

YESHWANTH CHAUHAN

ELASTIC BEANSTALK -

1. create an Env with below
 - Python install
 - Custom VPC selected
 - ALB
 - Autoscaling
 - SG for the webserver
2. Clone the Env for Dev
3. Change the code and upload it to the Dev Env
4. Swap the URLs to check if the DEv Env is accessible from the first URL.
5. SWAP it back.

YESHWANTH CHAUHAN

The image displays two screenshots of the AWS Elastic Beanstalk console interface, showing the process of creating and monitoring an environment.

Top Screenshot: All environments

The 'All environments' page shows a list of environments. The 'Testapp-env' environment is highlighted, indicating it is the active environment.

Environment name	Health	Application name	Date created	Last modified	URL	Running versions	Platform	Platform state	Tier name
Hi-env (terminated)	-	hi	2023-03-21 12:50:25 UTC+0530	2023-03-21 15:16:47 UTC+0530	Hi-env-eba-tst2gww.us-east-1.elasticbeanstalk.com	Sample Application	Python 3.8 running on 64-bit Amazon Linux 2	Supported	WebServer
Testapp-env	Ok	Testapp	2023-03-21 15:28:58 UTC+0530	2023-03-21 15:31:51 UTC+0530	Testapp-env-eba-pjh5smhz.us-east-1.elasticbeanstalk.com	Sample Application	Python 3.8 running on 64-bit Amazon Linux 2	Supported	WebServer

Bottom Screenshot: Creating Testapp-env

The 'Creating Testapp-env' page shows the progress of creating the environment. The status is 'Creating Testapp-env' and it indicates that the environment will take a few minutes to create.

3:32pm Environment health has transitioned from Pending to Ok. Initialization completed 7 seconds ago and took 3 minutes.

3:31pm Successfully launched environment: Testapp-env

3:31pm Application available at Testapp-env-eba-pjh5smhz.us-east-1.elasticbeanstalk.com.

3:31pm Instance deployment completed successfully.

3:31pm Instance deployment successfully generated a 'Profile'.

3:31pm Created CloudWatch alarm named: awsseb-e-h22vgjpm-sta-AWSEBCloudwatchAlarmLow-10XFAVLL2SF6

3:31pm Created CloudWatch alarm named: awsseb-e-h22vgjpm-sta-AWSEBCloudwatchAlarmHigh-SNQN1L056H5Q

3:31pm Created Auto Scaling group policy named: arn:aws:autoscaling:us-east-1:471963814578:scalingPolicy:9952c36f-ebcb-45bc-d8b92d1dea3:autoScalingGroupName/awsseb-e-h22vgjpm-sta-AWSEBAutoScalingGroup-XGAQ7CE2QJ4C:policyName/awsseb-e-h22vgjpm-sta-AWSEBAutoScalingScaleUpPolicy-vPdVKIpuhQ

3:31pm Created Auto Scaling group policy named: arn:aws:autoscaling:us-east-1:471963814578:scalingPolicy:eeef628-7ded-468a-8d28-8d91625c711d:autoScalingGroupName/awsseb-e-h22vgjpm-sta-AWSEBAutoScalingGroup-XGAQ7CE2QJ4C:policyName/awsseb-e-h22vgjpm-sta-AWSEBAutoScalingScaleDownPolicy-uA1TvcWYr9

3:31pm Added instance [i-069d1ef689ebc8fb3] to your environment.

The image shows two screenshots of the AWS Elastic Beanstalk console. The top screenshot displays the 'Testapp-env' dashboard, which includes sections for Health (Ok), Running version (Sample Application), Platform (Python 3.8), and Recent events. The bottom screenshot shows a 'Congratulations' message for launching the first AWS Elastic Beanstalk Python Application.

Testapp-env Dashboard

Health: Ok

Running version: Sample Application

Platform: Python 3.8 running on 64bit Amazon Linux 2/5.5.0

Recent events

Time	Type	Details
2023-03-21 15:32:24 UTC+0530	INFO	Environment health has transitioned from Pending to Ok. Initialization completed 7 seconds ago and took 3 minutes.
2023-03-21 15:31:51 UTC+0530	INFO	Successfully launched environment: Testapp-env
2023-03-21 15:31:49 UTC+0530	INFO	Application available at Testapp-env.eba-pjh5smhz.us-east-1.elasticbeanstalk.com.
2023-03-21 15:31:34 UTC+0530	INFO	Instance deployment completed successfully.
2023-03-21 15:31:30 UTC+0530	INFO	Instance deployment successfully generated a 'Profile'.

Congratulations

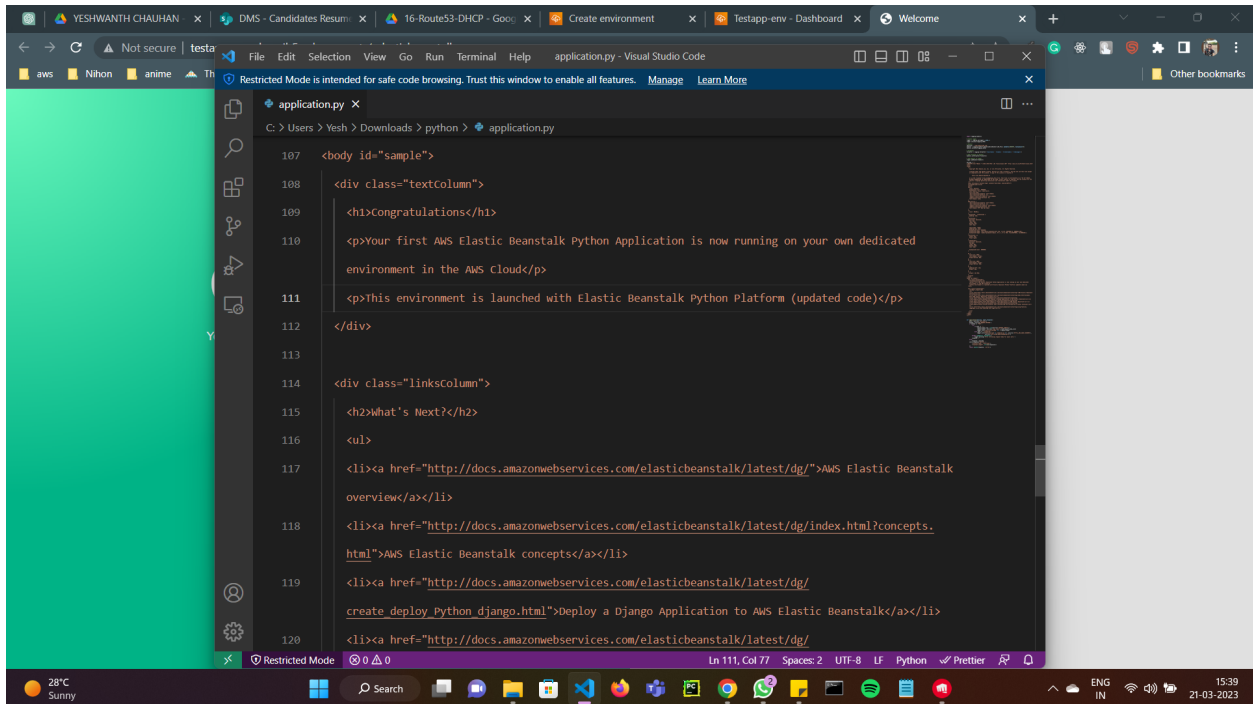
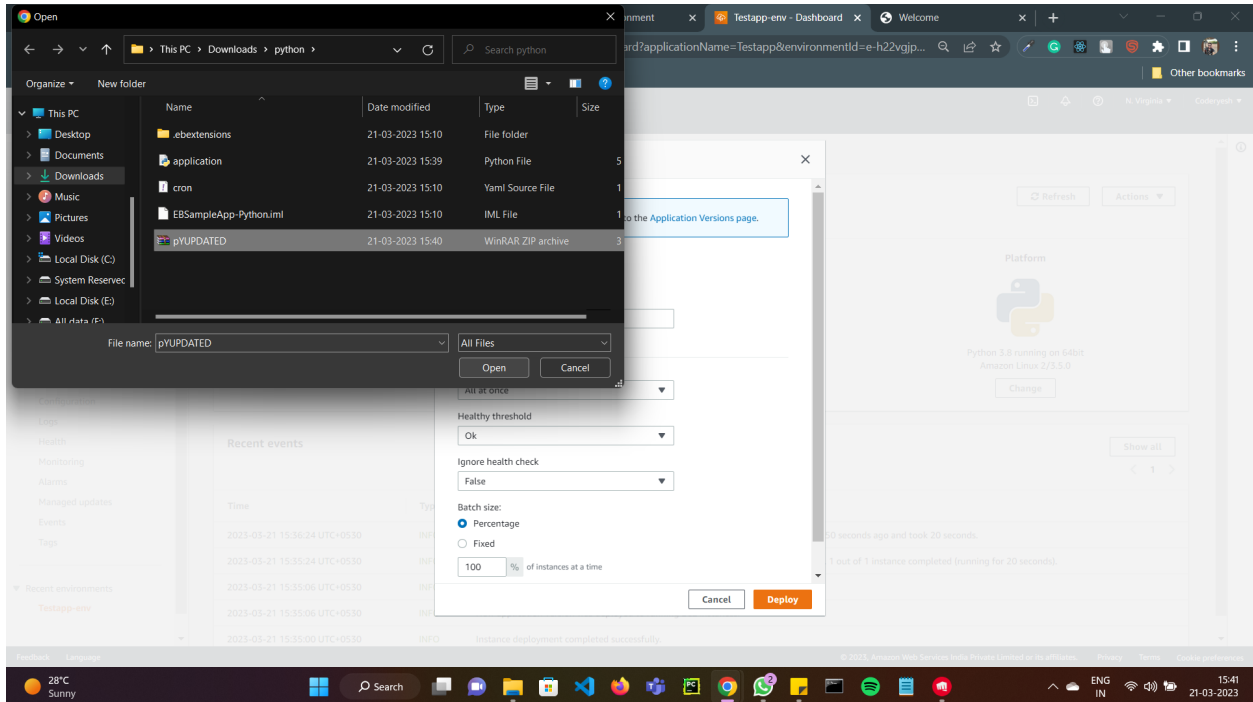
Your first AWS Elastic Beanstalk Python Application is now running on your own dedicated environment in the AWS Cloud

This environment is launched with Elastic Beanstalk Python Platform

What's Next?

- [AWS Elastic Beanstalk overview](#)
- [AWS Elastic Beanstalk concepts](#)
- [Deploy a Django Application to AWS Elastic Beanstalk](#)
- [Deploy a Flask Application to AWS Elastic Beanstalk](#)
- [Customizing and Configuring a Python Container](#)
- [Working with Logs](#)

YESHWANTH CHAUHAN



The screenshot shows the AWS Elastic Beanstalk console. The top section displays a 'Congratulations' message: 'Your first AWS Elastic Beanstalk Python Application is now running on your own dedicated environment in the AWS Cloud. This environment is launched with Elastic Beanstalk Python Platform (updated code)'. To the right, under 'What's Next?', there are links for 'AWS Elastic Beanstalk overview', 'AWS Elastic Beanstalk concepts', 'Deploy a Django Application to AWS Elastic Beanstalk', 'Deploy a Flask Application to AWS Elastic Beanstalk', 'Customizing and Configuring a Python Container', and 'Working with Logs'.

The bottom section shows the 'All environments' table. The table has columns for Environment name, Health, Application name, Date created, Last modified, URL, Running versions, Platform, Platform state, and Tier name. The table lists three environments: 'Hi-env (terminated)', 'Testapp-env', and 'Testapp-env-1'. The 'Testapp-env-1' environment is highlighted in blue and has a green 'OK' health status.

Environment name	Health	Application name	Date created	Last modified	URL	Running versions	Platform	Platform state	Tier name
Hi-env (terminated)	-	hi	2023-03-21 12:50:25 UTC+0530	2023-03-21 15:16:47 UTC+0530	Hi-env-eba-tst2gww.us-east-1.elasticbeanstalk.com	Sample Application	Python 3.8 running on 64bit Amazon Linux 2	Supported	WebServer
Testapp-env	OK	Testapp	2023-03-21 15:28:58 UTC+0530	2023-03-21 15:41:48 UTC+0530	Testapp-env-eba-pjh5smhz.us-east-1.elasticbeanstalk.com	Sample Application-2	Python 3.8 running on 64bit Amazon Linux 2	Supported	WebServer
Testapp-env-1	OK	Testapp	2023-03-21 15:45:02 UTC+0530	2023-03-21 15:48:12 UTC+0530	testapp-env-11.us-east-1.elasticbeanstalk.com	Sample Application-2	Python 3.8 running on 64bit Amazon Linux 2	Supported	WebServer

YESHWANTH CHAUHAN

The screenshot displays the AWS Elastic Beanstalk console interface. A file explorer window is open in the foreground, showing the 'Downloads' folder with files like 'CLONE', 'application', 'cron', and 'EBSampleApp-Python.iml'. The 'CLONE' file is selected. In the background, the Elastic Beanstalk console shows the 'Testapp-env-1' environment. A notification banner at the top states: 'Elastic Beanstalk is updating your environment. To cancel this operation select Abort Current Operation from the Actions dropdown. View Events'. The environment status is 'Grey' under the 'Health' section. The 'Running version' is 'Sample Application-2'. The 'Platform' is 'Python 3.8 running on 64bit Amazon Linux 2/3.5.0'. The 'Recent events' table shows two entries: 'Environment health has transitioned from Pending to Ok. Initialization completed 5 seconds ago and took 3 minutes.' and 'Successfully launched environment: Testapp-env-1'.

File Explorer (Foreground):

- Path: This PC > Downloads > python
- Files:
 - CLONE (Selected)
 - application
 - EBSampleApp-Python.iml

AWS Elastic Beanstalk Console (Background):

- Environment: Testapp-env-1
- Health: Grey
- Running version: Sample Application-2
- Platform: Python 3.8 running on 64bit Amazon Linux 2/3.5.0
- Recent events:

Time	Type	Details
2023-03-21 15:48:23 UTC+0530	INFO	Environment health has transitioned from Pending to Ok. Initialization completed 5 seconds ago and took 3 minutes.
2023-03-21 15:48:12 UTC+0530	INFO	Successfully launched environment: Testapp-env-1

The image is a screenshot of a Windows desktop with two browser windows open. The top window displays a web application titled "CLONE OF TEST01" on a teal background. The text on the page reads: "CLONE OF TEST01" and "I HAVE CLONED THE ORIG CODE HERE AND PUSHED THE NEW UPDTAES (updated code 1.0)". To the right, under "What's Next?", there are several links: "AWS Elastic Beanstalk overview", "AWS Elastic Beanstalk concepts", "Deploy a Django Application to AWS Elastic Beanstalk", "Deploy a Flask Application to AWS Elastic Beanstalk", "Customizing and Configuring a Python Container", and "Working with Logs". The bottom window shows the AWS Elastic Beanstalk console. The left sidebar lists "Elastic Beanstalk" with sub-items: "Environments", "Applications", and "Change history". The main content area is titled "Swap environment URLs" and includes a warning: "Swapping the environment URL will modify the Route 53 DNS configuration, which may take a few minutes. Your application will continue to run while the changes are propagated." Below this, the "Environment details" section shows: "Environment name: Testapp-env (e-h22vgjpamm)" and "Environment URL: Testapp-env-eba-gjh5smhz.us-east-1.elasticbeanstalk.com". The "Select an environment to swap" section shows a dropdown menu with "Testapp-env-1 (e-aceefsg9v)" selected. At the bottom right of the console, there are "Cancel" and "Swap" buttons. The Windows taskbar at the bottom shows the date as 21-03-2023 and the time as 17:07.

CLONE OF TEST01

I HAVE CLONED THE ORIG CODE HERE AND PUSHED THE NEW UPDTAES (updated code 1.0)

What's Next?

- [AWS Elastic Beanstalk overview](#)
- [AWS Elastic Beanstalk concepts](#)
- [Deploy a Django Application to AWS Elastic Beanstalk](#)
- [Deploy a Flask Application to AWS Elastic Beanstalk](#)
- [Customizing and Configuring a Python Container](#)
- [Working with Logs](#)

Swap environment URLs

When you swap an environment's URL with another environment's URL, you can deploy versions with no downtime. [Learn more](#)

⚠ Swapping the environment URL will modify the Route 53 DNS configuration, which may take a few minutes. Your application will continue to run while the changes are propagated.

Environment details

Environment name:
Testapp-env (e-h22vgjpamm)

Environment URL:
Testapp-env-eba-gjh5smhz.us-east-1.elasticbeanstalk.com

Select an environment to swap

Environment name:
Testapp-env-1 (e-aceefsg9v)

Environment URL:
testapp-env-11.us-east-1.elasticbeanstalk.com

Cancel Swap

