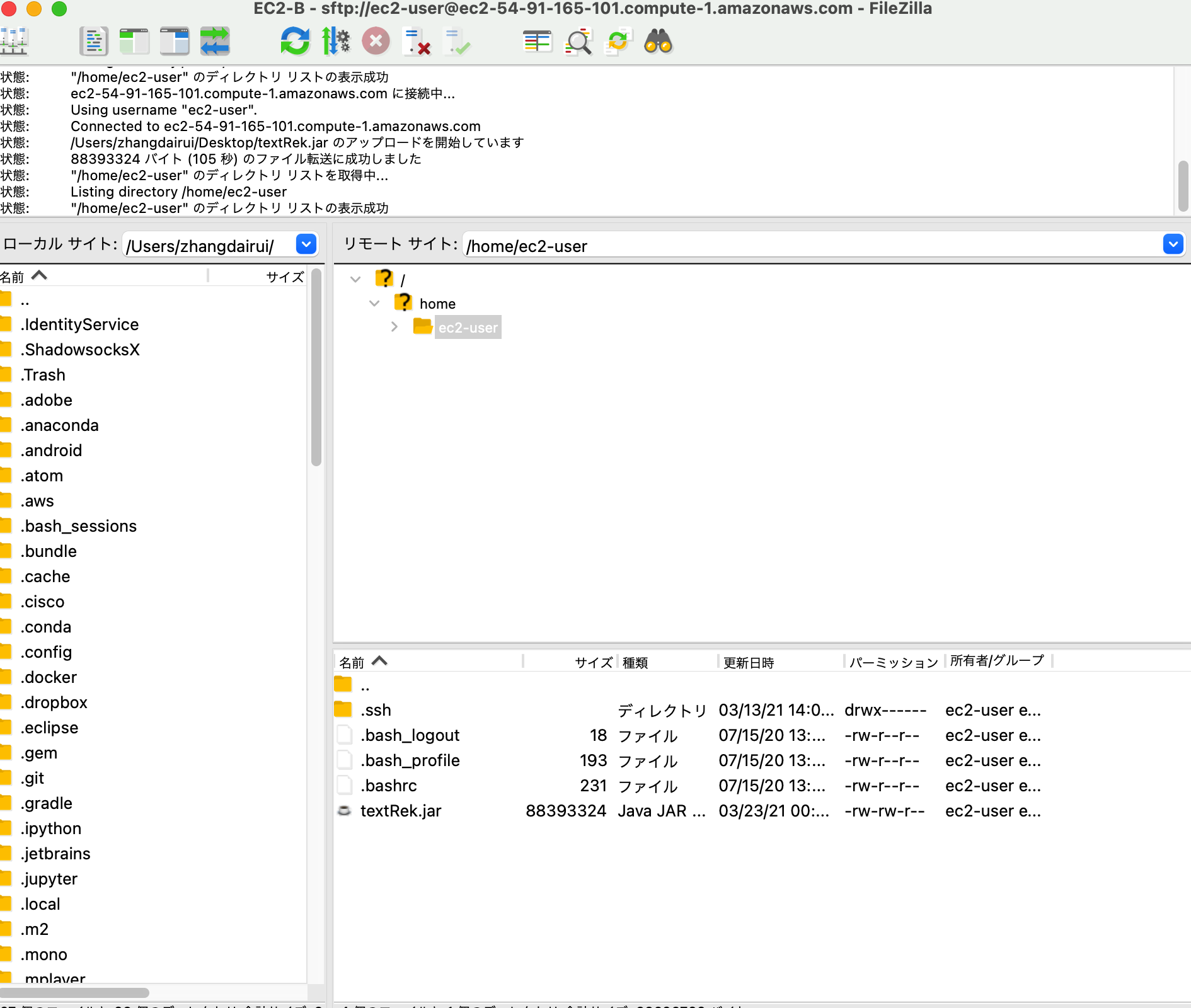
1. sudo yum update to make sure all the environment is the latest.



