

STEP 1 – Configure AWS Credentials, using the “aws configure” command and then proceeding to supply the “Access Key ID and Secret Access Key ID” as well as the region Region name of the AWS and default format.

```
Command Prompt
Microsoft Windows [Version 10.0.19045.3448]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ECL-Francis>aws configure
AWS Access Key ID [*****ZNSU]: AKIA2VED3S1NL056ZNSU
AWS Secret Access Key [*****HakK]: EKgBqfojvHqruq6y3g5eR0i04JV90d2ItaA2HakK
Default region name [us-west-2]: us-west-2
Default output format [json]: json

C:\Users\ECL-Francis>
```

Create Key Pair

```
Command Prompt

transfer                | translate
verifiedpermissions    | voice-id
vpc-lattice             | waf
waf-regional            | wafv2
wellarchitected         | wisdom
workdocs               | worklink
workmail               | workmailmessageflow
workspaces              | workspaces-web
xray                   | xray
ec2                    | s3api
configure              | ddb
configservice          | deploy
history                | opsworke-cm
help                   | cli-dev

C:\Users\ECL-Francis>aws ec2 create-key-pair --key-name kenkeys --query 'KeyMaterial' --output text > kenkeys.pem
could not connect to the endpoint URL: "https://ec2.us-west-2.amazonaws.com/"

C:\Users\ECL-Francis>aws ec2 create-key-pair --key-name kenkeys --query 'KeyMaterial' --output text > kenkeys.pem

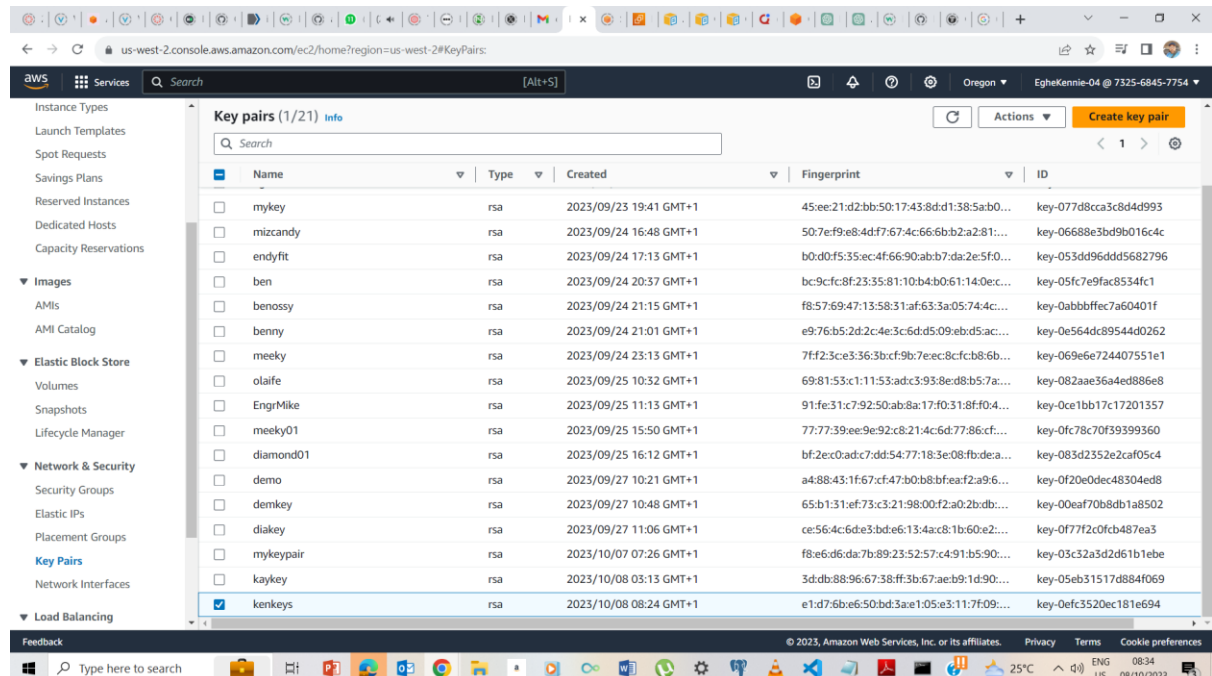
C:\Users\ECL-Francis>
```

