**Amazon S3 Buckets**

Benjamin M. Brandhorst

University of Maryland Global Campus

SDEV 400 7980 Secure Programming in the Cloud

Doctor James Robertson

March 29th, 2020

**1. (20 points) Using the AWS Management Console, create two (2) S3 buckets with DNS-compliant names of your choice. In each of the buckets create 2 folders named as follows:**

**• Jobs  
• Support**

A screenshot of a cell phone

Description automatically generated

Figure 1 – Amazon Web Services (AWS) Bucket Creation Demonstration

A screenshot of a social media post

Description automatically generated

Figure 2 – AWS Bucket Folder Demonstration 1

A screenshot of a cell phone

Description automatically generated

Figure 3 – AWS Bucket Folder Demonstration 2

**2. (10 points) Using the AWS Management Console, copy 2 files (or your choice) to each of the 2 folders you created for your S3 buckets. (This means a total of 8 files will be moved to the S3 buckets.)**

A screenshot of a cell phone

Description automatically generated

Figure 4 – AWS File Upload Demonstration 1

A screenshot of a cell phone

Description automatically generated

Figure 5 – AWS File Upload Demonstration 2

A screenshot of a cell phone

Description automatically generated

Figure 6 – AWS File Upload Demonstration 3

A screenshot of a cell phone

Description automatically generated

Figure 7 – AWS File Upload Demonstration 4

**3. (50 points) Using the Cloud9 IDE within your assigned AWS Educate classroom, write and provide documentation supporting a Python command line menu-driven interface application that performs the following AWS S3 functionality:**