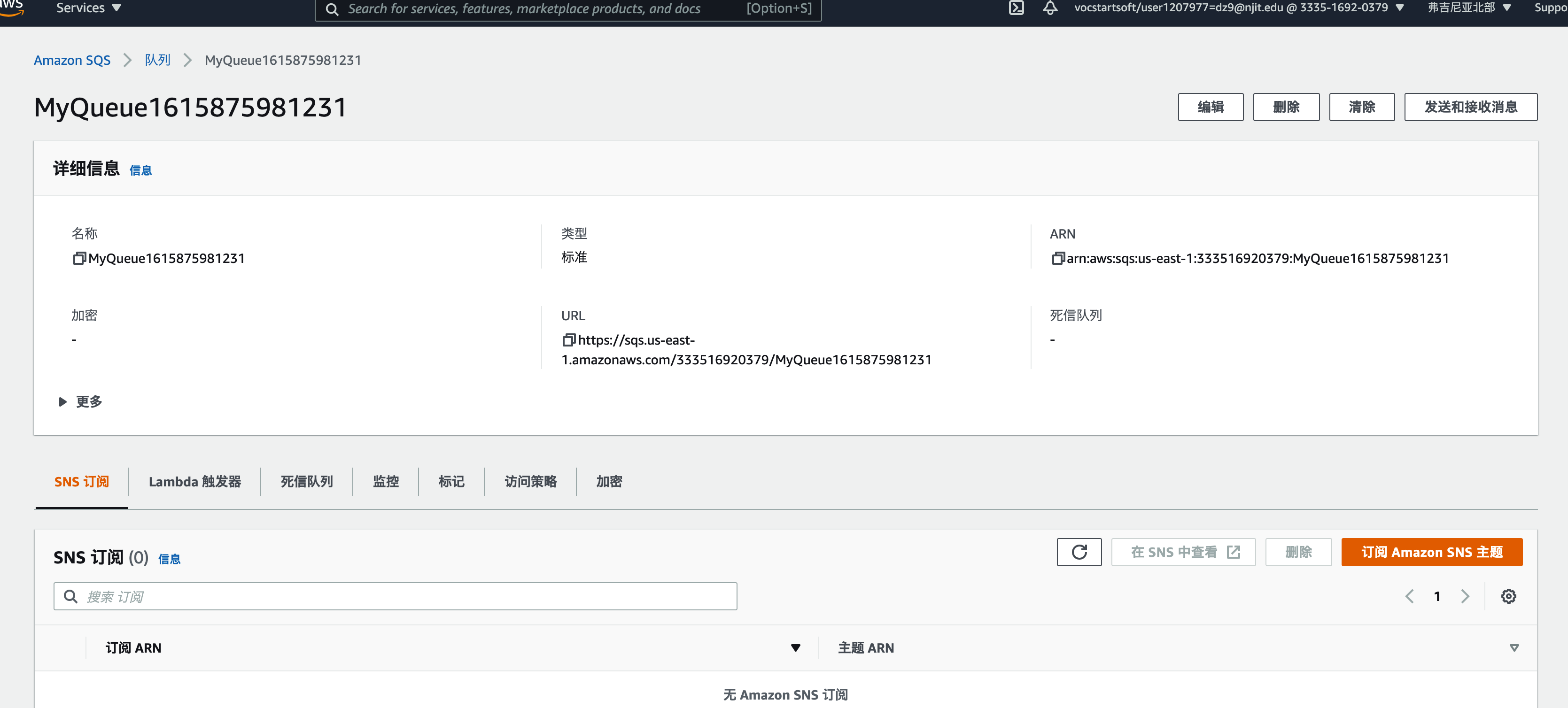
1. Make the project to JAR file



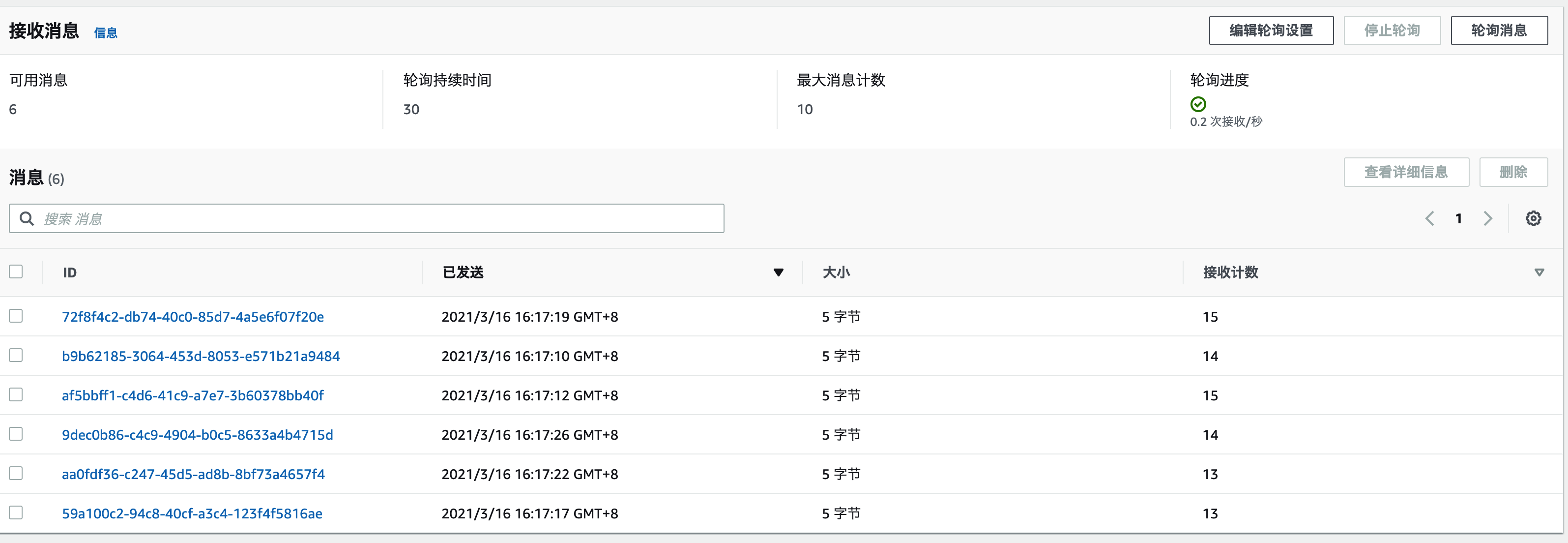
1. Use FilaZilla to upload JAR files to A instance and B instance.



