

# Top Gear Cloud Training

## Lesson 1

---

### Learning Objective

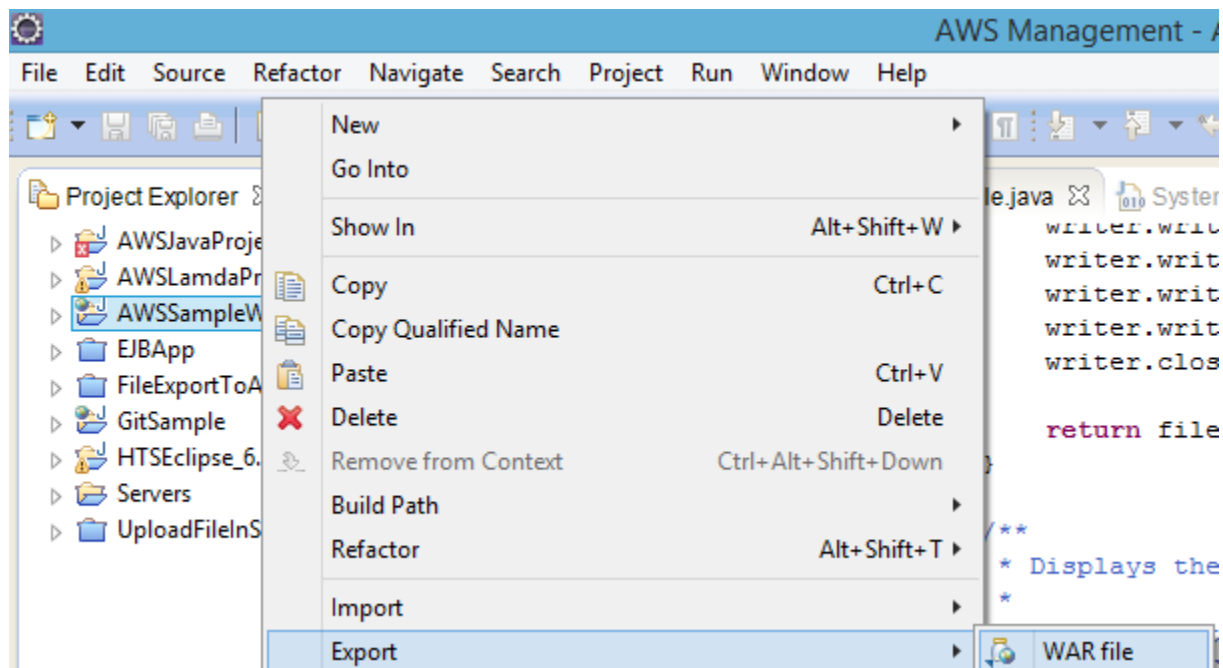
How to create cloud infrastructure and deploy web application to cloud. You will see your application running on AWS cloud platform.

### Prerequisite for trainer:

Create AWS account. Create an AWS group in IAM. Provide required permission to the group. Create user for each employee using Wipro ID of trainee. Assign them to the created group. Hand over User Id, password and account id to trainees. Create a Key pair for EC2 access. Provide Key pair name to trainee.

### Prerequisite for trainee:

In your local eclipse system build a web application and test it in Tomcat 8 server in your local machine on Java 8. User need to have AWS account id, User id and Password. The web application should not be dependent of database or perform file access. Create war file from export option of eclipse of your web project and save it to your local drive.

