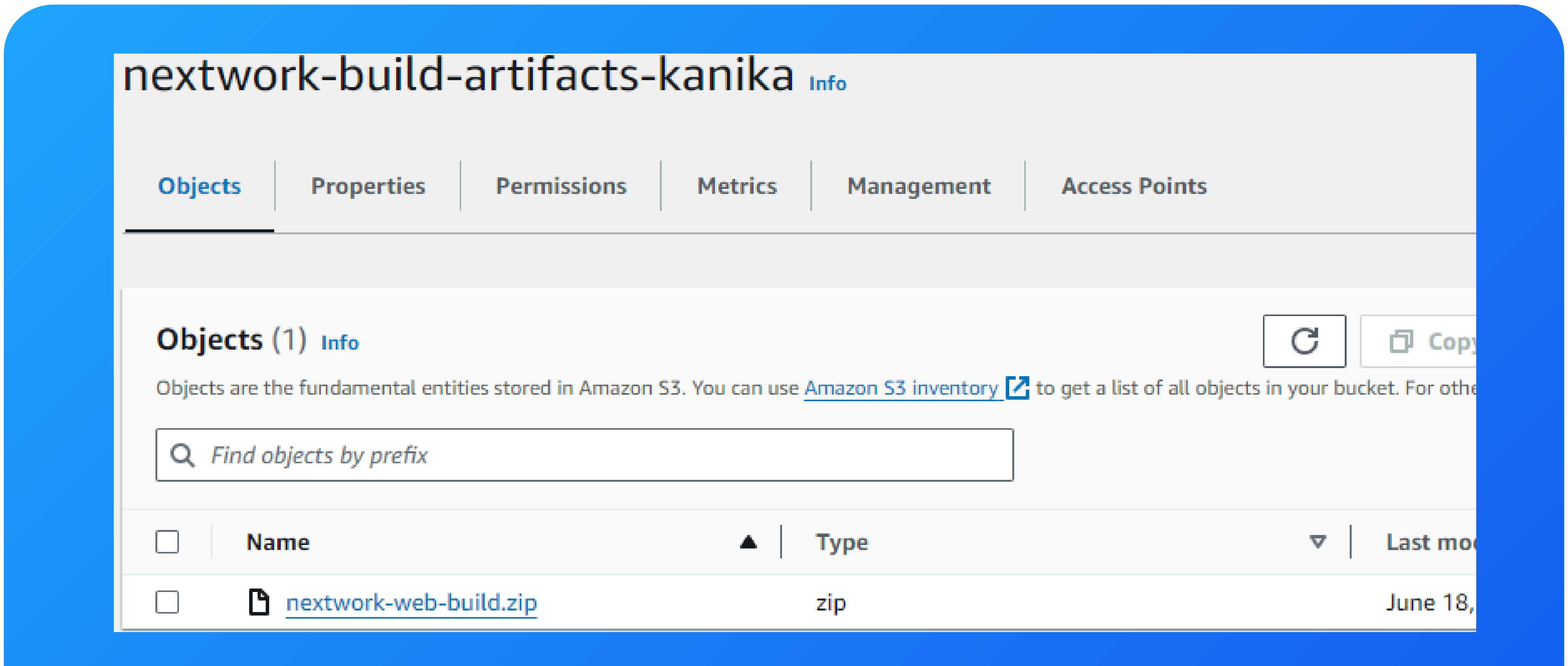
NextWork.org

Package an App with CodeBuild

Kanika Mathur
github.com/KanikaGenesis



nextwork-build-artifacts-kanika Info

Objects (1) Info

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For othe

Find objects by prefix

□	Name ▲	Type ▽	Last mo
□	nextwork-web-build.zip	zip	June 18,

Introducing AWS CodeBuild!

What it does & how it’s useful

AWS CodeBuild is a fully managed continuous integration service that compiles source code, runs tests, and produces software packages.
Developers and teams use AWS CodeBuild because it scales automatically, integrates with other AWS services, and speeds up the build process.

How I’m using it in today’s project

I’m using AWS CodeBuild in this project to eliminate the need to provision, manage, and scale my own build servers.