Describe Key Pairs with the command aws ec2 describe-key-pairs

```
Command Prompt - aws ec2 describe-key-pairs

{
  "KeyPairs": [
    {
      "KeyPairId": "key-01c65068fc2760aa0",
      "KeyFingerprint": "0c:56:e2:61:e0:51:a9:e9:02:b8:52:d3:60:3b:ef:b0:9b:a9:c4:b6",
      "KeyName": "instance-key-pair",
      "KeyType": "rsa",
      "Tags": [],
      "CreateTime": "2023-09-23T09:00:10.600000+00:00"
    },
    {
      "KeyPairId": "key-07490544df260556",
      "KeyFingerprint": "88:26:5b:a5:f6:65:28:2c:2a:a2:3c:cc:61:6e:fb:2e:1b:07:38",
      "KeyName": "CalebKeyPair",
      "KeyType": "rsa",
      "Tags": [],
      "CreateTime": "2023-09-23T11:15:59.950000+00:00"
    },
    {
      "KeyPairId": "key-01290ac96bb4299dc",
      "KeyFingerprint": "85:42:fc:f7:b2:ae:ab:4e:77:46:e1:59:aa:b3:6f:f8:bb:56:f9:3f",
      "KeyName": "Benkeypair",
      "KeyType": "rsa",
      "Tags": [],
      "CreateTime": "2023-09-23T12:55:12.280000+00:00"
    },
    {
      "KeyPairId": "key-bd52f9b14f7958b5",
      "KeyFingerprint": "8d:c7:31:82:b8:21:24:43:76:1f:d0:05:98:53:cc:f2:29:75:e5:cb",
      "KeyName": "kenkeys",
      "KeyType": "rsa",
      "Tags": [],
      "CreateTime": "2023-10-08T08:24:00.000000+00:00"
    }
  ]
}
```

Confirm that the Key pair is created on the AWS Management Console



Name	Type	Created	Fingerprint	ID
mykey	rsa	2023/09/23 19:41 GMT+1	45:ee:21:d2:bb:50:17:43:8d:d1:38:5a:b0...	key-077d8cca3c8d4d993
mizcandy	rsa	2023/09/24 16:48 GMT+1	50:7ef9:e8:4d:f7:67:4c:66:6b:b2:a2:81:...	key-06688e3bd9b016c4c
endyfit	rsa	2023/09/24 17:13 GMT+1	b0:d0:f5:35:ec:4f:66:90:abb7:da:2e:5f:...	key-053dd96dd5682796
ben	rsa	2023/09/24 20:37 GMT+1	bc9:fc:8f:23:35:81:10:b4:b0:61:14:0e:...	key-05fc7e9fac8534fc1
benossy	rsa	2023/09/24 21:15 GMT+1	f8:57:69:47:13:58:31:af:63:3a:05:74:4c:...	key-0abbbffec7a60401f
benney	rsa	2023/09/24 21:01 GMT+1	e9:76:b5:2d:2c:4e:3c:6d:d5:09:eb:d5:ac:...	key-0e564dc89544d0262
meeky	rsa	2023/09/24 23:13 GMT+1	7ff2:3ce3:36:3b:cf:9b:7e:ec:8c:fb:8b:...	key-0696e6724407551e1
olaife	rsa	2023/09/25 10:32 GMT+1	69:81:53:c1:11:53:adc3:93:8e:d8:b5:7a:...	key-082aae364ed886e8
EngrMike	rsa	2023/09/25 11:13 GMT+1	91:fe:31:c7:92:50:ab:8a:17:f0:31:8f:f0:...	key-0ce1bb17c17201357
meeky01	rsa	2023/09/25 15:50 GMT+1	77:77:39:ee:9e:92:c8:21:4c:6d:77:86:cf:...	key-0fc78c70f39399360
diamond01	rsa	2023/09/25 16:12 GMT+1	bf:2e:c0:adc7:dd:54:77:18:3e:08:fb:dea:...	key-083d2352e2caf05c4
demo	rsa	2023/09/27 10:21 GMT+1	a4:88:43:1f:67:cf:47:b0:b8:bfeaf2a9:6...	key-0f20e0dec48304ed8
demkey	rsa	2023/09/27 10:48 GMT+1	65:b1:31:ef:73:c3:21:98:00:f2:a0:2b:db:...	key-00eaf70b8db1a8502
diakey	rsa	2023/09/27 11:06 GMT+1	ce:56:4c:6d:e3:bde6:13:4ac8:1b:60:e2:...	key-0f77f2c0fcb487ea3
mykeypair	rsa	2023/10/07 07:26 GMT+1	f8:e6:d6:da:7b:89:23:52:57:c4:91:b5:90:...	key-03c32a3d2d61b1ebe
kaykey	rsa	2023/10/08 03:13 GMT+1	3d:db:88:96:67:38:ff:3b:67:aeb9:1d:90:...	key-05eb31517d884f069
kenkeys	rsa	2023/10/08 08:24 GMT+1	e1:d7:6b:e6:50:bd:3a:e1:05:e3:11:7f:09:...	key-0efc3520ec181e694

