# Serverless Framework Deploy Lambda from YAML

## CONCEPTS

sls is a short alias for serverless. Every command reads serverless.yml in the current folder and defaults to stage dev.

**serverless.yml** functions + events
↓ package
**.serverless/** zips + CloudFormation
↓ deploy
**CloudFormation stack** Lambda, API, IAM
↓ invoke / logs
**Running service** per stage and region

A stage (--stage) is a separate full copy of the stack, e.g. dev, staging, prod. Region comes from --region or provider.region.

## ACCOUNT

**serverless login**
Sign in and pick the default org

**serverless** *<command>* **--help**
Show help for one command

## DEPLOY

**serverless deploy**
Deploy the whole service

**serverless deploy --stage** *<stage>* **--region** *<region>*
Deploy to another stage or region

**serverless deploy --verbose**
Show stack events and outputs

**serverless deploy --aws-profile** *<name>*
Deploy with a named AWS profile

**serverless deploy --package** *<dir>*
Deploy a pre-built package

**serverless deploy function -f** *<fn>*
Upload one function's code only

**serverless deploy function -f** *<fn>* **--update-config**
Push only Lambda config changes

**serverless deploy list**
List stored deployments

## PACKAGE

**serverless package**
Build artifacts without deploying

**serverless package --package** *<dir>*
Write the package to a folder

## INVOKE

**serverless invoke -f** *<fn>*
Call the deployed function

**serverless invoke -f** *<fn>* **--log**
Also print the invocation log

**serverless invoke -f** *<fn>* **-d** '*<json>*'
Pass an inline event

**serverless invoke -f** *<fn>* **-p** *<file>*
Pass an event from a JSON file

**serverless invoke -f** *<fn>* **-t Event**
Fire asynchronously

## LOCAL

**serverless invoke local -f** *<fn>*
Run a function on your machine

**serverless invoke local -f** *<fn>* **-p** *<file>*
Use an event file

**serverless invoke local -f** *<fn>* **-d** '*<json>*'
Use an inline event

**serverless invoke local -f** *<fn>* **-e** *<NAME>*=*<value>*
Set an environment variable

**serverless invoke local -f** *<fn>* **--docker**
Run in the declared runtime

**serverless dev**
Run deployed functions on local code

## LOGS

**serverless logs -f** *<fn>*
Show the last 10 minutes of logs

**serverless logs -f** *<fn>* **-t**
Tail logs live

**serverless logs -f** *<fn>* **--startTime** *<time>*
Logs since 30m, 2h, 3d or a date

## LOGS (cont.)

**serverless logs -f** *<fn>* **--filter** *<text>*
Only lines containing a pattern

**serverless metrics**
Service-wide metrics, last 24 hours

**serverless metrics -f** *<fn>* **--startTime** *<t>*
Metrics for one function

## INSPECT

**serverless info**
Show endpoints and functions

**serverless info --verbose**
Include all stack outputs

**serverless print**
Show config with variables resolved

**serverless print --path** *<a.b>*
Print one sub-value

**serverless print --format json**
Output as yaml, json or text

## ROLLBACK AND REMOVE

**serverless rollback**
List deployments to roll back to

**serverless rollback -t** *<timestamp>*
Return to a past deployment

**serverless remove**
Delete the service from AWS

**serverless remove --stage** *<stage>* **--region** *<region>*
Delete one stage

## CONFIG

**Minimal serverless.yml**

```
service: my-api

provider:
  name: aws
  runtime: nodejs20.x
  region: eu-central-1

functions:
  hello:
    handler: handler.hello
    events:
      - httpApi: GET /hello
```