This project took me...

This project took me around 2 hours to complete.

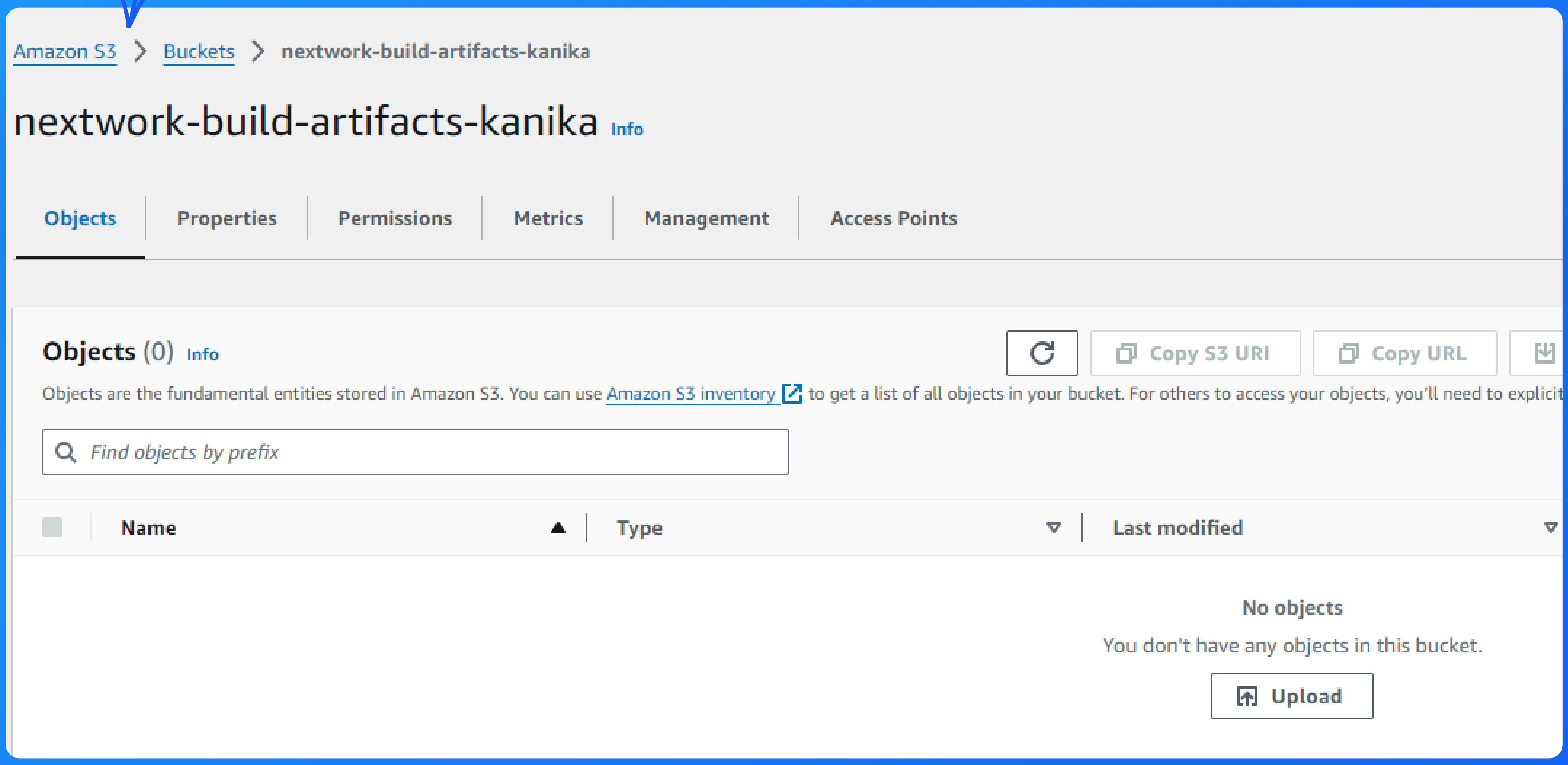
Set up an S3 bucket

- I started my project by creating an S3 bucket because this bucket will later catch an important artifact that gets created from the build process I am setting up with CodeBuild.