STEP 2 CREATE SECURITY GROUP

1, We first check for the VPC with the command `aws ec2 describe-vpcs`

```
Command Prompt
"KeyType": "rsa",
"Tags": [],
"CreateTime": "2023-09-24T15:48:33.386000+00:00"

C:\Users\ECL-Francis>aws ec2 describe-vpcs
{
  "Vpcs": [
    {
      "CidrBlock": "172.31.0.0/16",
      "DhcpOptionsId": "dopt-0c5804041f8d9422e",
      "State": "available",
      "VpcId": "vpc-0450fd5913e0189b1",
      "OwnerId": "732568457754",
      "InstanceTenancy": "default",
      "CidrBlockAssociationSet": [
        {
          "AssociationId": "vpc-cidr-assoc-06edf52470940a7f9",
          "CidrBlock": "172.31.0.0/16",
          "CidrBlockState": {
            "State": "associated"
          }
        }
      ],
      "IsDefault": true
    }
  ]
}

C:\Users\ECL-Francis>
```

Copy the vpc id and input it in the create security group command and the security group id is revealed.

```
Command Prompt
usage: aws [options] <command> [<subcommand> ...] [<parameters>]
to see help text, you can run:

    aws help
    aws <command> help
    aws <command> <subcommand> help

Unknown options: group --vpc-id vpc-0450fd5913e0189b1, security

C:\Users\ECL-Francis>aws ec2 create-security-group group-name kays-sg --description "kay security group" --vpc-id vpc-0450fd5913e0189b1
usage: aws [options] <command> [<subcommand> ...] [<parameters>]
to see help text, you can run:

    aws help
    aws <command> help
    aws <command> <subcommand> help

aws: error: the following arguments are required: --group-name

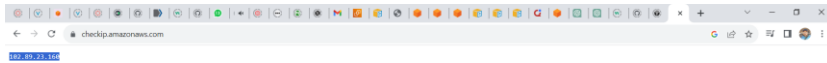
C:\Users\ECL-Francis>aws ec2 create-security-group --group-name kays-sg --description "kay security group" --vpc-id vpc-0450fd5913e0189b1
{
  "GroupId": "sg-0defdda5749f01328"
}

C:\Users\ECL-Francis>
```

To confirm the creation of the security Group we just created, we use the command “`aws ec2 describe-security-groups`”. And also confirm from the AWS Management console Security Groups

TO CREATE SECURITY GROUP

First Check Ip address using checkip.amazonaws.com



CREATE SECURITY GROUP

Create the Security group using aws ec2 authorize-security-group-ingress --group-id (security group id) --protocol tcp --port 22 --cidr(ip addresss got earlier)

