EC2

(I)

1. Launch an instance with .pem file
2. Check all the 3 boxes in Network settings
3. Connect it using SSH client or GUI
4. Go to downloads
   cd Downloads
5. Sudo yum update -y
   sudo yum install httpd -y
   Sudo systemctl start httpd
   sudo systemctl enable httpd.
6. Open public IPv4 and give http://=

(II)

1. Launch an EC2 instance with .ppk file (Select ubuntu in AMI)
2. Check all the 3 boxes in Network settings
3. Connect it using putty. Copy the IP address from ubuntu.
4. paste it in putty host name & SSH → credentials → Auth browse the key (ppk) & accept
5. sudo apt-get update -y
   sudo apt install docker.io -y
   sudo docker pull nginx
   sudo docker run -d -p 80:80 --name my-nginx nginx
   sudo docker exec -it my-nginx bash
   cd /var/share/nginx/html/
   apt update → apt install nano -y
   nano index.html
   ctrl+O & enter → ctrl+X → exit

S3

1. Create a bucket
2. Upload .html file
3. bukel → permissions → block public acces (Uncheck) → ACL / Object ownership
   ACL → Everyone public access.
   Object ownership → ACL's enabled
4. Object → permission → ACL → everyone public access.
5. Copy object URL & paste
6. Bucket → permissions → Versioning → On
   → Edit static web hosting → Enable → .html
7. Copy endpoint URL & paste (web hosting)
8. Delete the object and toggle the show versions option & delete the marker file to get back the del object.
9. Create 2 buckets (src, dest)
10. Enable versioning for both
11. Go to src → management and create
12. Create a replication role
13. Give replication rule name
14. Check apply to all objects
15. Give IAM role as LabRole
16. Same Creating.
17. Upload a file in src and you can see it in the dest.

CloudFront

1. Go to S3
2. Create a bucket
3. Give name
4. Choose ACLs enabled
5. Uncheck "block all public access"
6. Click "create bucket".
7. Go to bucket properties
8. Edit static web hosting
9. Click Enable
10. Type index.html.
11. Save changes
12. Upload 2 files (index.html & .jpg)
13. Edit ACL for 2 files
    Give permission Everyone (public access)
14. Go to cloudfront
15. Create a distribution
16. Select origin domain
17. WAF → Enable security protections
18. Select bucket edit ACL
19. Give all permissions
20. Now paste the cloudfront.net web page & open

SQS

1. Open SQS service
2. Click "create queue"
3. Give name and create queue
4. Go to Lambda service
5. "Create Lambda" ~~function~~ click
6. Clicke "Use a blueprint".
7. Give function name
8. Select "create a new role from AWS policy".
9. Give "Role name".
10. Create function
11. Add a trigger → SQS
12. Give Queue. name
13. Go to IAM role → roles
14. Select the role
15. Add permissions → Attach policy
16. Give "SQS Full Acess" & add permission
17. Now add trigger.
18. Change the Lambda code to send key-value pairs message.
19. Send message
20. Check the output

SNS

1. Go to SNS
2. Click create topic
3. Select standard
4. Give name & create SNS
5. Create subscription
6. Select email
7. Give email
8. Confirm
9. Publish message
10. Any message
11. Send & check mail
12. Create another topic for number
13. Select standard
14. Give name and create SNS
15. Create subscription
16. Select SMS
17. Add & Confirm phone number
18. Publish message → Give any msg
19. Check notifications
20. Create another topic
21. Click create subscription
22. Select email & confirm subscription
23. Creat a bucket
24. Uncheck block all public access
    Check I acknowledge
25. Now create bucket.
26. Go to bucket → properties → Create new event notificatn
27. Gives event name
28. Check All object events
    Check All object removal events
29. Select SNS topic
30. It gives error
31. Paste the code in
    SNS - topic - Access policy (optional)
    and chage the code.
32. Upload and check the mail.

IAM

1. Create ~~user~~ IAM user
2. Give username
3. Check the box (Provide user access)
   → I want to create IAM user (checkbox)
   → custom password
4. Uncheck the box "users must create"
5. Attach policy (select)
6. Give S3fullAccess
7. Download & retrieve passwords
8. Go to security credentials ~~for~~ of roles
9. Create Access key
10. Select CLI & check box and next
11. Download Access key & SAK.csv file credentials
12. Signout of root & login to IAM user.
13. Create bucket in IAM
14. Try EC2 → it shows error
15. Go back to root.
16. aws configure
    Access key
    Secret Access key
    Region
    json.
17. ~~Go~~ aws s3 ls

    aws s3 rb s3://bucketname
18. Create group
19. Give group name
20. Attach ~~go~~ users & policy
21. Create an basic EC2 instance
22. Connect using ssh key
23. aws configure

    }
24. aws ec2 describe-instances --region us-east-1
25. Select users → add permission

    ↓

    Administrators ← Attach poli
26. aws iam create-user --~~username~~ user-name iamuser
27. aws iam create-login-profile --user-name iamuser --password newuser9aj@123

Lambda

1. Create a lambda function → Author from scratch
2. Give function name
3. Choose Python 3.9
4. Use an existing role ~~& give name~~
5. Go to IAM role and select AWS service and select Lambda in use case & click next
6. Add permissions → ~~S3 Full~~ DynamoDB → S3 Full Access
7. Give role name
8. Select the role in Lambda fune
9. Add trigger → S3
10. Create a S3 bucket (~~default~~) (give name & create)
11. Choose bucket in lambda
12. Event types → First ↓ & Create
13. Create a table & give name as 'newtable'
14. Partition key → unique
15. Copy the code and and paste in Code section of lambda and deploy
16. Upload the object in bucket and
17. Check it in explore items