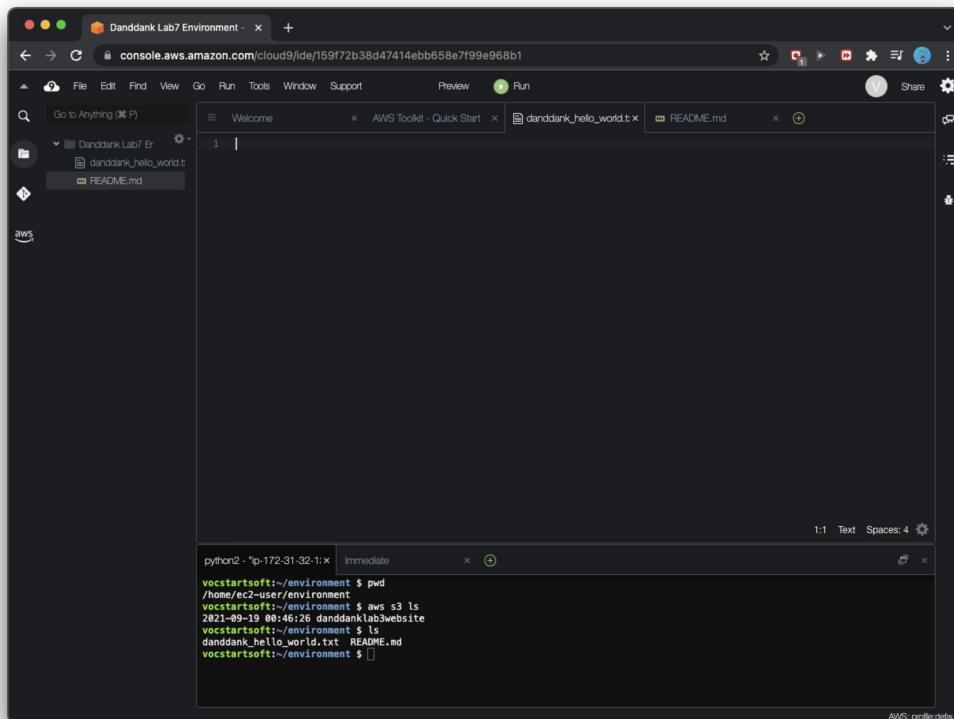


## Lab #7: Programmatic Access to AWS and GCP using CLI & SDK

Exercise 1: Programmatically working with AWS S3 using CLI & SDK.

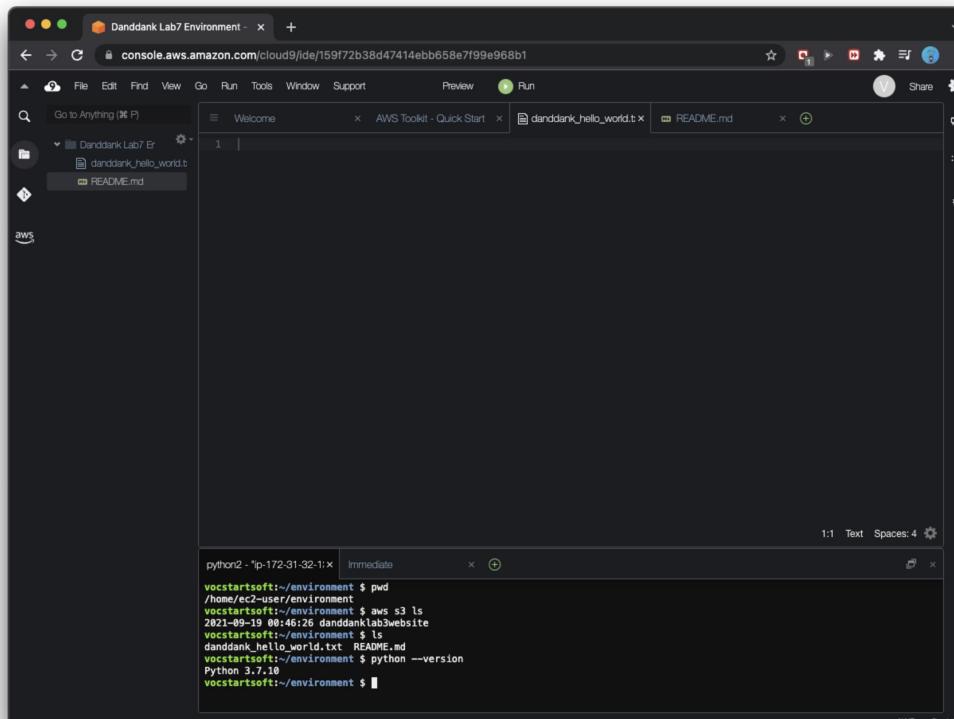
### Lab report screen-shot #1:



A screenshot of the AWS Toolkit for VS Code interface. The terminal window shows the following AWS CLI commands being run:

```
python2 - "ip-172-31-32-1:x" Immediate
vocstartsoft:/environment $ pwd
/home/ec2-user/environment
vocstartsoft:/environment $ aws s3 ls
2021-09-19 08:46:26 danddanklab3website
vocstartsoft:/environment $ ls
danddank_hello_world.txt README.md
vocstartsoft:/environment $
```

### Lab report screen-shot #2:



A screenshot of the AWS Toolkit for VS Code interface. The terminal window shows the following AWS CLI commands being run:

```
python2 - "ip-172-31-32-1:x" Immediate
vocstartsoft:/environment $ pwd
/home/ec2-user/environment
vocstartsoft:/environment $ aws s3 ls
2021-09-19 08:46:26 danddanklab3website
vocstartsoft:/environment $ ls
danddank_hello_world.txt README.md
vocstartsoft:/environment $ python --version
python 3.7.10
vocstartsoft:/environment $
```

### Lab report screen-shot #3:

A screenshot of the AWS Cloud9 IDE interface. The left sidebar shows a file tree with a directory structure: Danddark Lab7 Environment > DanddarkLab7 > danddark\_hello\_world. Inside danddark\_hello\_world, there is a file named danddark\_hello\_world.py and a README.md file. The main workspace contains a code editor with the following Python script:

```
1 import sys
2
3 my_name = sys.argv[1]
4 print("Hi There " + my_name)
```

Below the code editor is a terminal window titled "python3 - ip-172-31-32-1:x". The terminal output shows the script being run and printing "Hi There Nalongsone".

## Lab report screen-shot #4:

A screenshot of the AWS Cloud9 IDE interface, identical to screen-shot #4. The left sidebar shows the same file tree. The main workspace contains the same Python script danddark\_hello\_world.py. The terminal window below shows the script running and printing "Hi There Nalongsone".

## Lab report screen-shot #5:

The screenshot shows the AWS Lambda console interface. On the left, there's a file tree for a project named 'Danddank Lab7'. The 'functions' folder contains two files: 'create\_s3\_bucket.py' and 'danddank\_hello\_world.py'. The right side has two tabs: 'Preview' and 'Run'. The 'Run' tab is active, showing the Python code for creating an S3 bucket and setting up a website. Below the code is a terminal window displaying the output of a pip command to install boto3, followed by its version information.

```

import boto3
s3 = boto3.client('s3', region_name="us-east-1")
bucket_name = "danddanklab7website"
s3.create_bucket(Bucket=bucket_name)
website_configuration = {
    "ErrorDocument": {"Key": "error.html"},
    "IndexDocument": {"Suffix": "index.html"}
}
s3.put_bucket_website(
    Bucket = bucket_name,
    WebsiteConfiguration = website_configuration
)

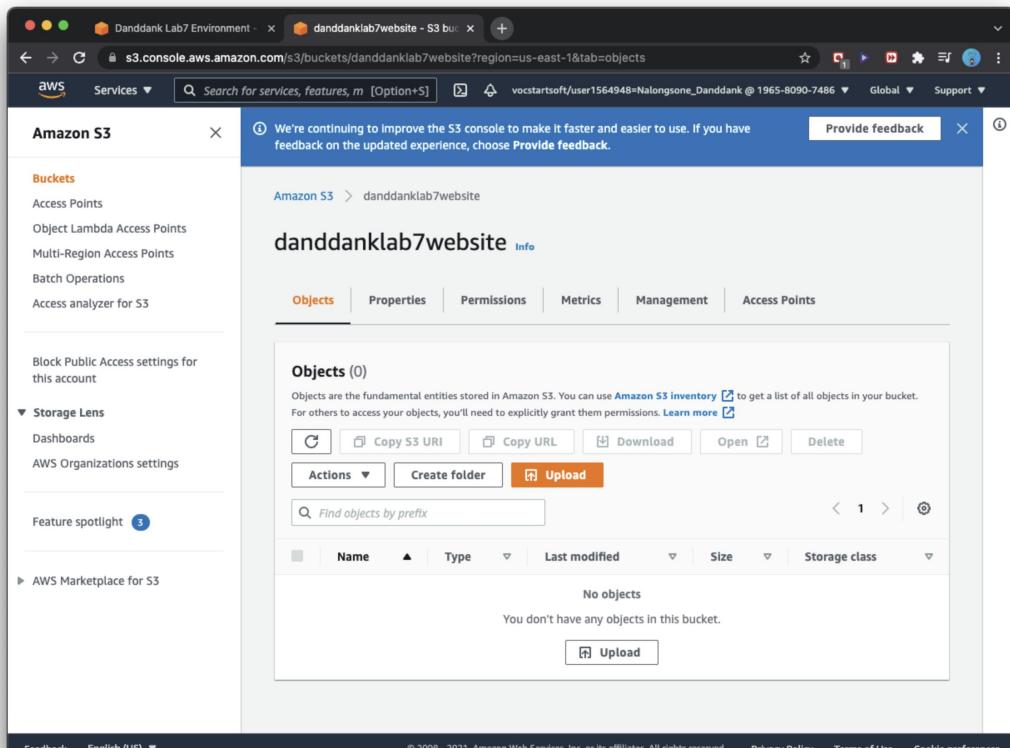
```

```

python3 -i ip-172-31-32-1.x
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.7/site-packages (from python-dateutil<3.0.0,>=2.1->botocore<1.23.0,>=1.22.1->boto3) (1.16.0)
Installing collected packages: botocore, s3transfer, boto3
  Attempting uninstall: botocore
    Found existing installation: botocore 1.21.61
      Uninstalling botocore-1.21.61...
      Successfully uninstalled botocore-1.21.61
Successfully installed boto3-1.19.1 botocore-1.22.1 s3transfer-0.5.0
vocstartsoft:~/environment/DanddankLab7 $ python -m pip show boto3
Name: boto3
Version: 1.19.1
Summary: The AWS SDK for Python
Home-page: https://github.com/boto/boto3
Author: Amazon Web Services
Author-email: None
License: Apache License 2.0
Location: /usr/local/lib/python3.7/site-packages
Requires: s3transfer, botocore, jmespath
Required-by:

```

## Lab report screen-shot #6:



## Lab report screen-shot #7:

```

import boto3
s3 = boto3.client('s3', region_name="us-east-1")
bucket_name = "danddanklab7website"
file_name_1 = '/home/ec2-user/environment/DanddankLab7/lab-07-files/index.html'
response = s3.upload_file(file_name_1, bucket_name, "index.html", ExtraArgs={"ContentType": "text/html"})
file_name_2 = '/home/ec2-user/environment/DanddankLab7/lab-07-files/index.jpg'
response = s3.upload_file(file_name_2, bucket_name, "index.jpg", ExtraArgs={"ContentType": "image/jpg"})
file_name_3 = '/home/ec2-user/environment/DanddankLab7/lab-07-files/logo.jpg'
response = s3.upload_file(file_name_3, bucket_name, "logo.jpg", ExtraArgs={"ContentType": "image/jpg"})

```

Process exited with code: 0

## Lab report screen-shot #8:

Name	Type	Last modified	Size	Storage class
index.html	html	October 22, 2021, 00:31:28 (UTC-05:00)	909.0 B	Standard
index.jpg	jpg	October 22, 2021, 00:31:28 (UTC-05:00)	67.3 KB	Standard
logo.jpg	jpg	October 22, 2021, 00:31:28 (UTC-05:00)	9.8 KB	Standard

## Lab report screen-shot #9:

The screenshot shows the AWS Cloud9 IDE interface. On the left, a file tree displays a directory structure for 'Danddank Lab7' containing files like 'create\_s3\_bucket.py', 'list\_bucket\_objects.py', and 'upload\_file.py'. The main code editor window contains the following Python script:

```
import boto3
s3 = boto3.resource("s3")
bucket_name = "danddanklab7website"
my_bucket = s3.Bucket(bucket_name)
my_files = my_bucket.objects.all()
for file in my_files:
    print(file)
```