CREATING EC2 INSTANCE

```
Command Prompt - aws ec2 run-instances --image-id ami-09100e341bda441c0 --count 1 --instance-type t2.micro --key-name kenkeys --security-group-ids sg-0b51e59a22b863045 --subnet-id subnet-0d4a74ff3ef2bf33b
C:\Users\ECL-Francis>aws ec2 run-instances --image-id ami-09100e341bda441c0 --count 1 --instance-type t2.micro --key-name kenkeys --security-group-ids sg-0b51e59a22b863045 --subnet-id subnet-0d4a74ff3ef2bf33b
{
  "Groups": [],
  "Instances": [
    {
      "AmiLaunchIndex": 0,
      "ImageId": "ami-09100e341bda441c0",
      "InstanceId": "i-090f03e1f06a125d5",
      "InstanceType": "t2.micro",
      "KeyName": "kenkeys",
      "LaunchTime": "2023-10-08T09:16:30+00:00",
      "Monitoring": {
        "State": "disabled"
      },
      "Placement": {
        "AvailabilityZone": "us-west-2c",
        "GroupName": "",
        "Tenancy": "default"
      },
      "PrivateDnsName": "ip-172-31-2-78.us-west-2.compute.internal",
      "PrivateIpAddress": "172.31.2.78",
      "ProductCodes": [],
      "PublicDnsName": "",
      "State": {
        "Code": 0,
        "Name": "pending"
      },
      "StateTransitionReason": "",
      "SubnetId": "subnet-0d4a74ff3ef2bf33b",
      ... More ...
    }
  ]
}
```

CONFIRM THE INSTANCE

The screenshot shows the AWS Management Console for the us-west-2 region. The left sidebar contains navigation links for EC2 Dashboard, EC2 Global View, Events, Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images, Elastic Block Store, and Network & Security. The main content area displays the 'Instances (2)' page with a table of running instances. A 'Select an instance' dialog box is open in the foreground.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
Kay	i-04bd61d903e655c69	Running	t2.micro	2/2 checks passed	No alarms	us-west-2b	ec2-35-90-7-63.a
-	i-090f03e1f06a125d5	Running	t2.micro	2/2 checks passed	No alarms	us-west-2c	ec2-34-209-66-2