- The key artifact that this S3 bucket will capture is called a WAR (Web Application Resource) file.

- This file is important because it ensure that any server that will host my web app will have all of the resource/tools it needs to successfully run my application.

My S3 bucket!



Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicit

No objects

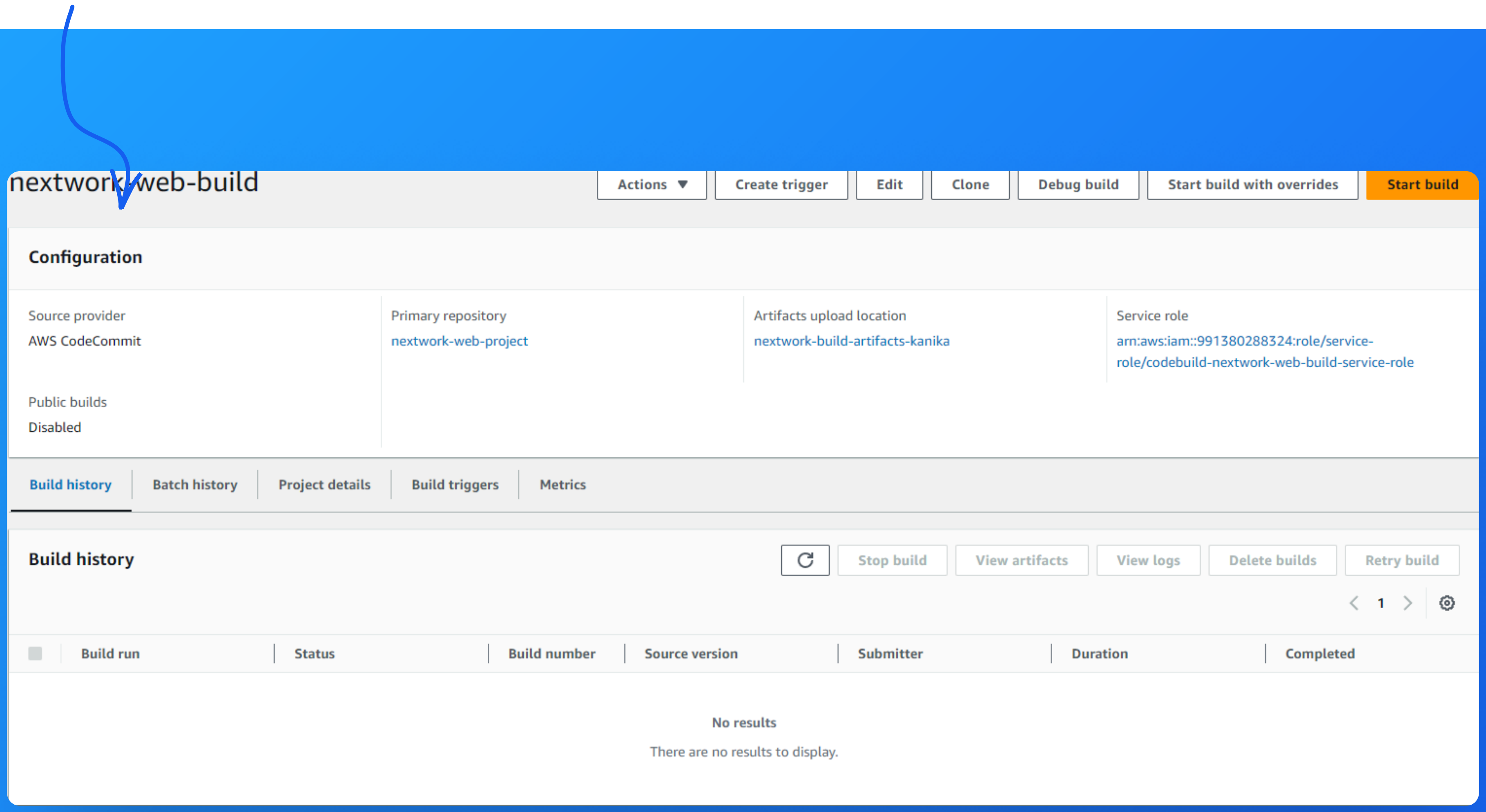
You don't have any objects in this bucket.