Below the code editor is a terminal window titled 'bash - ip-172-31-32-138.x' showing the output of pip package management and an AWS S3 command:

```
Installing collected packages: botocore, s3transfer, boto3
  Attempting uninstall: botocore
    Found existing installation: botocore 1.21.61
      Uninstalling botocore-1.21.61:
        Successfully uninstalled botocore-1.21.61
Successfully installed boto3-1.19.1 botocore-1.22.1 s3transfer-0.5.0
vocstartsoft:/environment/DanddankLab7$ python -m pip show boto3
Name: boto3
Version: 1.19.1
Summary: The AWS SDK for Python
Home-page: https://github.com/boto/boto3
Author: Amazon Web Services
Author-email: None
License: Apache License 2.0
Location: /usr/local/lib/python3.7/site-packages
Requires: s3transfer, botocore, jmespath
Required-by:
vocstartsoft:/environment/DanddankLab7$ aws s3 ls
2021-09-17 00:46:26 danddanklab7website
2021-09-17 05:18:01 danddanklab7website
vocstartsoft:/environment/DanddankLab7$
```

## Lab report screen-shot #10:

This screenshot is similar to the previous one but shows the execution of the 'list\_bucket\_objects.py' script. The terminal window now displays the output of the script, which lists three objects in the S3 bucket:

```
s3.ObjectSummary(bucket_name='danddanklab7website', key='index.html')
s3.ObjectSummary(bucket_name='danddanklab7website', key='index.jpg')
s3.ObjectSummary(bucket_name='danddanklab7website', key='logo.jpg')

Process exited with code: 0
```

## Lab report screen-shot #11:

The screenshot shows a browser window with three tabs open:

- Danddank Lab7 Environment
- danddanklab7website - S3 buck
- danddanklab7website - S3 buck

The main content area displays a Python script named `list_bucket_objects.py`:

```

1 import boto3
2 import json
3
4 s3 = boto3.client("s3", region_name="us-east-1")
5 bucket_name = "danddanklab7website"
6
7 policy_file = open("/home/ec2-user/enumerate/DanddankLab7/public_policy.json", "r")
8
9 s3.put_bucket_policy(
10     Bucket = bucket_name,
11     Policy = policy_file.read()
12 )

```

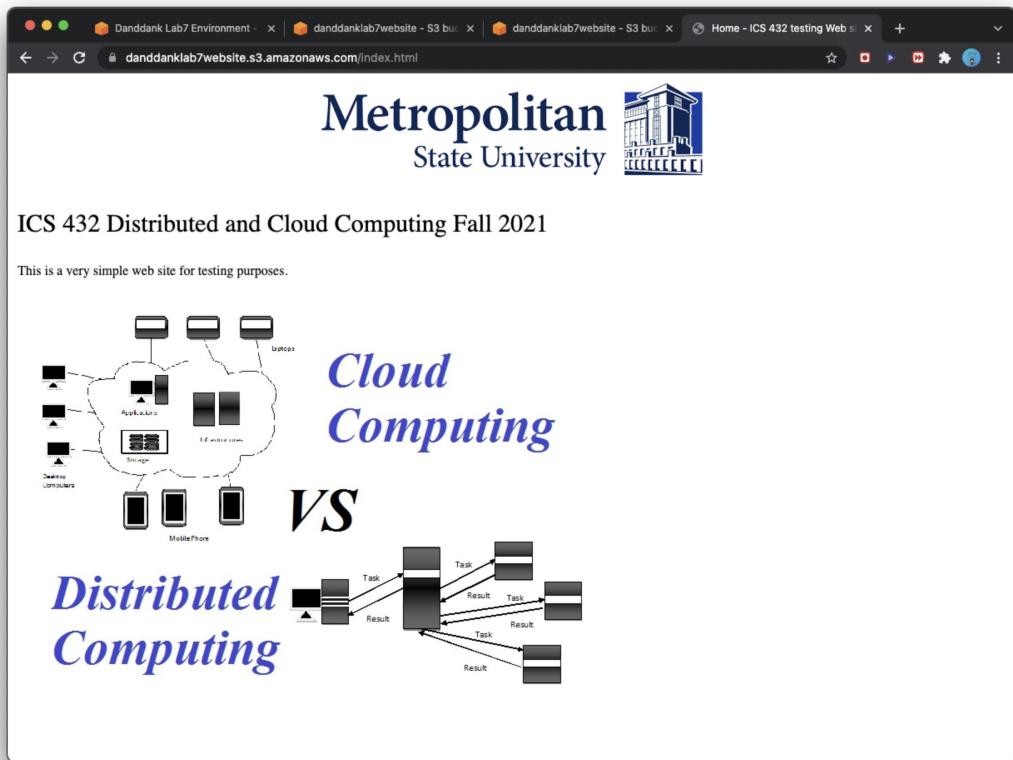
Below the script, the terminal output shows the results of running the script:

```

bash - ip-172-31-32-138 ~ | Immediate (Javascript) (brow DanddankLab7/list_bucket.py +)
[{"Key": "index.jpg", "Owner": {"DisplayName": "awsalbscw2490713t1624921646", "ID": "d3a4af408e9623d7a4b78fef0a27de47933ea18ccc476db68a7846f6fb743976"}, "Size": 6889}, {"Key": "logo.jpg", "Owner": {"DisplayName": "awsalbscw2490713t1624921646", "ID": "d3a4af408e9623d7a4b78fef0a27de47933ea18ccc476db68a7846f6fb743976"}, "Size": 10049}]
vocstartsoft:~/environment/DanddankLab7 $ 