1. # File: Homework1.py
2. # Author: Ben Brandhorst
3. # Date: March 24th, 2020
4. # Purpose: SDEV400 Homework
6. **from** random **import** randrange
7. **import** logging
8. **import** boto3
9. **import** botocore
10. **import** time
11. **from** botocore.exceptions **import** ClientError
13. ## Command line menu format source https://extr3metech.wordpress.com/2014/09/14/simple-text-menu-in-python/
14. **def** print\_menu():
15. **print** (50 \* '-' , 'MENU' , 50 \* '-')
16. **print** ("Welcome to Ben Brandhorst's SDEV400 Homework 1 Submission. Please select from the menu below.")
17. **print** ('')
18. **print** ('1. Create a S3 bucket with the name consisting of your firstname, lastname and a random 6- digit suffix.')
19. **print** ('2. Upload File1.txt and File2.txt to benbrandhorst1 bucket.')
20. **print** ('3. Delete Uploaded1.txt from benbrandhorst1.')
21. **print** ('4. Delete an empty bucket from a list of buckets.')
22. **print** ('5. Copy Uploaded2 from benbrandhorst1 to benbrandhorst2.')
23. **print** ('6. Download Copied1.txt from benbrandhorst2.')
24. **print** ('7. Exit the Program')
25. **print** (107 \* '-')
27. **def** bucket():
28. # Creates the filename for the bucket using input for First name and Last name
29. fname = input('Please enter your first name ')
30. lname = input('Please enter your last name ')
31. filename = ('')
32. number = randrange(100000,999999,1)
33. dash = ("-")
34. filename += fname.lower()
35. filename += lname.lower()
36. filename += dash
37. filename += str(number)
39. # Create bucket used in menu selection 1
40. **try**:
41. s3\_client = boto3.client('s3')
42. s3\_client.create\_bucket(Bucket=filename)
43. **print**('Created a bucket named '+str(filename) )
44. **except** ClientError as e:
45. logging.error(e)
47. **def** upload():
48. # Uploads 2 text files to benbrandhorst1 and saves as Uploaded1.txt and Uploaded2.txt
49. s3\_client = boto3.client('s3')
50. **try**:
51. response = s3\_client.upload\_file('/home/ec2-user/environment/File1.txt','benbrandhorst1', 'Uploaded1.txt')
52. **print** ('File1.txt uploaded to a bucket named benbrandhorst1 and saved as Uploaded1.txt')
53. response = s3\_client.upload\_file('/home/ec2-user/environment/File2.txt','benbrandhorst1', 'Uploaded2.txt')
54. **print** ('File2.txt uploaded to a bucket named benbrandhorst1 and saved as Uploaded2.txt')
55. **except** ClientError as e:
56. logging.error(e)
57. **return** False
58. **return** True
60. **def** deleteFile():
61. # Delete Uploaded1.txt from benbrandhorst1
62. s3 = boto3.client('s3')
63. **try**:
64. s3.delete\_object(Bucket='benbrandhorst1', Key='Uploaded1.txt')
65. **print** ('Deleted Uploaded1.txt from bucket benbrandhorst1.')
66. **except** ClientError as e:
67. logging.error(e)
68. **return** False
69. **return** True
71. **def** deleteBucket():
72. # Lists the buckets created
73. s3 = boto3.client('s3')
74. response = s3.list\_buckets()
75. buckets = [bucket['Name'] **for** bucket **in** response['Buckets']]
76. **print**("Bucket list: %s" % buckets)
77. # Deletes the empty bucket the user selects
78. **try**:
79. bucketDelete = input("Please select the bucket to delete. ")
80. **print**('Deleted bucket ' +str(bucketDelete))
81. s3.delete\_bucket(Bucket=bucketDelete)
82. **except** ClientError as e:
83. logging.error(e)
84. **return** False
85. **return** True
87. **def** copyObject():
88. # Copies Uploaded2.txt from benbrandhorst1 bucket to benbrandhorst2 bucket and renames file Copied1.txt
89. **try**:
90. copy\_source = { 'Bucket': 'benbrandhorst1', 'Key': 'Uploaded2.txt'}
91. s3 = boto3.resource('s3')
92. bucket = s3.Bucket('benbrandhorst2')
93. bucket.copy(copy\_source, 'Copied1.txt')
94. **print**('Copied Uploaded2.txt from bucket benbrandhorst1 to bucket benbrandhorst2 and renamed the file Copied1.txt. ')
95. **except** ClientError as e:
96. logging.error(e)
97. **return** False
98. **return** True
100. **def** downloadFile():
101. # Downloads Copied1.txt from benbrandhorst2 and saves as DownloadedFile.txt
102. s3 = boto3.resource('s3')
103. **try**:
104. s3.Bucket('benbrandhorst2').download\_file('Copied1.txt', 'DownloadedFile.txt')
105. **print**('Copied1.txt has been downloaded from bucket benbrandhorst2 and saved as DownloadedFile.txt')
106. **except** botocore.exceptions.ClientError as e:
107. **if** e.response['Error']['Code'] == "404":
108. **print**("The object does not exist.")
109. **else**:
110. **raise**
112. **def** displayTime():
113. # Source for code https://www.tutorialspoint.com/python/python\_date\_time.htm
114. # Displays local time
115. localtime = time.asctime( time.localtime(time.time()) )
116. **print** ('Program shut down at:', localtime)
118. # Loop for menu that runs until "Exit Program" is selected
119. loop=True
120. **while** loop:
121. print\_menu()
122. choice = input('Enter your choice [1-7]: ')
123. **if** choice==('1'):
124. # Runs bucket() method to create a new bucket
125. bucket()
126. **elif** choice==('2'):
127. # Runs upload() method to upload objects
128. upload()
129. **elif** choice==('3'):
130. # Runs deleteFile() method to delete files
131. deleteFile()
132. **elif** choice==('4'):
133. # Runs deleteBucket() method to delete selected bucket
134. deleteBucket()
135. **elif** choice==('5'):
136. # Runs copyObject() method to copy object from one bucket to another
137. copyObject()
138. **elif** choice==('6'):
139. # Runs downloadFile() method to download file from bucket
140. downloadFile()
141. **elif** choice==('7'):
142. # Exits program
143. **print**("You have selected 'Exit Program'")
144. displayTime()
145. loop=False
146. **else**:
147. # Any integer inputs other than values 1-7 returns this error
148. input("Wrong option selection. Enter any key to try again.")

Figure 8 – Python code written for assignment

**TEST CASES**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Case** | **Input** | **Expected Output** | **Actual Output** | **Pass/Fail** |
| 1 | 1, Ben, Brandhorst | Created bucket named benbrandhorst-?????? | Created bucket named benbrandhorst-646036 | PASS |
| 2 | 1, ben, brandhorst | Created bucket named benbrandhorst-??????? | Created bucket named benbrandhorst-286659 | PASS |
| 3 | 2 | File1.txt uploaded to a bucket named benbrandhorst1 and saved as Uploaded1.txt  File2.txt uploaded to a bucket named benbrandhorst1 and saved as Uploaded2.txt | File1.txt uploaded to a bucket named benbrandhorst1 and saved as Uploaded1.txt  File2.txt uploaded to a bucket named benbrandhorst1 and saved as Uploaded2.txt | PASS |
| 4 | 3 | Deleted bucket benbrandhorst-286659 | Deleted bucket benbrandhorst-286659 | PASS |
| 5 | 4, benbrandhorst-286659 | Deleted bucket benbrandhorst-286659 | Deleted bucket benbrandhorst-286659 | PASS |
| 6 | 5 | Copied Uploaded2.txt from bucket benbrandhorst1 to bucket benbrandhorst2 and renamed the file Copied1.txt. | Copied Uploaded2.txt from bucket benbrandhorst1 to bucket benbrandhorst2 and renamed the file Copied1.txt. | PASS |
| 7 | 6 | Copied1.txt has been downloaded from bucket benbrandhorst2 and saved as DownloadedFile.txt | Copied1.txt has been downloaded from bucket benbrandhorst2 and saved as DownloadedFile.txt | PASS |
| 8 | 7 | You have selected 'Exit Program'  Program shut down at: ?????? | You have selected 'Exit Program'  Program shut down at: Wed Mar 25 19:14:08 2020 | PASS |

A screenshot of a cell phone

Description automatically generated

Figure 9 – Test Case 1 example demonstrating conversion of uppercase character input to lowercase for bucket name