Set up a CodeBuild project

When creating a project in CodeBuild, there were 5 key configurations I set up:

1. **Source**, which means the source of the project's code/files. I selected AWS CodeCommit, as that is my cloud repository for my project's files.
2. **Environment**, which means the set of resources/configurations required to build my web app. I selected an EC2 instance running Amazon Linux 2 and Java Coretto 8 as the runtime.
3. **Buildspec**, which are the set of commands to run in my local computer's terminal during the build process. I selected "use a buildspec file."
4. **Artifacts**, which means where CodeBuild should store the build artifacts that get produced during the build process. I selected the S3 bucket I created.
5. **Logs**, which means whether I'd like to keep records of every single build process that CodeBuild does. I enabled CloudWatch Logs.

My completed project ready for the first build!



nextwork-web-build

Configuration

Source provider
AWS CodeCommit

Primary repository
nextwork-web-project

Artifacts upload location
nextwork-build-artifacts-kanika

Service role
arn:aws:iam::991380288324:role/service-role/codebuild-nextwork-web-build-service-role

Public builds
Disabled

Build history

□	Build run	Status	Build number	Source version	Submitter	Duration	Completed

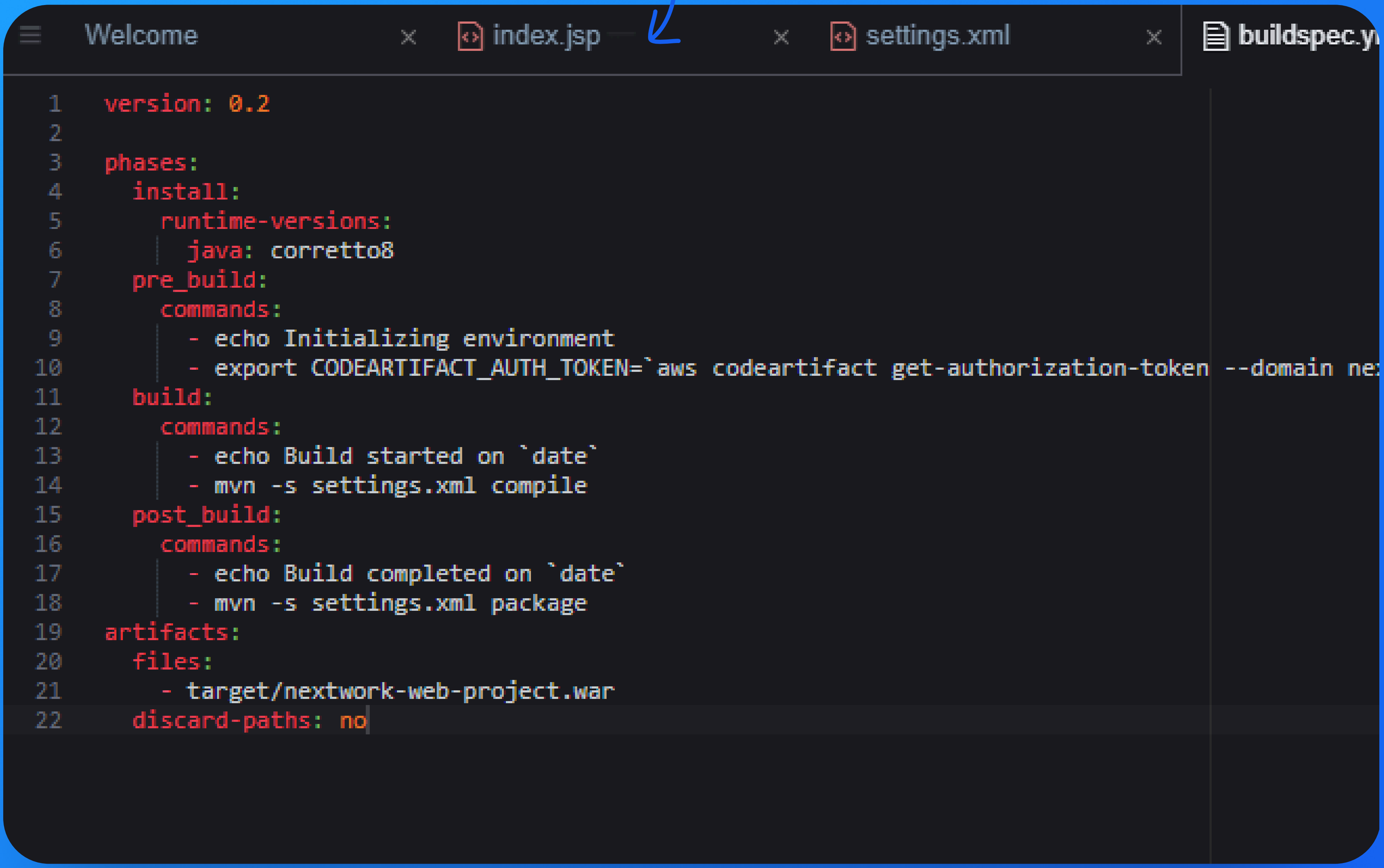
No results
There are no results to display.