```

## Lab report screen-shot #12:



## Lab report screen-shot #13:

```

import boto3
import json
s3 = boto3.client("s3", region_name="us-east-1")
bucket_name = "danddanklab7website"
policy_file = open("/home/ec2-user/environment/DanddankLab7/public_policy.json", "r")
s3.put_bucket_policy(
    Bucket = bucket_name,
    Policy = policy_file.read()
)

```

The screenshot shows a browser window with several tabs open, all related to AWS services. The main content area displays a Python script for interacting with AWS S3. The script imports boto3 and json, creates an S3 client for the 'us-east-1' region, defines a bucket name ('danddanklab7website'), opens a local JSON file ('public\_policy.json') containing a bucket policy, and then uses the S3 client's put\_bucket\_policy method to apply the policy to the specified bucket.

## Lab report screen-shot #14:

Instance summary for i-0317aa4ca415791c3 (aws-cloud9-Danddank-Lab7-Environment-159f72b38d47414ebb658e7f99e968b1)			
Instance ID	Public IPv4 address	Private IPv4 addresses	
i-0317aa4ca415791c3 (aws-cloud9-Danddank-Lab7-Environment-159f72b38d47414ebb658e7f99e968b1)	3.81.105.146   open address	172.31.32.138	
IPv6 address	Instance state	Public IPv4 DNS	
-	Running	ec2-3-81-105-146.compute-1.amazonaws.com   open address	
Private IPv4 DNS	Instance type	Elastic IP addresses	
ip-172-31-32-138.ec2.internal	t2.micro	-	
VPC ID	AWS Compute Optimizer finding	IAM Role	
vpc-64fa8419	<small>retry</small>	-	
Subnet ID			
subnet-310b256e			

This screenshot shows the AWS EC2 Instances details page for a specific instance. The instance ID is i-0317aa4ca415791c3, which is associated with the environment 'aws-cloud9-Danddank-Lab7-Environment-159f72b38d47414ebb658e7f99e968b1'. The instance is currently running and has a private IP address of 172.31.32.138. It is part of a VPC with a subnet ID of subnet-310b256e. The AWS Compute Optimizer is finding a user role for the instance, but it is denied access to the 'compute-optimizer:GetEnrollmentStatus' action. The instance is of type t2.micro.

## Lab report screen-shot #15:

Danddank Lab7 Environment - Instance details | EC2 Manager | Home - ICS 432 testing Web | console.aws.amazon.com/cloud9/ide/159f72b38d47414ebb658e7f99e968b1

```

File Edit Find View Go Run Tools Window Support Preview Run
Go to Anything (⌘ P)
Danddank Lab7 Br
  DanddankLab7
    lab-07-files
      create_s3_bucket.py
      danddank_hello_world
      list_bucket_objects.py
      permissions.py
      public_policy.json
      upload_file.py
      working_with_ec2.py
  README.md
aws
danddank_hello_world.t

danddank_he x create_s3_bu x upload_file.py x list_bucket_o x public_policy.x permissions.i x working_with_x +
5
6 import boto3
7
8 ec2_client = boto3.client("ec2", region_name="us-east-1")
9 response = ec2_client.describe_instances()
10
11 for reservation in response["Reservations"]:
12     for instance in reservation["Instances"]:
13         print(instance["InstanceId"])
14         print(instance["LaunchTime"])
15         print(instance["ImageId"])
16         print("-----")
15:35 Python Spaces: 4
python2 -ip-172-31-32-1: x Immediate (Javascript) brow x DanddankLab7/danddank_ x +
Run Command: DanddankLab7/danddank_hello_world.py Runner: Python 3 CWD ENV
i-05a3851178bf90a0
2021-09-19 07:37:16+00:00
ami-087c17d1fe0178315
=====
i-022df3823332594e
2021-09-26 02:22:45+00:00
ami-087c17d1fe0178315
=====
i-0317aa4ca415791c3
2021-10-22 04:48:59+00:00
ami-0c18c7e4018114f52
=====
i-0a4aff20a030b1214a
2021-10-20 03:52:06+00:00
ami-02e136e904f3da870
=====
i-04b45fe4b62fdf45
2021-10-20 06:08:14+00:00
ami-02e136e904f3da870
=====

Process exited with code: 0
AWS: profile.default