We have successfully created and ran and EC2 Instance using CLI

TERMINATING INSTANCES

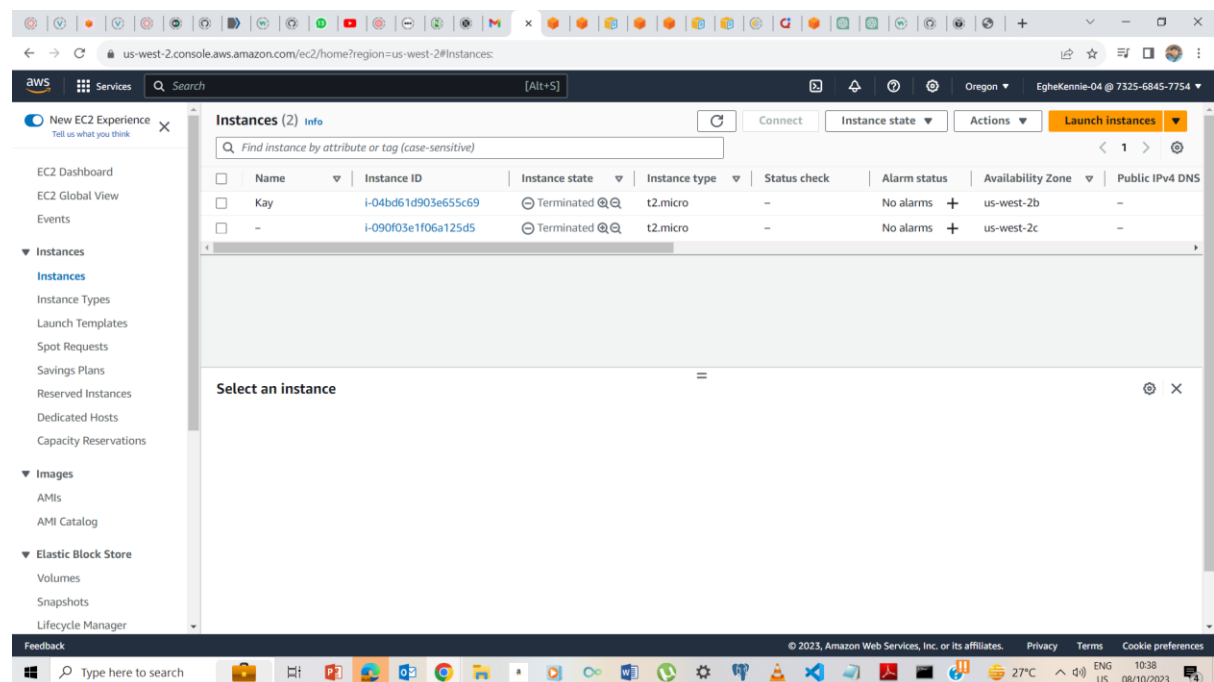
```
Command Prompt
C:\Users\ECL-Francis>aws ec2 terminate-instances i-090f03e1f06a125d5
usage: aws [options] <command> [<subcommand> [<subcommand> ...] [parameters]]
to see help text, you can run:
    aws help
    aws <command> help
    aws <command> <subcommand> help
aws: error: the following arguments are required: --instance-ids

C:\Users\ECL-Francis>aws ec2 terminate-instances --instance-ids i-090f03e1f06a125d5
{
  "TerminatingInstances": [
    {
      "CurrentState": {
        "Code": 32,
        "Name": "shutting-down"
      },
      "InstanceId": "i-090f03e1f06a125d5",
      "PreviousState": {
        "Code": 16,
        "Name": "running"
      }
    }
  ]
}

C:\Users\ECL-Francis>aws ec2 terminate-instances --instance-ids i-04bd61d903e655c69
{
  "TerminatingInstances": [
    {
      "CurrentState": {
        "Code": 32,
        "Name": "shutting-down"
      },
      "InstanceId": "i-04bd61d903e655c69",
      "PreviousState": {
        "Code": 16,
        "Name": "running"
      }
    }
  ]
}

C:\Users\ECL-Francis>
```

We terminate by using `aws ec2 terminate-instances - - instance id (instance id number)`



Confirmation that the Instance has been terminated.

