**Code commit and Code build and S3 Artifacts**

#### **1) Setting Up CodeCommit**

1. First, navigate to the **IAM** service, go to **Users**, and select **Security Credentials**.
2. Under **HTTPS Git credentials for AWS CodeCommit**, generate a password, download it locally, and grant all necessary CodeCommit permissions to the user, such as **AWSCodeCommitFullAccess** and **AWSCodeCommitPowerUser**.
3. If needed, create a custom policy under **Policies** and attach it to the user for further access management.
4. Next, go to **CodeCommit**, create a new repository by providing a name and an optional description, then select **Create**.
5. Copy the HTTP clone URL of the repository for local cloning.
6. After setting up the repository locally, add an index.html file to the project.

Push this file to CodeCommit using the following commands:  
bash  
Copy code  
git add .

git commit -m "added HTML file"

git push origin master

#### **2 ) Setting Up CodeBuild**

1. Go to the **CodeBuild** service and select **Create Build Project**.
2. Assign a project name.
3. Under **Source**, select **Source Provider** as **CodeCommit**, and select your repository (e.g., demoapp).
   * **Branch**: master (or your desired branch)
4. Under **Environment**, select:
   * **Environment Image**: Managed Image
   * **Operating System**: Ubuntu
   * **Runtime**: Standard
   * **Image**: aws/codebuild/standard:6.0
   * **Environment Type**: Linux
   * **Service Role**: New service role (provide a name)
5. Under **Build Specifications**, choose **Buildspec File**.

Within the project directory, create a buildspec.yml file with the following content:  
yaml  
Copy code  
version: 0.2

phases:

install:

commands:

- echo Installing NGINX

- sudo apt-get update

- sudo apt-get install nginx -y

build:

commands:

- echo Build started on `date`

- cp index.html /var/www/html/

post\_build:

commands:

- echo Configuring NGINX

- echo Creating ZIP file

- zip -r buildspec-output.zip ./\*

artifacts:

files:

- buildspec-output.zip

discard-paths: yes

base-directory: .

name: buildspec-output.zip

override-artifact-name: true

1. Commit and push the buildspec.yml file to CodeCommit.
2. Click on **Create Build** to initialize the build.

#### **3) Storing Artifacts in S3**

1. Open your build project, select **Edit**, and go to the **Artifacts** section.
2. Set **Type** to **S3**.
3. Go to **S3** and create a bucket. Assign a bucket name (e.g., your-bucket-name).
4. Specify the artifact name within **Name** as a folder (e.g., buildspecs-folder).
5. Update the artifact configuration.
6. Once the build completes, the artifact (ZIP file) will be available in the S3 bucket.

Download and unzip the file on your local machine with:  
bash  
Copy code  
unzip buildspec-output.zip

The unzipped content will contain the index.html, appspec.yml, buildspec.yml, and script files from your project.