1. Make sure the SQS is ok

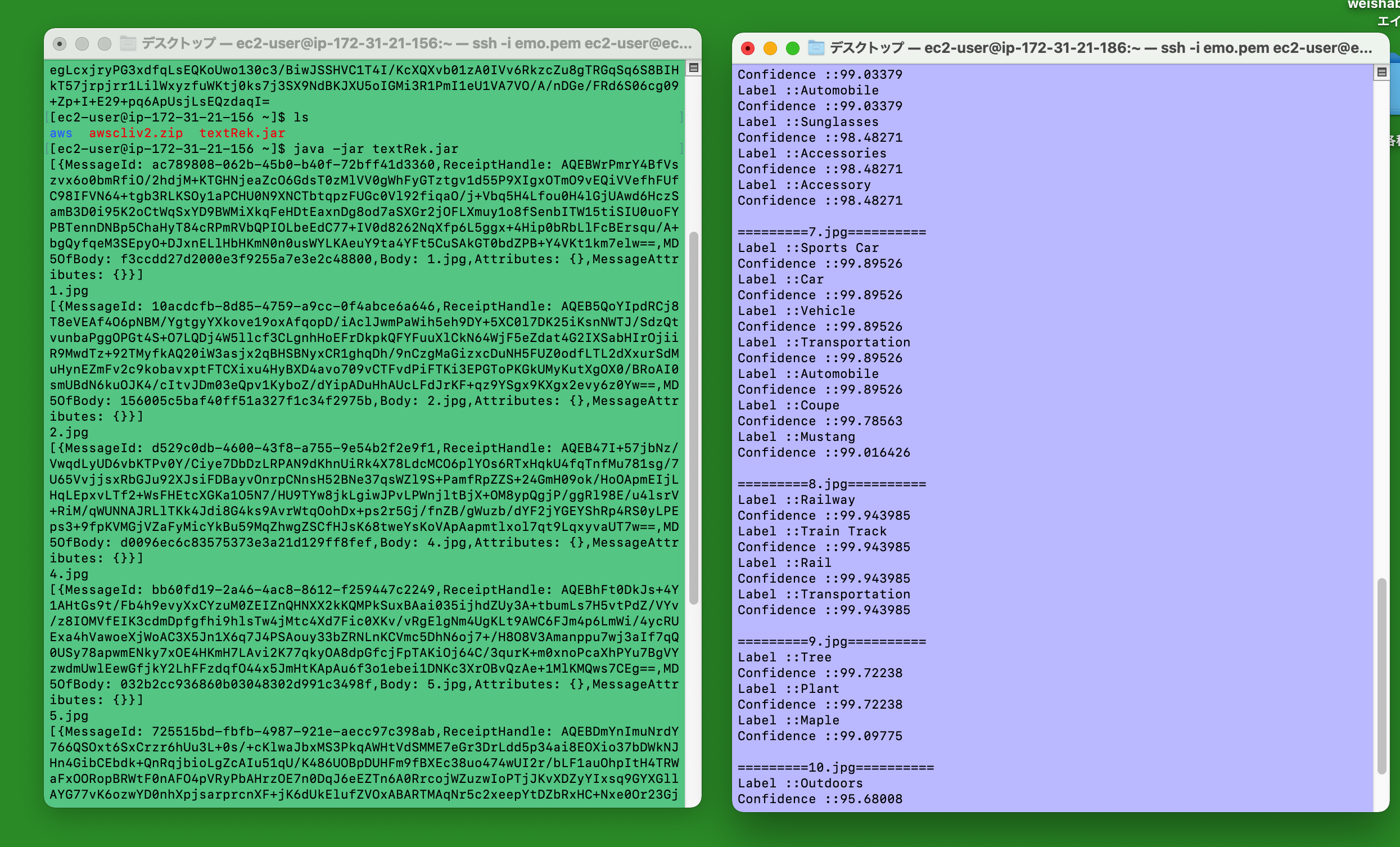


1. After A ran, the Messages in SQS (storing the images’ file name) will look like this.

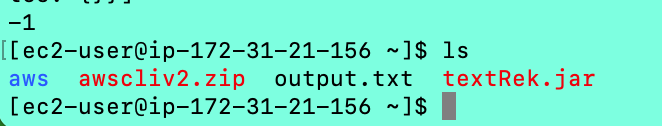




1. Run the car recognition and text recognition by instance A and B in parallel



1. When the 2 programs are finished, the output.txt file will be outputted.



1. The output.txt is looks like:

