RoundTower®
Business impact through technology

Terraform Feature Flag Deployments with Lambda Functions

TENNIS SMITH

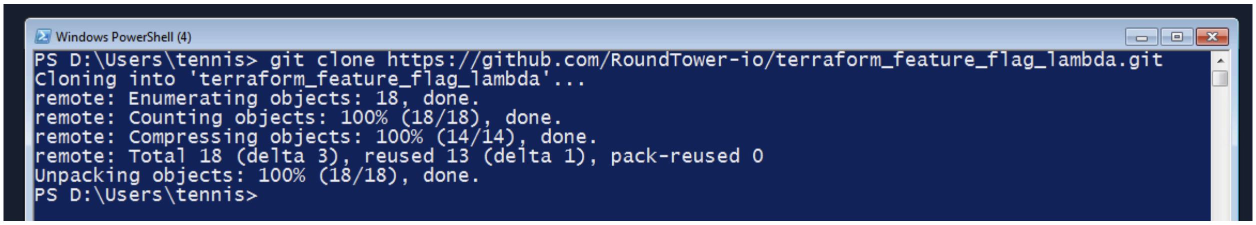
Table of Contents

Overview

The purpose of this lab is to illustrate the use of feature flags. Feature flags are defined as *"…a technique to turn some functionality of your application off, via configuration, without deploying new code. Feature flags play key part in CI scheme where features are constantly being deployed but not necessarily "released" into production."*[*] It is an extremely useful technique of introducing new functionality in an existing application.

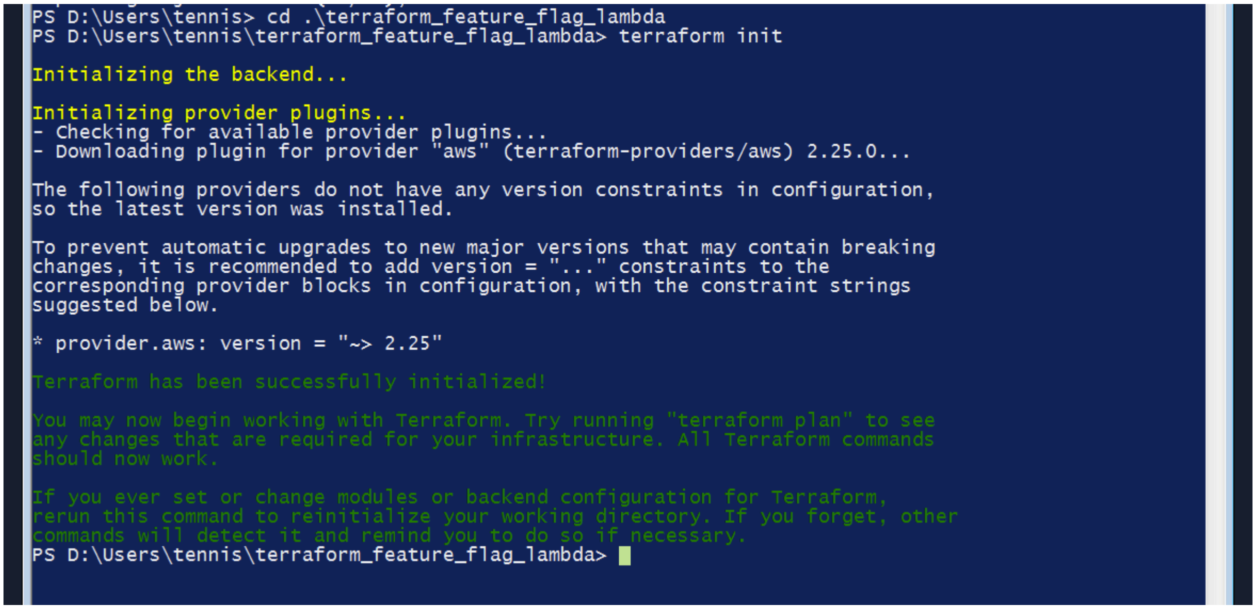
Lab 1

1. Open a shell in your WorkSpace
2. Clone a copy of https://github.com/RoundTower-io/terraform_feature_flag_lambda.git



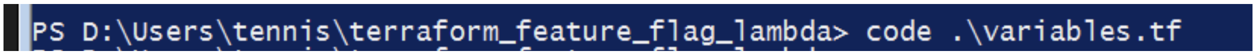
```
PS D:\Users\tennis> git clone https://github.com/RoundTower-io/terraform_feature_flag_lambda.git
Cloning into 'terraform_feature_flag_lambda'...
remote: Enumerating objects: 18, done.
remote: Counting objects: 100% (18/18), done.
remote: Compressing objects: 100% (14/14), done.
remote: Total 18 (delta 3), reused 13 (delta 1), pack-reused 0
Unpacking objects: 100% (18/18), done.
PS D:\Users\tennis>
```

3. Go into the "terraform_feature_flag_lambda" directory and initialize terraform with the command
   `terraform init`



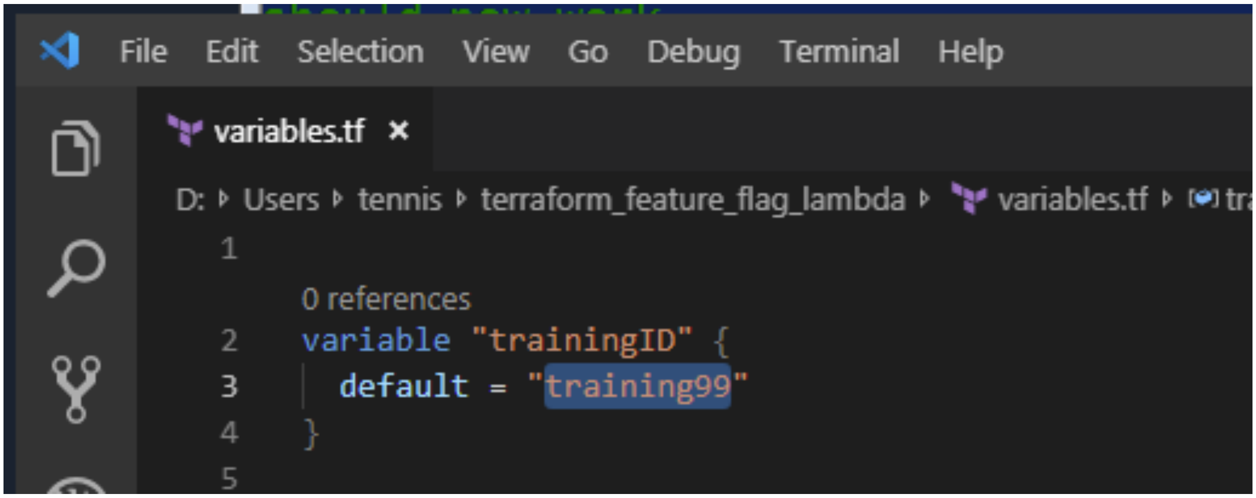
```
PS D:\Users\tennis> cd .\terraform_feature_flag_lambda
PS D:\Users\tennis\terraform_feature_flag_lambda> terraform init

Initializing the backend...

Initializing provider plugins...
- Checking for available provider plugins...
- Downloading plugin for provider "aws" (terraform-providers/aws) 2.25.0...

The following providers do not have any version constraints in configuration,
so the latest version was installed.

To prevent automatic upgrades to new major versions that may contain breaking
changes, it is recommended to add version = "..." constraints to the
corresponding provider blocks in configuration, with the constraint strings
suggested below.

* provider.aws: version = "~> 2.25"

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
PS D:\Users\tennis\terraform_feature_flag_lambda>
```

4. Edit the "variables.tf" file and change "training99" to match your training ID



```
PS D:\Users\tennis\terraform_feature_flag_lambda> code .\variables.tf
```

[*] stackoverflow - https://stackoverflow.com/questions/7707383/what-is-a-feature-flag

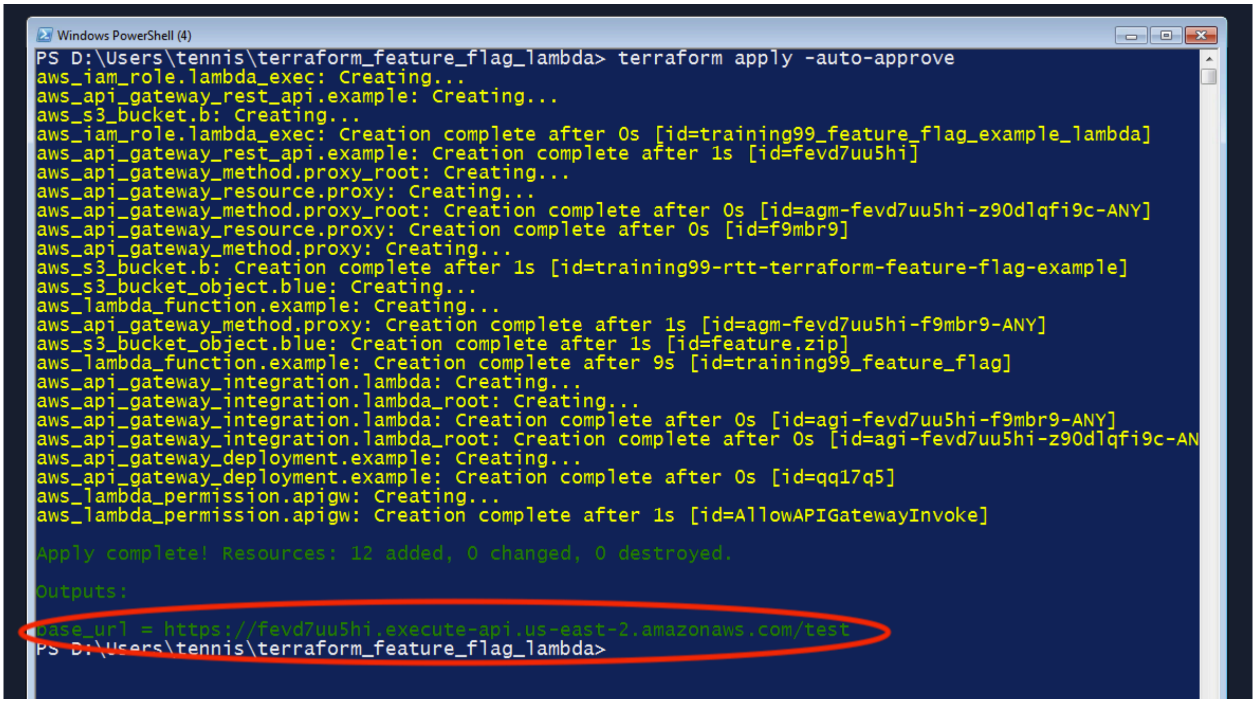

File
Edit
Selection
View
Go
Debug
Terminal
Help
variables.tf
D: › Users › tennis › terraform_feature_flag_lambda › variables.tf › tr
0 references

```
variable "trainingID" {
  default = "training99"
}
```

save and exit the file

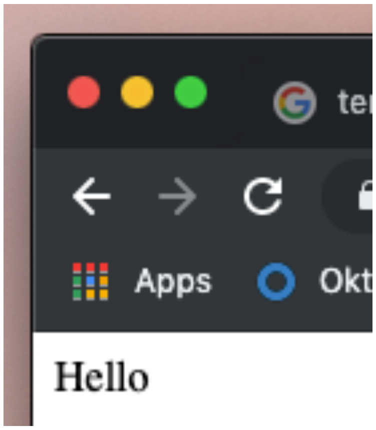
5. Create the lambda function by applying the terraform script

```
terraform apply -auto-approve
```



```
PS D:\Users\tennis\terraform_feature_flag_lambda> terraform apply -auto-approve
aws_iam_role.lambda_exec: Creating...
aws_api_gateway_rest_api.example: Creating...
aws_s3_bucket.b: Creating...
aws_iam_role.lambda_exec: Creation complete after 0s [id=training99_feature_flag_example_lambda]
aws_api_gateway_rest_api.example: Creation complete after 1s [id=fevd7uu5hi]
aws_api_gateway_method.proxy_root: Creating...
aws_api_gateway_resource.proxy: Creating...
aws_api_gateway_method.proxy_root: Creation complete after 0s [id=agm-fevd7uu5hi-z90dlqfi9c-ANY]
aws_api_gateway_resource.proxy: Creation complete after 0s [id=f9mbr9]
aws_api_gateway_method.proxy: Creating...
aws_s3_bucket.b: Creation complete after 1s [id=training99-rtt-terraform-feature-flag-example]
aws_s3_bucket_object.blue: Creating...
aws_lambda_function.example: Creating...
aws_api_gateway_method.proxy: Creation complete after 1s [id=agm-fevd7uu5hi-f9mbr9-ANY]
aws_s3_bucket_object.blue: Creation complete after 1s [id=feature.zip]
aws_lambda_function.example: Creation complete after 9s [id=training99_feature_flag]
aws_api_gateway_integration.lambda: Creating...
aws_api_gateway_integration.lambda_root: Creating...
aws_api_gateway_integration.lambda: Creation complete after 0s [id=agi-fevd7uu5hi-f9mbr9-ANY]
aws_api_gateway_integration.lambda_root: Creation complete after 0s [id=agi-fevd7uu5hi-z90dlqfi9c-AN
aws_api_gateway_deployment.example: Creating...
aws_api_gateway_deployment.example: Creation complete after 0s [id=qq17q5]
aws_lambda_permission.apigw: Creating...
aws_lambda_permission.apigw: Creation complete after 1s [id=AllowAPIGatewayInvoke]

Apply complete! Resources: 12 added, 0 changed, 0 destroyed.

Outputs:

base_url = https://fevd7uu5hi.execute-api.us-east-2.amazonaws.com/test
PS D:\Users\tennis\terraform_feature_flag_lambda>
```

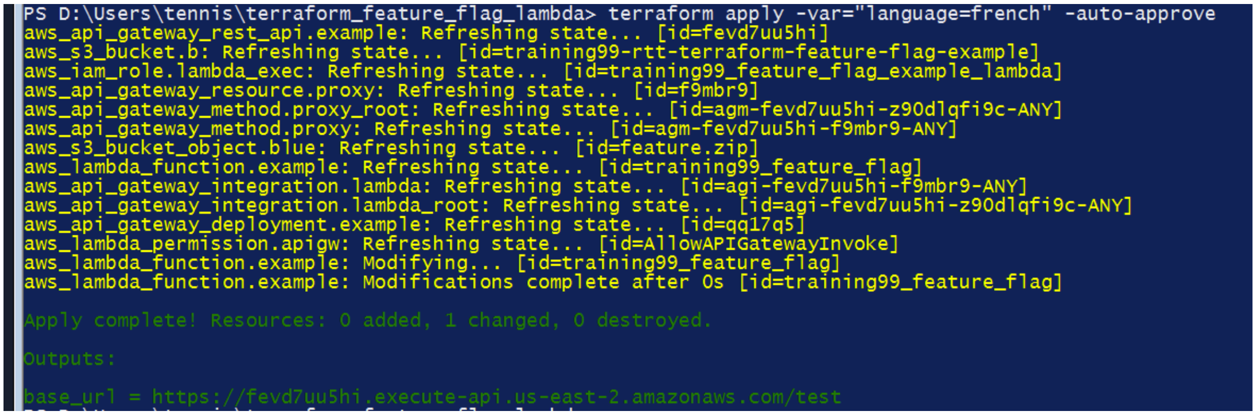
6. Notice the base_url in the above output. Copy that url to your browser and you should see something like this



Hello

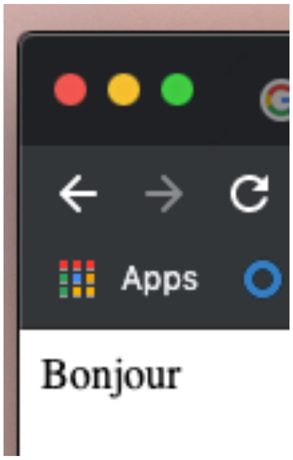
7. Now, we want to use a feature toggle to change the language of the above output. To do that, we pass a special environment variable to the lambda function
8. Enter this command

```
terraform apply -var="language=french" -auto-approve
```



```
PS D:\Users\tennis\terraform_feature_flag_lambda> terraform apply -var="language=french" -auto-approve
aws_api_gateway_rest_api.example: Refreshing state... [id=fevd7uu5hi]
aws_s3_bucket.b: Refreshing state... [id=training99-rtt-terraform-feature-flag-example]
aws_iam_role.lambda_exec: Refreshing state... [id=training99_feature_flag_example_lambda]
aws_api_gateway_resource.proxy: Refreshing state... [id=f9mbr9]
aws_api_gateway_method.proxy_root: Refreshing state... [id=agm-fevd7uu5hi-z90dlqfi9c-ANY]
aws_api_gateway_method.proxy: Refreshing state... [id=agm-fevd7uu5hi-f9mbr9-ANY]
aws_s3_bucket_object.blue: Refreshing state... [id=feature.zip]
aws_lambda_function.example: Refreshing state... [id=training99_feature_flag]
aws_api_gateway_integration.lambda: Refreshing state... [id=agi-fevd7uu5hi-f9mbr9-ANY]
aws_api_gateway_integration.lambda_root: Refreshing state... [id=agi-fevd7uu5hi-z90dlqfi9c-ANY]
aws_api_gateway_deployment.example: Refreshing state... [id=qq17q5]
aws_lambda_permission.apigw: Refreshing state... [id=AllowAPIGatewayInvoke]
aws_lambda_function.example: Modifying... [id=training99_feature_flag]
aws_lambda_function.example: Modifications complete after 0s [id=training99_feature_flag]

Apply complete! Resources: 0 added, 1 changed, 0 destroyed.

Outputs:

base_url = https://fevd7uu5hi.execute-api.us-east-2.amazonaws.com/test
```

9. Again, go to the url at the bottom of the output. You should see something like this

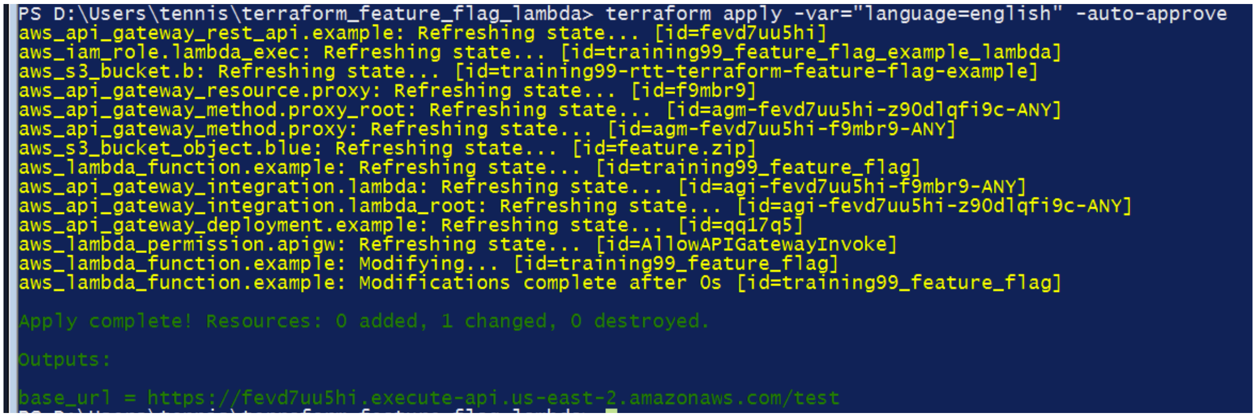


Bonjour

10. Thus, using the "language" feature toggle, we can permanently change the output of a function.

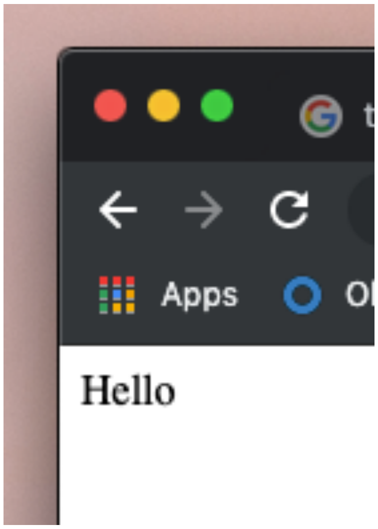
11. Now we want to change it back. Enter the command

```
terraform apply -var="language=english" -auto-approve
```



```
PS D:\Users\tennis\terraform_feature_flag_lambda> terraform apply -var="language=english" -auto-approve
aws_api_gateway_rest_api.example: Refreshing state... [id=fevd7uu5hi]
aws_iam_role.lambda_exec: Refreshing state... [id=training99_feature_flag_example_lambda]
aws_s3_bucket.b: Refreshing state... [id=training99-rtt-terraform-feature-flag-example]
aws_api_gateway_resource.proxy: Refreshing state... [id=f9mbr9]
aws_api_gateway_method.proxy_root: Refreshing state... [id=agm-fevd7uu5hi-z90dlqfi9c-ANY]
aws_api_gateway_method.proxy: Refreshing state... [id=agm-fevd7uu5hi-f9mbr9-ANY]
aws_s3_bucket_object.blue: Refreshing state... [id=feature.zip]
aws_lambda_function.example: Refreshing state... [id=training99_feature_flag]
aws_api_gateway_integration.lambda: Refreshing state... [id=agi-fevd7uu5hi-f9mbr9-ANY]
aws_api_gateway_integration.lambda_root: Refreshing state... [id=agi-fevd7uu5hi-z90dlqfi9c-ANY]
aws_api_gateway_deployment.example: Refreshing state... [id=qq17q5]
aws_lambda_permission.apigw: Refreshing state... [id=AllowAPIGatewayInvoke]
aws_lambda_function.example: Modifying... [id=training99_feature_flag]
aws_lambda_function.example: Modifications complete after 0s [id=training99_feature_flag]

Apply complete! Resources: 0 added, 1 changed, 0 destroyed.

Outputs:

base_url = https://fevd7uu5hi.execute-api.us-east-2.amazonaws.com/test
```

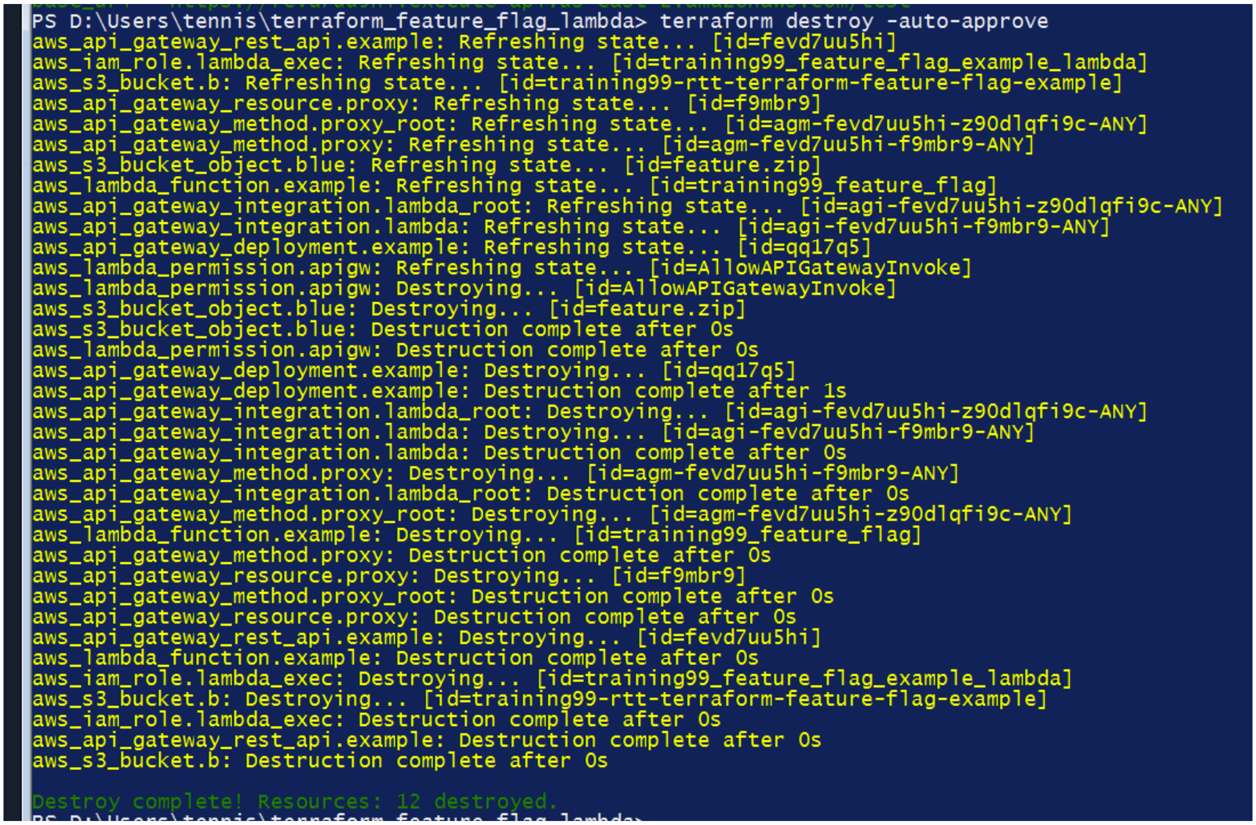
12. Now go back to your browser and refresh the screen, you should see this