Create a buildspec.yml file

- I created a buildspec.yml file at the root of our code repository.
- This file contains four phases that tells our build environment what commands to run. These four phases are:

1. install -i.e. install these dependencies before you start compiling (in this project, it’s Java Coretto 8).
2. pre_build - i.e. before Maven starts building retrieve an access token to our CodeArtifact repository so we can fetch dependencies later.
3. build - i.e. to build our web app project, run these commands.
4. post_build - i.e. after building our web app project, package up the machine code using settings in the setting.xml file.

A peek into my buildspec.yml



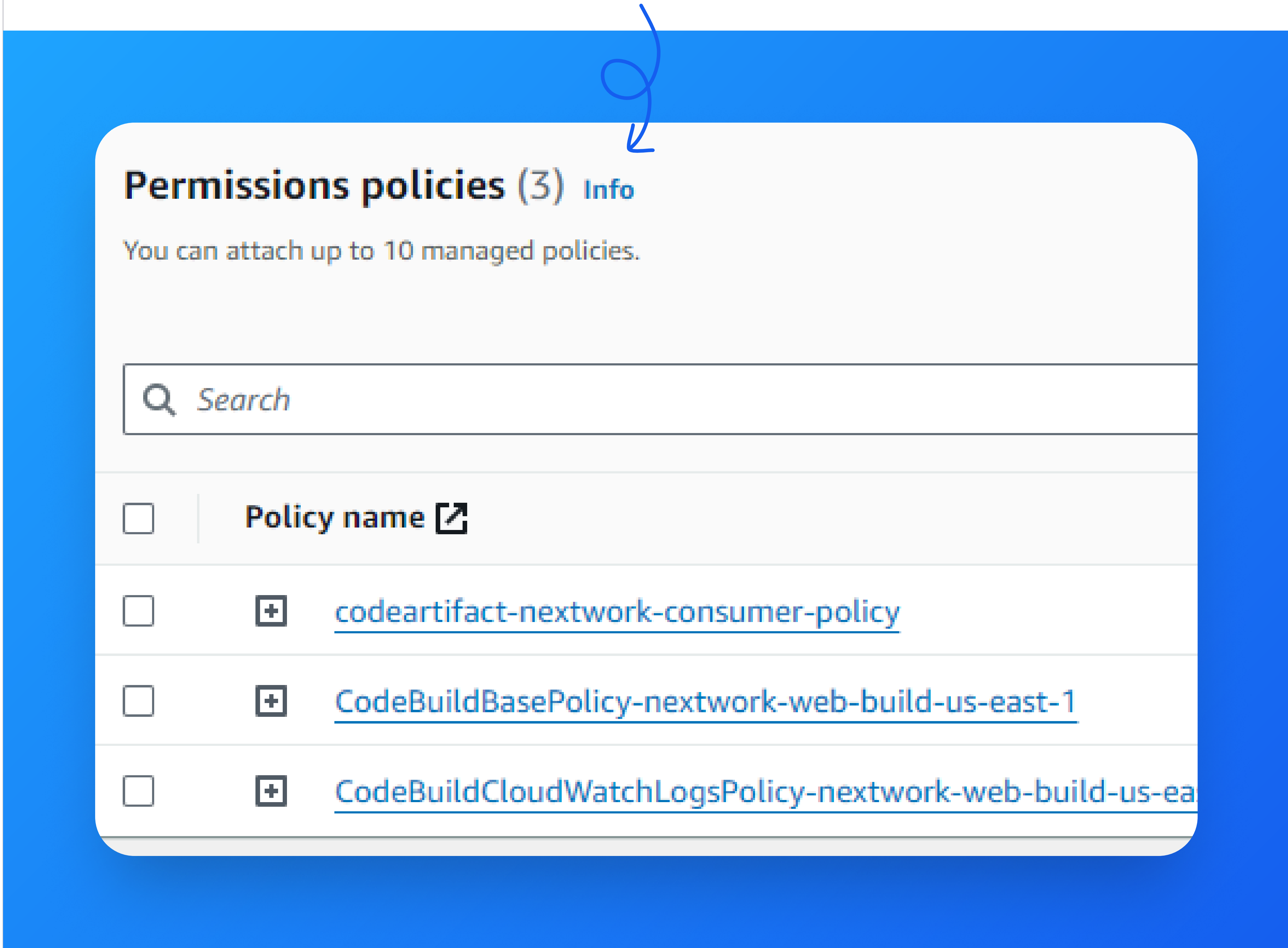
```yaml
version: 0.2

phases:
  install:
    runtime-versions:
      java: corretto8
  pre_build:
    commands:
      - echo Initializing environment
      - export CODEARTIFACT_AUTH_TOKEN=`aws codeartifact get-authorization-token --domain ne
  build:
    commands:
      - echo Build started on `date`
      - mvn -s settings.xml compile
  post_build:
    commands:
      - echo Build completed on `date`
      - mvn -s settings.xml package
artifacts:
  files:
    - target/nextwork-web-project.war
  discard-paths: no
```