Figure 10 – Test Case 1 verification that bucket was created

A screenshot of a cell phone

Description automatically generated

Figure 11 – Test Case 2 demonstration bucket creation with lowercase input



Figure 12 – Test Case 2 verification bucket was created

A screenshot of a cell phone

Description automatically generated

Figure 13 – Test Case 3 showing program output indicating files were uploaded

A screenshot of a cell phone

Description automatically generated

Figure 14 – Test Case 3 verifying text files were uploaded to the bucket

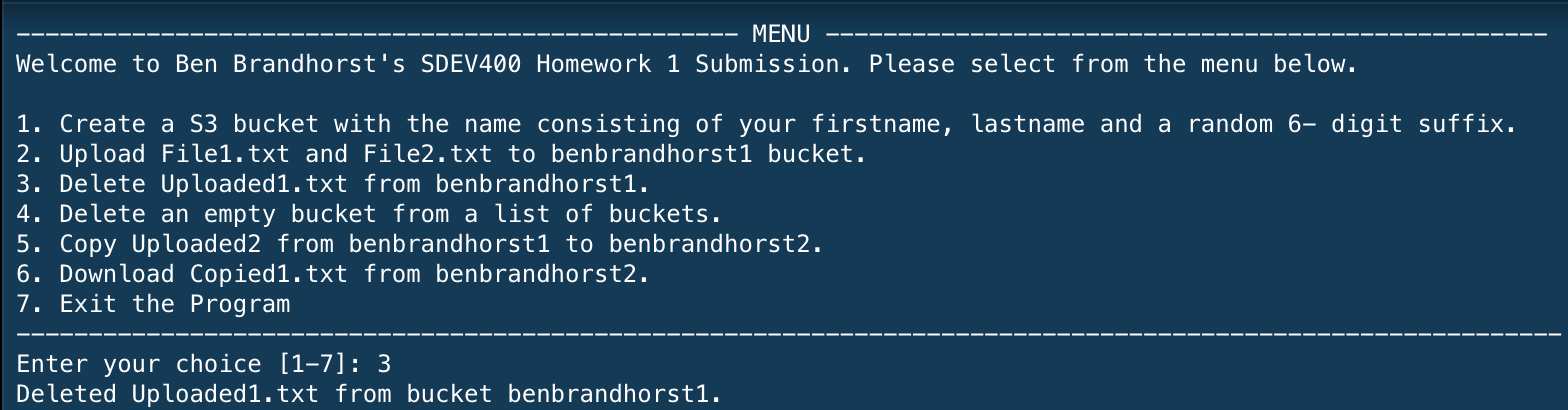


Figure 15 – Test Case 4 showing program output indicated text file deletion

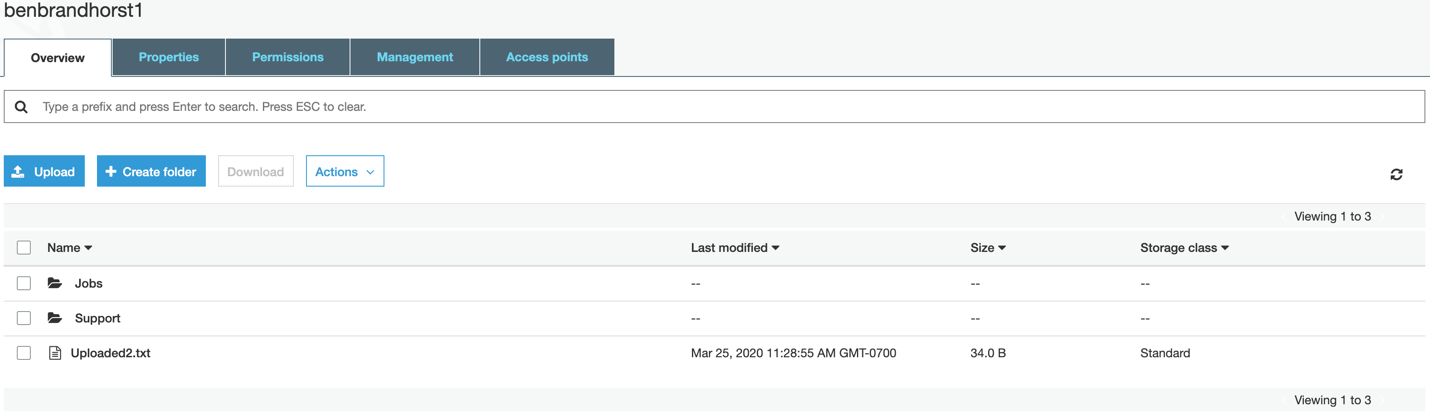


Figure 16 – Test Case 4 verifying text file deletion from bucket

A screenshot of a cell phone

Description automatically generated

Figure 17 – Test Case 5 showing program output indicating bucket deletion

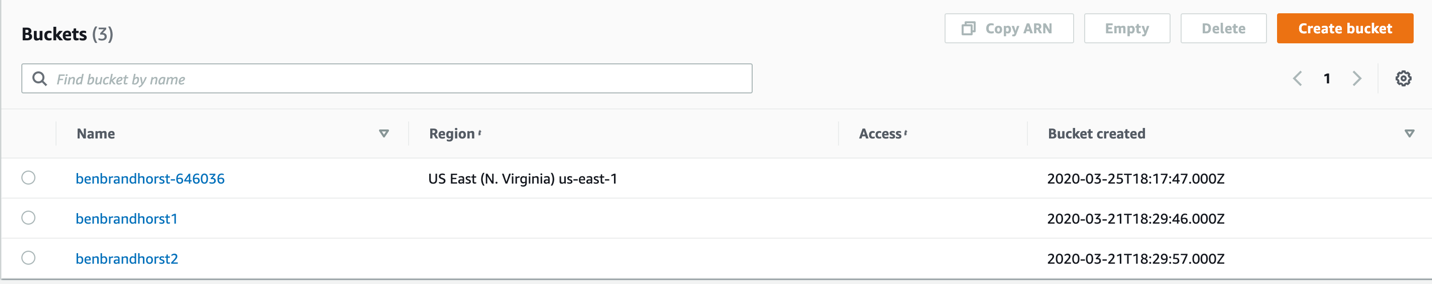


Figure 18 – Test Case 5 verifying bucket was deleted

A screenshot of a cell phone

Description automatically generated

Figure 19 – Test Case 6 showing program output indicating file was copied from one bucket to another