Hello

13. Now we can take down the whole system. Enter the command

```
terraform destroy -auto-approve
```



```
PS D:\Users\tennis\terraform_feature_flag_lambda> terraform destroy -auto-approve
aws_api_gateway_rest_api.example: Refreshing state... [id=fevd7uu5hi]
aws_iam_role.lambda_exec: Refreshing state... [id=training99_feature_flag_example_lambda]
aws_s3_bucket.b: Refreshing state... [id=training99-rtt-terraform-feature-flag-example]
aws_api_gateway_resource.proxy: Refreshing state... [id=f9mbr9]
aws_api_gateway_method.proxy_root: Refreshing state... [id=agm-fevd7uu5hi-z90dlqfi9c-ANY]
aws_api_gateway_method.proxy: Refreshing state... [id=agm-fevd7uu5hi-f9mbr9-ANY]
aws_s3_bucket_object.blue: Refreshing state... [id=feature.zip]
aws_lambda_function.example: Refreshing state... [id=training99_feature_flag]
aws_api_gateway_integration.lambda_root: Refreshing state... [id=agi-fevd7uu5hi-z90dlqfi9c-ANY]
aws_api_gateway_integration.lambda: Refreshing state... [id=agi-fevd7uu5hi-f9mbr9-ANY]
aws_api_gateway_deployment.example: Refreshing state... [id=qq17q5]
aws_lambda_permission.apigw: Refreshing state... [id=AllowAPIGatewayInvoke]
aws_lambda_permission.apigw: Destroying... [id=AllowAPIGatewayInvoke]
aws_s3_bucket_object.blue: Destroying... [id=feature.zip]
aws_s3_bucket_object.blue: Destruction complete after 0s
aws_lambda_permission.apigw: Destruction complete after 0s
aws_api_gateway_deployment.example: Destroying... [id=qq17q5]
aws_api_gateway_deployment.example: Destruction complete after 1s
aws_api_gateway_integration.lambda_root: Destroying... [id=agi-fevd7uu5hi-z90dlqfi9c-ANY]
aws_api_gateway_integration.lambda: Destroying... [id=agi-fevd7uu5hi-f9mbr9-ANY]
aws_api_gateway_integration.lambda: Destruction complete after 0s
aws_api_gateway_method.proxy: Destroying... [id=agm-fevd7uu5hi-f9mbr9-ANY]
aws_api_gateway_integration.lambda_root: Destruction complete after 0s
aws_api_gateway_method.proxy_root: Destroying... [id=agm-fevd7uu5hi-z90dlqfi9c-ANY]
aws_lambda_function.example: Destroying... [id=training99_feature_flag]
aws_api_gateway_method.proxy: Destruction complete after 0s
aws_api_gateway_resource.proxy: Destroying... [id=f9mbr9]
aws_api_gateway_method.proxy_root: Destruction complete after 0s
aws_api_gateway_resource.proxy: Destruction complete after 0s
aws_api_gateway_rest_api.example: Destroying... [id=fevd7uu5hi]
aws_lambda_function.example: Destruction complete after 0s
aws_iam_role.lambda_exec: Destroying... [id=training99_feature_flag_example_lambda]
aws_s3_bucket.b: Destroying... [id=training99-rtt-terraform-feature-flag-example]
aws_iam_role.lambda_exec: Destruction complete after 0s
aws_api_gateway_rest_api.example: Destruction complete after 0s
aws_s3_bucket.b: Destruction complete after 0s

Destroy complete! Resources: 12 destroyed.
PS D:\Users\tennis\terraform_feature_flag_lambda>
```