DELETE KEY PAIR USING THE COMMAND `aws ec2 delete-key-pair --keyname (name of key)`

```
Command Prompt
release-ipam-pool-allocation      replace-iam-instance-profile-association
replace-network-acl-association   replace-network-acl-entry
replace-route                    replace-route-table-association
replace-transit-gateway-route    replace-vpn-tunnel
report-instance-status           request-spot-fleet
request-spot-instances           reset-address-attribute
reset-ec2-default-vpc-key-id     reset-fpga-image-attribute
reset-image-attribute            reset-instance-attribute
reset-network-interface-attribute reset-snapshot-attribute
restore-address-to-classic       restore-image-from-recycle-bin
restore-managed-prefix-list-version restore-snapshot-from-recycle-bin
restore-snapshot-tier            revoke-client-vpn-ingress
revoke-security-group-egress     revoke-security-group-ingress
run-instances                   run-scheduled-instances
search-local-gateway-routes      search-transit-gateway-multicast-groups
search-transit-gateway-routes    send-diagnostic-interrupt
start-instances                  start-network-insights-access-scope-analysis
start-network-insights-analysis start-vpc-endpoint-service-private-dns-verification
stop-instances                   terminate-client-vpn-connections
terminate-instances              unassign-ipw-addresses
unassign-private-ip-addresses    unassign-private-nat-gateway-address
unmonitor-instances              update-security-group-rule-descriptions-egress
update-security-group-rule-descriptions-ingress | withdraw-byoip-cidr
wait                             | help

C:\Users\ECL-Francis>clear
'clear' is not recognized as an internal or external command,
operable program or batch file.

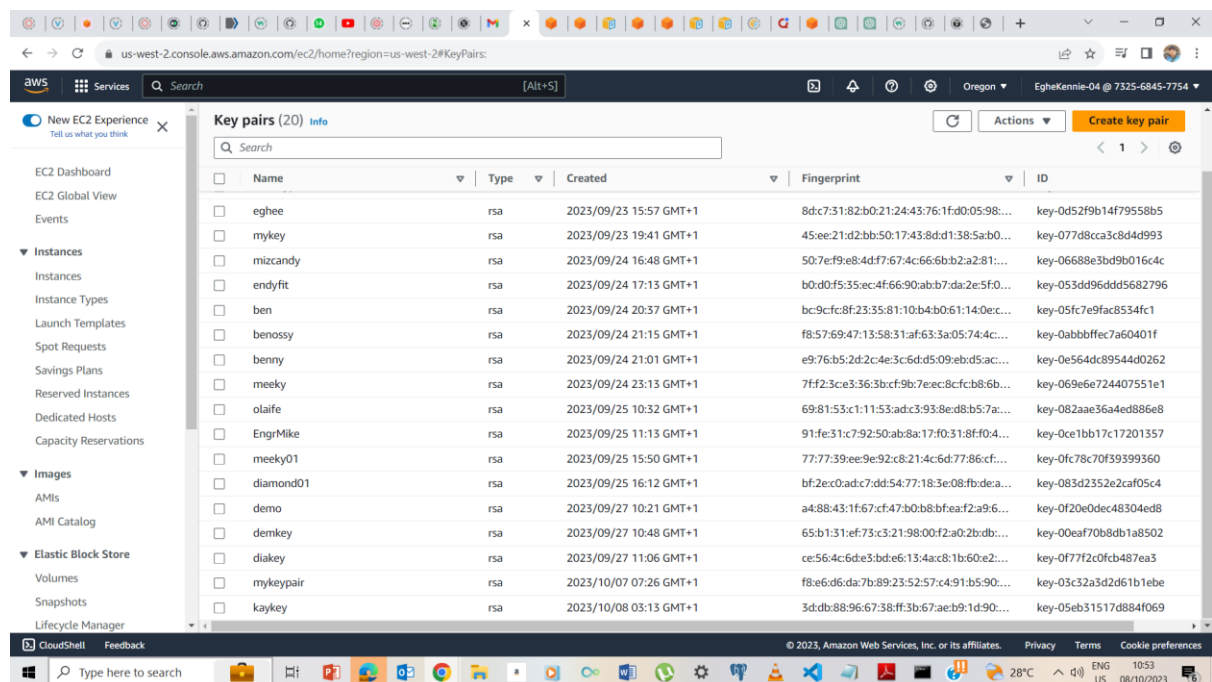
C:\Users\ECL-Francis>aws ec2 delete-key-pair --key-pair-name kenkeys

usage: aws [options] <command> [<subcommand>] [<subcommand> ...] [parameters]
To see help text, you can run:

    aws help
    aws <command> help
    aws <command> <subcommand> help

Unknown options: --key-pair-name, kenkeys

C:\Users\ECL-Francis>aws ec2 delete-key-pair --key-name kenkeys
{
  "Returns": true,
  "KeyPairId": "key-0efc3520ec181e694"
}
```