Edit CodeBuild’s IAM role

- Before I start building my web app project (exciting!), I modified my CodeBuild project’s service role first. This role was first created when I set up my CodeBuild project (I checked the setting for create new service role).

- I attached a new policy called **codeartifact-nextwork-consumer-policy** to my CodeBuild project’s IAM role. This means my CodeBuild project now has access to the packages/dependencies that it will later compile.

Updating permission policies for my CodeBuild project’s IAM role.



Permissions policies (3) Info

You can attach up to 10 managed policies.

□		Policy name
□	⊞	codeartifact-nextwork-consumer-policy
□	⊞	CodeBuildBasePolicy-nextwork-web-build-us-east-1
□	⊞	CodeBuildCloudWatchLogsPolicy-nextwork-web-build-us-ea

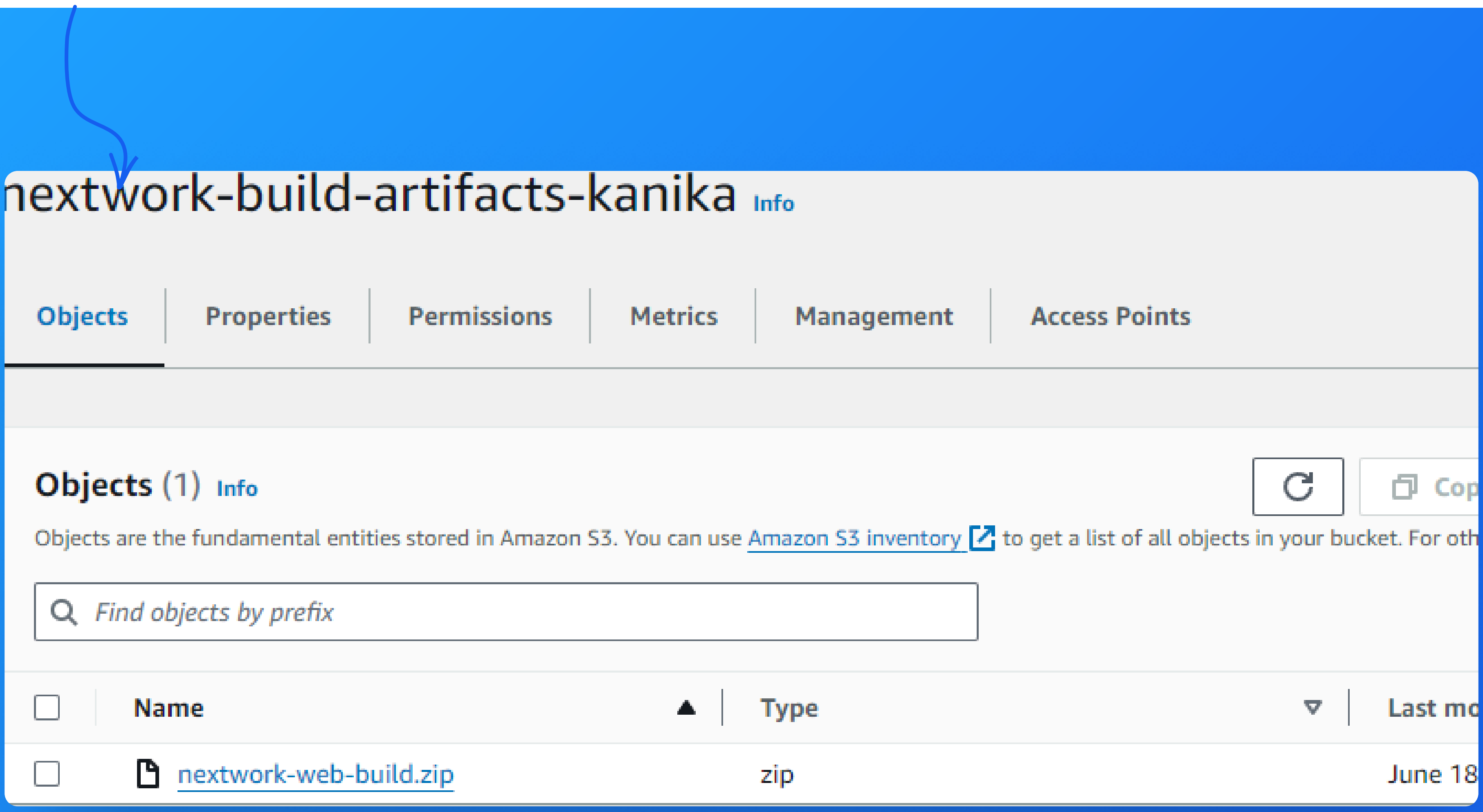
My first project build 💪

- To build my project, all I had to do was select the Start build button in my CodeBuild console - super simple!

- The build process in CodeBuild took about 5 minutes.

- Once the build is complete, I checked our S3 bucket was set up at the start of this project to catch the build artifacts that get produced from our build process. I saw the zip file i.e. the WAR file (a bundle up package of all files/resources a server will need to run my web app), which verified that the build was completed successfully.

My completed project ready for the first build!



nextwork-build-artifacts-kanika Info

Objects (1) Info

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For oth

□	Name	Type	Last mo
□	nextwork-web-build.zip	zip	June 18

My key learnings

1. The build process, or ‘building’, means converting source code into executable software. This involves compiling code, running tests, and packaging the software, ensuring that all dependencies are included and the code is ready for deployment.

2. The buildspec.yml file is a configuration file for AWS CodeBuild. It defines the build commands and settings used in the build process, including environment variables, build phases, and artifacts. This file ensures consistency and automation in building your project.

3. Even though CodeBuild creates a new service role for my build environment, I still have to modify the role’s permission policies because the default permissions may not cover all the specific actions and resources my project requires.

4. One thing I didn’t expect was how straightforward and efficient the integration of AWS CodeBuild with other AWS services would be, significantly streamlining our CI/CD pipeline and reducing manual overhead.

NEXTWORK

Everyone should be in a job they love. yes!

Check out community.nextwork.org for more free projects

Ask me about it

linkedin.com/in/kanika-mathur-083080121/