

```
terraform > cat main.tf
1 resource "aws_ecr_repository" "flask" {
2   name = "flask-backend"
3 }
4
5 resource "aws_ecr_repository" "express" {
6   name = "express-frontend"
7 }
8 }

terraform >

PS F:\workspace\terraform> aws sts get-caller-identity
{
  "UserId": "382749986244",
  "Account": "382749986244",
  "Arn": "arn:aws:iam::382749986244:root"
}

PS F:\workspace\terraform> docker login --username A65 --password-stdin 382749986244.dkr.ecr.ap-south-1.amazonaws.com
Password:
PS F:\workspace\terraform> docker login --username A65 --password-stdin 382749986244.dkr.ecr.ap-south-1.amazonaws.com
Password:
PS F:\workspace\terraform> aws ecr get-login-password --region ap-south-1 |
>> docker login --username A65 --password-stdin 382749986244.dkr.ecr.ap-south-1.amazonaws.com
>>
Login Succeeded
PS F:\workspace\terraform>
PS F:\workspace\terraform>
```

Create terraform file to create ecr resource and we also login to aws with temporary password now we initialize and apply it



The screenshot shows the AWS Private repositories page with two repositories listed:

Repository name	URI	Created at	Tag immutability	Encryption type
flask-backend	382749986244.dkr.ecr.ap-south-1.amazonaws.com/flask-backend	January 11, 2020, 18:11:46 (UTC+05:30)	Mutable	DES-256
express-frontend	382749986244.dkr.ecr.ap-south-1.amazonaws.com/express-frontend	January 11, 2020, 18:11:46 (UTC+05:30)	Mutable	DES-256

TWO ECR repository is created now we add docker image in it

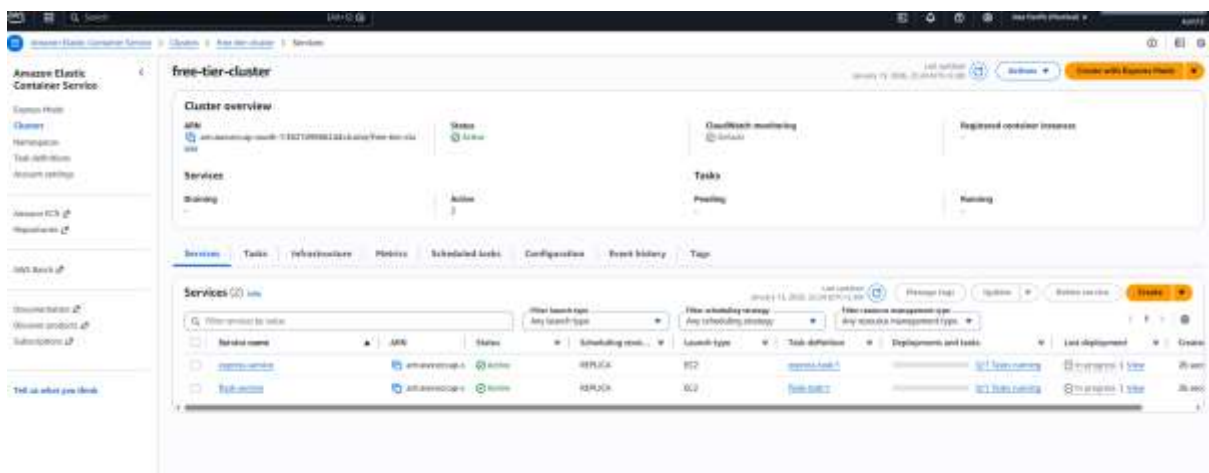
```
17 docker login
18 (password use access token in place of password)
19
20 docker build -t flask-backend ./Backend
21 docker tag flask-backend:latest 302749086244.dkr.ecr.ap-south-1.amazonaws.com/flask-backend:latest
22 docker push 302749086244.dkr.ecr.ap-south-1.amazonaws.com/flask-backend:latest
23
24
25 docker build -t express-frontend ./Frontend
26 docker tag express-frontend:latest 302749086244.dkr.ecr.ap-south-1.amazonaws.com/express-frontend:latest
27 docker push 302749086244.dkr.ecr.ap-south-1.amazonaws.com/express-frontend:latest
```

Use these command to build image



Image pushed successfully on ECR

After this create vpc.tf, security.tf, ecs.tf etc. terraform files to build and deploy docker/ecr images on ecs.



Cluster created and service created using image in ecr and it successfully running