```

## Lab report screen-shot #16:

Danddank Lab7 Br S Management | Home - ICS 432 | python string split | Python String split | Python Tryit Editor | console.aws.amazon.com/cloud9/ide/159f72b38d47414ebb658e7f99e968b1

```

File Edit Find View Go Run Tools Window Support Preview Run
Go to Anything (⌘ P)
Danddank Lab7 Br
  DanddankLab7
    lab-07-files
      reading
        create_s3_bucket.py
        danddank_hello_world
        list_bucket_objects.py
        my_code_lab7_exercise
        upload_file.py
        working_with_ec2.py
  README.md
aws
danddank_hello_world.t

my_code_lab7_exercise x create_s3_bucket.py x list_bucket_objects.py x upload_file.py x +
1
2 import sys
3 import boto3
4 import os
5
6 bucket_name = sys.argv[1]
7 print("creating you new bucket with name: " + bucket_name)
8
9 s3 = boto3.client("s3", region_name="us-east-1")
10 s3.create_bucket(Bucket=bucket_name)
11 file_path = "/home/ec2-user/environment/DanddankLab7/reading/{0}"
12 arr = os.listdir("/home/ec2-user/environment/DanddankLab7/reading")
13 print("Upload files from AWS Cloud9 folder to Bucket S3: ")
14 for f in arr:
15     file_name = file_path.format(f)
16     type_ = f.split('.')[1]
17     if(type_ == 'jpg'):
18         response = s3.upload_file(file_name, bucket_name, f, ExtraArgs={"ContentType": "image/jpeg"})
19     elif(type_ == 'txt' or type_ == 'html'):
20         response = s3.upload_file(file_name, bucket_name, f, ExtraArgs={"ContentType": "text/html"})
21     print(file_name)
22
23 s3_res = boto3.resource("s3")
24 my_bucket = s3_res.Bucket(bucket_name)
25 my_files = my_bucket.objects.all()
26 print("Read files from Bucket S3: " + bucket_name)
27 for file in my_files:
28     print(file)
29 print("Upload all files successfully.")
24:35 Python Spaces: 4
bash - ip-172-31-32-138: x Immediate (Javascript) brow x +
vocstartsoft:~/environment/DanddankLab7 $ clear
vocstartsoft:~/environment/DanddankLab7 $ python3 my_code_lab7_exercise.py danddankbucketlab7exercise
creating you new bucket with name: danddankbucketlab7exercise
Upload files from AWS Cloud9 folder to Bucket S3:
/home/ec2-user/environment/DanddankLab7/reading/panda.jpg
/home/ec2-user/environment/DanddankLab7/reading/test.txt
/home/ec2-user/environment/DanddankLab7/reading/learning.txt
/home/ec2-user/environment/DanddankLab7/reading/404.html
Read files from Bucket S3:danddankbucketlab7exercise
s3.ObjectSummary(bucket_name='danddankbucketlab7exercise', key='404.html')
s3.ObjectSummary(bucket_name='danddankbucketlab7exercise', key='learning.txt')
s3.ObjectSummary(bucket_name='danddankbucketlab7exercise', key='panda.jpg')
s3.ObjectSummary(bucket_name='danddankbucketlab7exercise', key='test.txt')
Upload all files successfully.
vocstartsoft:~/environment/DanddankLab7 $ 
AWS: profile.default

```

## Lab report screen-shot #17:

```

import sys
import boto3
import os

bucket_name = sys.argv[1]
print("Creating new bucket with name: " + bucket_name)

s3 = boto3.client("s3", region_name="us-east-1")
s3.create_bucket(Bucket=bucket_name)
file_path = "/home/ec2-user/environment/DanddankLab7/reading/{0}"
arr = os.listdir("/home/ec2-user/environment/DanddankLab7/reading")
print("Upload files from AWS Cloud9 folder to Bucket S3: ")
for f in arr:
    file_name = file_path.format(f)
    type_ = f.split('.')[1]
    if(type_ == 'jpg'):
        response = s3.upload_file(file_name, bucket_name, f, ExtraArgs={"ContentType": "image/jpeg"})
    elif(type_ == 'txt' or type_ == 'html'):
        response = s3.upload_file(file_name, bucket_name, f, ExtraArgs={"ContentType": "text/html"})
    print(file_name)

s3_res = boto3.resource("s3")
my_bucket = s3_res.Bucket(bucket_name)
my_files = my_bucket.objects.all()
print("Read files from Bucket S3: " + bucket_name)
for file in my_files:
    print(file)
print("Upload all files successfully.")

```

bash - ip-172-31-32-138.x Immediat Javascript (brow +)

```

vocestartssoft:/environment/DanddankLab7 $ clear
vocestartssoft:/environment/DanddankLab7 $ python3 my_code_lab7_exercise.py danddankbucketlab7exercise
Creating new bucket with name: danddankbucketlab7exercise
Upload files from AWS Cloud9 folder to Bucket S3:
/home/ec2-user/environment/DanddankLab7/reading/panda.jpg
/home/ec2-user/environment/DanddankLab7/reading/test.txt
/home/ec2-user/environment/DanddankLab7/reading/learning.txt
/home/ec2-user/environment/DanddankLab7/reading/404.html
Read files from Bucket S3:danddankbucketlab7exercise
s3.ObjectSummary(bucket_name='danddankbucketlab7exercise', key='404.html')
s3.ObjectSummary(bucket_name='danddankbucketlab7exercise', key='learning.txt')
s3.ObjectSummary(bucket_name='danddankbucketlab7exercise', key='panda.jpg')
s3.ObjectSummary(bucket_name='danddankbucketlab7exercise', key='test.txt')
Upload all files successfully.
vocestartssoft:/environment/DanddankLab7 $ 

```

## Lab report screen-shot #18:

	Name	Type	Last modified	Size	Storage class
<input type="checkbox"/>	404.html	html	October 22, 2021, 02:13:57 (UTC-05:00)	309.0 B	Standard
<input type="checkbox"/>	learning.txt	txt	October 22, 2021, 02:13:57 (UTC-05:00)	0 B	Standard
<input type="checkbox"/>	panda.jpg	jpg	October 22, 2021, 02:13:56 (UTC-05:00)	290.1 KB	Standard
<input type="checkbox"/>	test.txt	txt	October 22, 2021, 02:13:57 (UTC-05:00)	56.0 B	Standard

## Exercise 2: Programmatic access of GCP Storage.

### Lab report screen-shot #19:

The screenshot shows the Google Cloud Platform Cloud Shell interface. On the left, there's a file explorer window titled 'EXPLORER: NALONGSON...' showing a directory 'DanddankLab7' containing files 'bucketaccess.py' and 'README-cloudshell.txt'. The main workspace contains a code editor with the following Python script:

```
1 import os
2 import sys
3 from google.cloud import storage
4
5 bucket_name = 'danddank-ics432-website-files'
6 storage_client = storage.Client()
7
8 bucket = storage_client.create_bucket(bucket_name, location="us")
9 bucket.storage_class = "COLDLINE"
10 buckets = storage_client.list_buckets()
11
12 for bucket in buckets:
13     print(bucket.name)
```

Below the code editor is a terminal window showing the command being run: `/usr/bin/python3 /home/nalongsone_danddank/DanddankLab7/bucketaccess.py`. The output shows the creation of a bucket named 'danddank-ics432-website-files'. The bottom status bar indicates the code is written in Python.

## Lab report screen-shot #20:

The screenshot shows the Google Cloud Platform Cloud Storage browser interface. On the left, there's a sidebar with 'Cloud Storage' selected, followed by 'Browser', 'Monitoring', and 'Settings'. The main area displays a table of buckets:

Name	Created	Location type	Location	Default storage class
danddank-ics432-website-files	Oct 22, 2021, 2:47:03 AM	Multi-region	us (multiple re...)	Standard

## Lab report screen-shot #21:

Name	Size	Type	Created	Storage class	Last modified	Public access
index.html	909 B	text/html	Oct 22, 2023	Standard	Oct 22, 2023	Not public
index.jpg	67.3 KB	image/jpeg	Oct 22, 2023	Standard	Oct 22, 2023	Not public
logo.jpg	9.8 KB	image/jpeg	Oct 22, 2023	Standard	Oct 22, 2023	Not public

## Lab report screen-shot #22:

```

EXPLORER: NALONGSON...  C  ...
└── DanddankLab7
    ├── bucketaccess.py
    ├── README-cloudshell.txt
    ├── uploadfiles.py
    ├── index.html
    ├── index.jpg
    └── logo.jpg

uploadfiles.py
12 # blob1.upload_from_filename(file_name_1)
13 # print("File {} uploaded to {}".format(file_name_1, bucket_name))
14
15 # blob2 = bucket.blob(file_name_2)
16 # blob2.upload_from_filename(file_name_2)
17 # print("File {} uploaded to {}".format(file_name_2, bucket_name))
18
19 # blob3 = bucket.blob(file_name_3)
20 # blob3.upload_from_filename(file_name_3)
21 # print("File {} uploaded to {}".format(file_name_3, bucket_name))
22
23
24 # List all objects that satisfy the filter.
25 blobs=bucket.list_blobs()
26 for blob in blobs:
27     print(blob)

Problems  ⟳ Python  ⟳ Python X
/home/nalongsone_danddank
nalongsone_danddank@cloudshell:~ (danddank-lab7)$ ls
DanddankLab7  README-cloudshell.txt
Lab7/uploadfiles.py
<Blob: danddank-ics432-website-files, /home/nalongsone_danddank/DanddankLab7/index.html, 1634889405526472>
<Blob: danddank-ics432-website-files, /home/nalongsone_danddank/DanddankLab7/index.jpg, 1634889405722669>
<Blob: danddank-ics432-website-files, /home/nalongsone_danddank/DanddankLab7/logo.jpg, 1634889405867188>
nalongsone_danddank@cloudshell:~(danddank-lab7)$ 

```

## Lab report screen-shot #23:

Google Cloud Platform - danddank-lab7 - Google Cloud Storage - Bucket - google cloud storage list files - How to download files from Google Cloud Storage

Cloud Shell Editor

```
File Edit Selection View Go Run Terminal Help
EXPLORER: NALONGSONE... C uploadfiles.py
15 # blob2 = bucket.blob(file_name_2)
16 # blob2.upload_from_filename(file_name_2)
17 # print("File {} uploaded to {}".format(file_name_2, bucket_name))
18
19 # blob3 = bucket.blob(file_name_3)
20 # blob3.upload_from_filename(file_name_3)
21 # print("File {} uploaded to {}".format(file_name_3, bucket_name))
22
23 # List all objects that satisfy the filter.
24 blobs=bucket.list_blobs()
25 for blob in blobs:
26     print(blob.name)
27
28 source_blob_name = "streamContent.png"
29 destination_file_name = "/home/nalongsone_danddank/DanddankLab7/" + source_blob_name
30
31 blob = bucket.blob(source_blob_name)
32 blob.download_to_filename(destination_file_name)

Problems > Python > Python
nalongsonedanddank@cloudshell:~/DanddankLab7 (danddank-lab7)$ /usr/bin/python3 /home/nalongsone_danddank/DanddankLab7/uploadfiles.py
/home/nalongsonedanddank/DanddankLab7/index.html
/home/nalongsonedanddank/DanddankLab7/index.jpg
/home/nalongsonedanddank/DanddankLab7/logo.jpg
streamContent.png
nalongsonedanddank@cloudshell:~/DanddankLab7 (danddank-lab7)$ pwd
/home/nalongsonedanddank/DanddankLab7
nalongsonedanddank@cloudshell:~/DanddankLab7 (danddank-lab7)$ /usr/bin/python3 /home/nalongsone_danddank/DanddankLab7/uploadfiles.py
nalongsonedanddank@cloudshell:~/DanddankLab7 (danddank-lab7)$
Ln 33, Col 49 LF UTF-8 Spaces: 4 Python
```

Home - danddank-lab7 - Google Cloud Storage - Bucket - google cloud storage list files - How to download files from Google Cloud Storage

Cloud Storage Bucket details

**danddank-ics432-website-files**

Location	Storage class	Public access	Protection
us (multiple regions in United States)	Standard	⚠ Subject to object ACLs	None

OBJECTS CONFIGURATION PERMISSIONS PROTECTION LIFECYCLE

Buckets > danddank-ics432-website-files

UPLOAD FILES UPLOAD FOLDER CREATE FOLDER MANAGE HOLDS DOWNLOAD DELETE

Filter by name prefix only ▾ Filter objects and folders Show deleted data

Name	Size	Type	Created	Storage class	Last modified	Public access	Version history
/	—	Folder	—	—	—	—	—
streamContent.png	54.2 KB	image/png	Oct 22, 2023	Standard	Oct 22, 2023	Not public	—

Release Notes

Uploads and danddank-lab7 operations

streamContent.png Complete