A screenshot of a cell phone

Description automatically generated

Figure 19 – Test Case 6 verifying file was copied and saved in new bucket

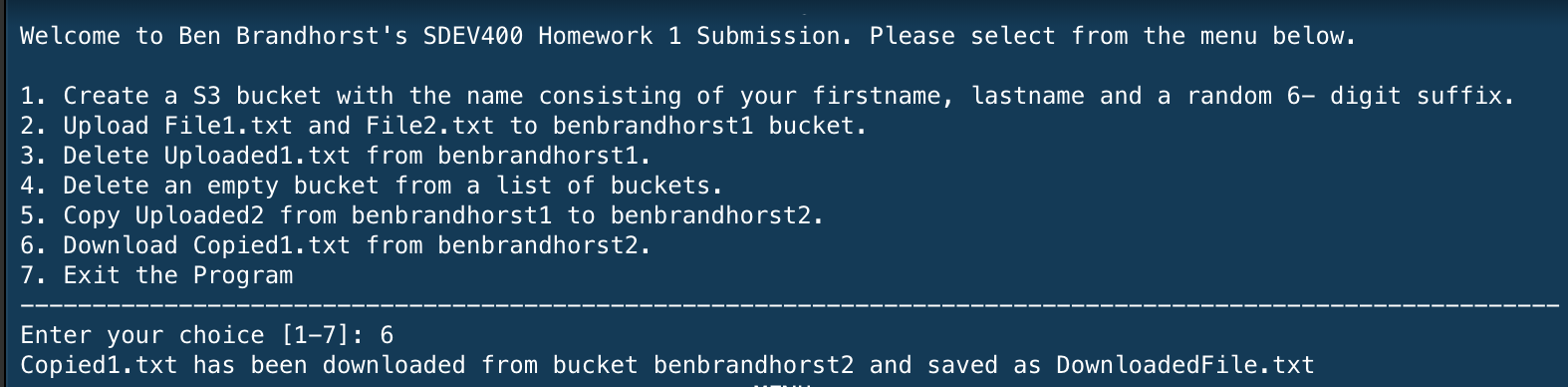


Figure 20 – Test Case 7 showing program output indication file was downloaded

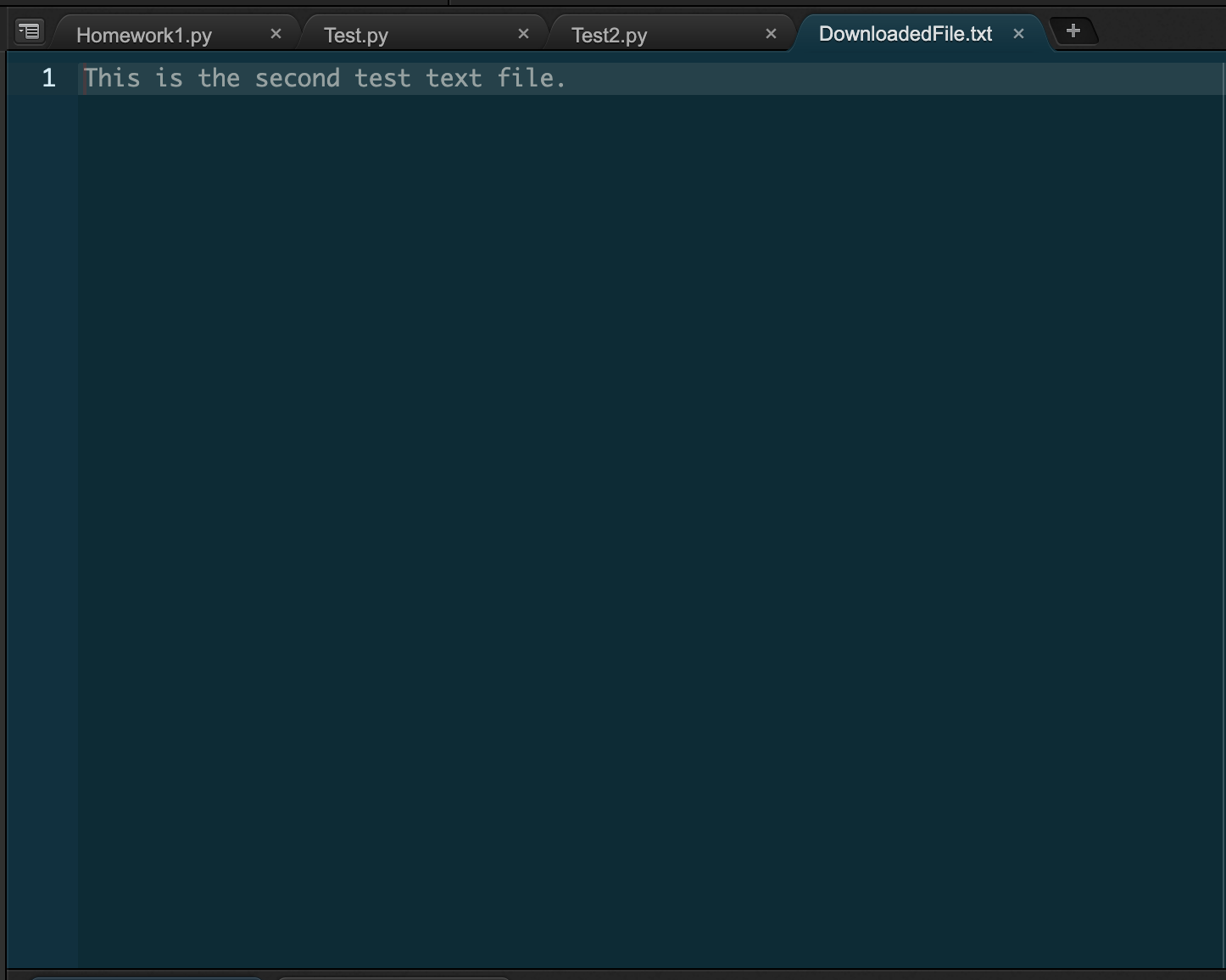


Figure 21 – Test Case 7 verifying file download in the Cloud9 IDE

