

ALGO

- **Create an Amazon Account:** Visit the AWS website to register for an account. The services and resources required to launch your Spring Boot application will be made available by this account.
- **Configure AWS CLI:** On your local computer, install the AWS Command Line Interface (CLI). Use the `aws configure` command to setup it using your AWS credentials. This enables command line interaction with AWS services.
- **Start an EC2 Instance:** To start an EC2 instance, use the AWS Management Console. Specify the key pair, security groups, and instance type. Your Spring Boot application will execute on the virtual server provided by the EC2 instance.
- **Install Java on EC2:** Open SSH and connect to the freshly launched EC2 instance. To guarantee that the Java runtime environment is accessible for executing Spring Boot apps, install Java on the instance.
- **GitHub Repository:** For your Spring Boot application, create a GitHub repository. Keep track of your source code in the repository and prevent sensitive files from being committed by utilizing a `.gitignore` file to block configuration files and build artifacts.
- **create and Package:** To create your Spring Boot application, use either Gradle or Maven. During this process, dependencies are resolved, your source code is compiled, and the application is packaged into an executable JAR file.
- **Move JAR to EC2:** Move the JAR file to the EC2 instance from your local computer. Tools such as SCP (Secure Copy Protocol) can be used for this, or the JAR can be downloaded onto an EC2 instance by uploading it to an S3 bucket.
- **Run Application on EC2:** Use the `java -jar` command to run the JAR file on the EC2 instance. This starts your Spring Boot application and makes it available from the instance on the given port.
- **Configure Security Groups:** Set your EC2 instance's security groups to permit incoming traffic on the port that your Spring Boot application uses. This guarantees that requests from outside sources can go to your program.
- **Access Application:** To access your Spring Boot application, go to your EC2 instance's public IP address or DNS, then add the port that your application is executing on. You may now test and interact with your deployed application thanks to this.