Kenkeys key pair deleted

DELETE SECURITY GROUPS by the command `aws ec2 delete-security-group --group-name (name of group)`

```
Command Prompt
replace-route
replace-transit-gateway-route
report-instance-status
request-spot-instances
reset-eks-default-kms-key-id
reset-image-attribute
reset-network-interface-attribute
restore-address-to-classic
restore-managed-prefix-list-version
restore-snapshot-tier
revoke-security-group-egress
run-instances
search-local-gateway-routes
search-transit-gateway-routes
start-instances
start-network-insights-analysis
stop-instances
terminate-instances
unassign-private-ip-addresses
unmonitor-instances
update-security-group-rule-descriptions-ingress
wait
replace-route-table-association
replace-vpn-tunnel
request-spot-fleet
reset-address-attribute
reset-fpga-image-attribute
reset-instance-attribute
reset-snapshot-attribute
restore-image-from-recycle-bin
restore-snapshot-from-recycle-bin
revoke-client-vpn-ingress
revoke-security-group-ingress
run-scheduled-instances
search-transit-gateway-multicast-groups
send-diagnostic-interrupt
start-network-insights-access-scope-analysis
start-vpc-endpoint-service-private-dns-verification
terminate-client-vpn-connections
unassign-ipv6-addresses
unassign-private-nat-gateway-address
update-security-group-rule-descriptions-egress
withdraw-byoip-cidr
help

C:\Users\ECL-Francis>clear
"clear" is not recognized as an internal or external command,
operable program or batch file.

C:\Users\ECL-Francis>aws ec2 delete-key-pair --key-pair-name kenkeys

usage: aws [options] <command> [<subcommand> ...] [parameters]
To see help text, you can run:

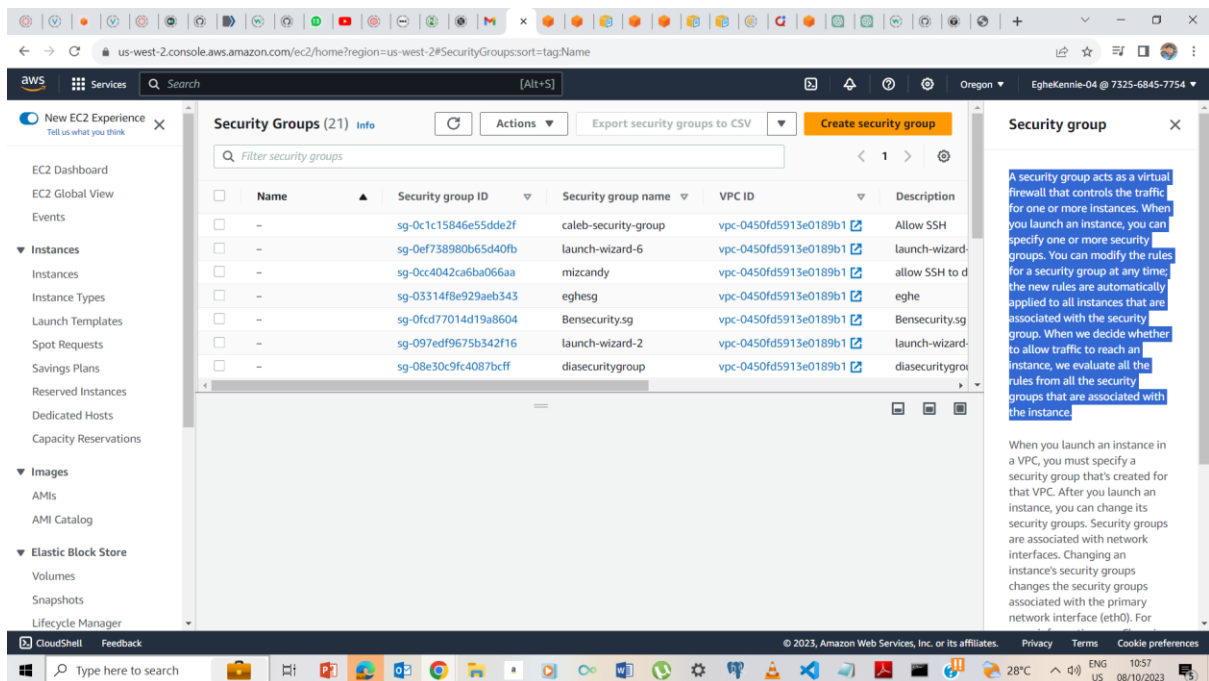
    aws help
    aws <command> help
    aws <command> <subcommand> help

Unknown options: --key-pair-name, kenkeys

C:\Users\ECL-Francis>aws ec2 delete-key-pair --key-name kenkeys
{
  "Return": true,
  "KeyPairId": "key-0efc3520ec181e694"
}

C:\Users\ECL-Francis>aws ec2 delete-security-group --group-name kays-sg

C:\Users\ECL-Francis>
```



Confirmation of the security group deleted.