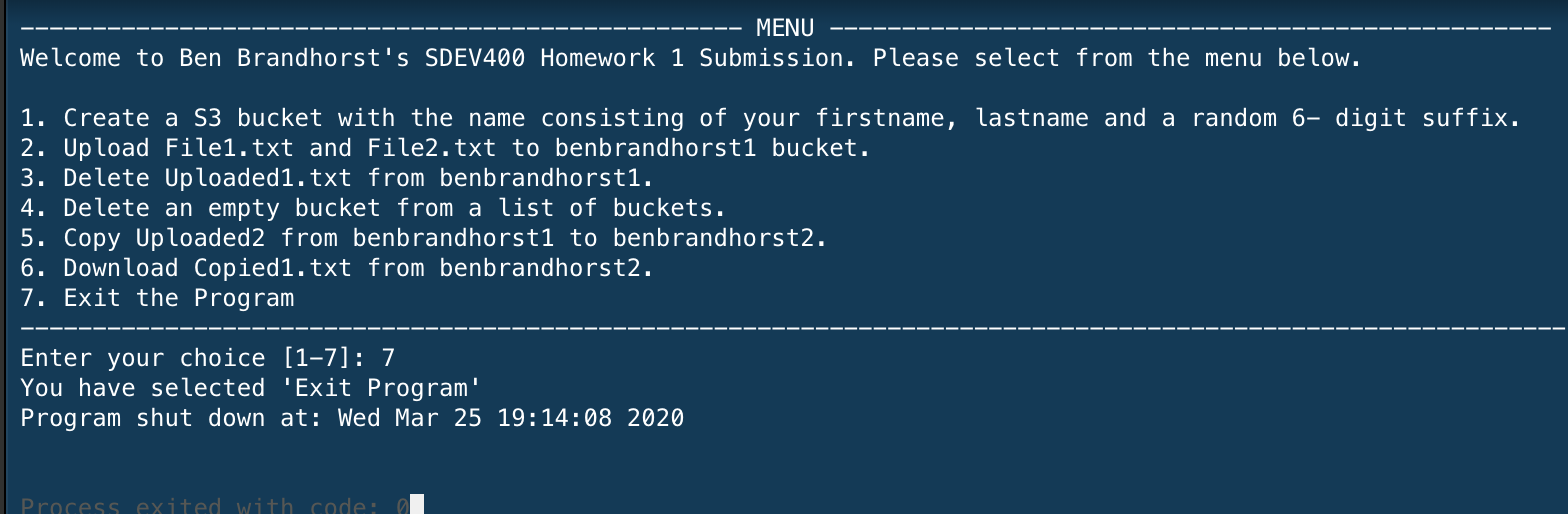


Figure 22 – Test Case 8 showing program output when “Exit Program” is selected from menu

References

Python - date & time - Tutorialspoint. (n.d.). Retrieved from https://www.tutorialspoint.com/python/python\_date\_time.htm

Simple text menu in Python. (2014, September 14). Retrieved from https://extr3metech.wordpress.com/2014/09/14/simple-text-menu-in-python/