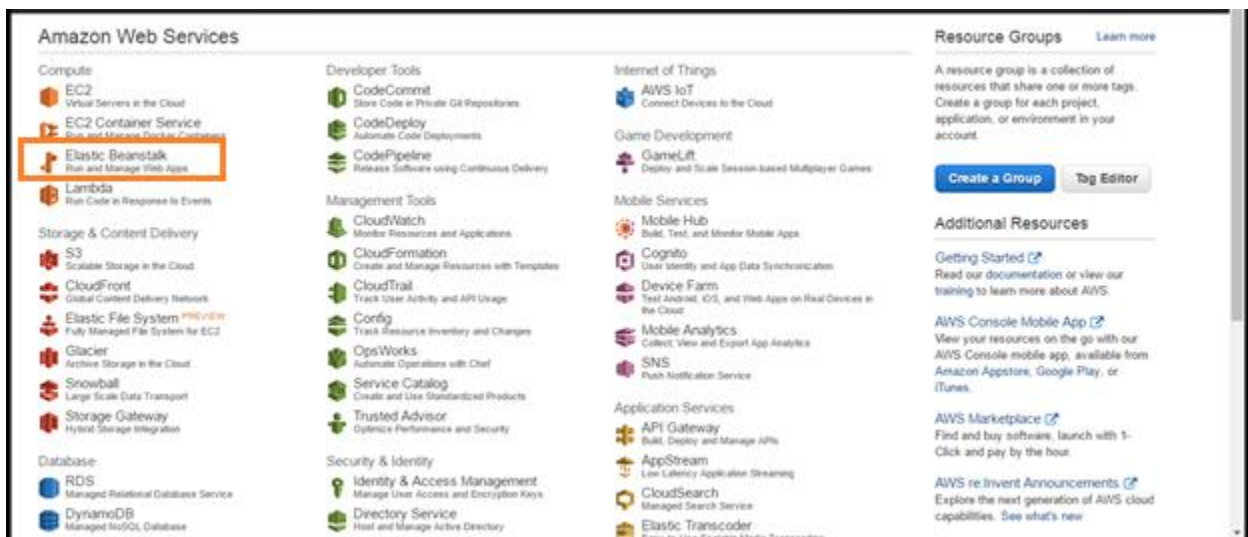


## Steps to follow:

Login to AWS environment:

1. Login to AWS environment. Paste the following link in your favorite browser.  
<https://940679525002.signin.aws.amazon.com/console>
2. Key in your user ID and password and login to the system. You should be able to see a screen like the following with all the AWS service listed. Click on Elastic Beanstalk.

<b>Username</b>	topgear-aws-master
<b>Password</b>	wipro_123



3. Click on Create New Application on the next page.
4. In application name put your WiproID and in description put your name. Click Next:

**Elastic Beanstalk** My First Elastic Beanstalk Application Create New Application

**Application Information**

New Environment

To create a new application, enter the details of your application.

Application name:  Must be less than 100 characters and cannot contain a /

Description:  Optional

Cancel Next

5. In New Environment page click Create Web Server.

Elastic Beanstalk My First Elastic Beanstalk Application Create New Environment

Application Info  
New Environment

### New Environment

AWS Elastic Beanstalk has two types of environment tiers to support different types of web applications. Web servers are standard applications that listen for and then process HTTP requests, typically over port 80. Workers are specialized applications that have a background processing task that listens for messages on an Amazon SQS queue. Worker applications post those messages to your application by using HTTP.

**Web Server Environment**

Provides resources for an AWS Elastic Beanstalk web server in either a single instance or load-balancing, auto scaling environment. [Learn more.](#)

**Worker Environment\***

Provides resources for an AWS Elastic Beanstalk worker application in either a single instance or load-balancing, auto scaling environment. [Learn more.](#)

\* Worker environments require additional permissions to access other AWS services. [Learn more.](#)

Cancel Done

- In Environment Type page define Environment type. In Predefined configuration put Tomcat. In environment type put single instance. Click Next:

Elastic Beanstalk Create New Environment

Application Info  
New Environment  
Environment Type

### Environment Type

Choose the platform and type of environment to launch.

Predefined configuration: Tomcat Looking for a different platform? [Let us know.](#)

AWS Elastic Beanstalk will create an environment running Tomcat 8 Java 8 on 64bit Amazon Linux 2016.03 v2.1.1. [Change platform version.](#)

Environment type: Single instance [Learn more](#)

Cancel Previous Next

- In Application Version for source choose upload your own.

AWS Services Edit Satyajit @ 9406-7952-5002 N. Virginia Support

Elastic Beanstalk My First Elastic Beanstalk Application Create New Environment

Application Info  
New Environment  
Environment Type  
Application Version

### Application Version

Select a source for your application version.

Source: ☐ Sample application ☒ Upload your own ([Learn more](#))

No file chosen

☐ S3 URL  (e.g. <https://s3.amazonaws.com/s3Bucket/s3Key>)

Cancel Previous Next

- In Application Version page, click on Choose File. Upload the .war file of web project created from export option in eclipse. Click next. Wait for the war file getting uploaded.

Elastic Beanstalk My First Elastic Beanstalk Application [Create New Environment](#)

Application Info  
New Environment  
Environment Type  
**Application Version**  
Environment Info  
Additional Resources  
Configuration Details  
Environment Tags  
Permissions  
Review Information

### Application Version

Select a source for your application version.

Source: ☐ Sample application ☒ Upload your own ([Learn more](#)) ☐ S3 URL

[Choose File](#) AWSSampleWeb1.war

(e.g. <https://s3.amazonaws.com/s3Bucket/s3Key>)

[Cancel](#) [Previous](#) [Next](#)

9. In Environment Information page, click on check availability to ensure the URL or your application is unique. Add description. Click Next button.

Elastic Beanstalk My First Elastic Beanstalk Application [Create New Environment](#)

New Environment  
Environment Type  
Application Version  
**Environment Info**  
Additional Resources  
Configuration Details  
Environment Tags  
Permissions  
Review Information

### Environment Information

Enter your environment information.

Environment name:

Environment URL:

[Check availability](#)

Description:  Optional: 200 character maximum

[Cancel](#) [Previous](#) [Next](#)

10. In Additional Resources page, don't select anything on next page. Click Next.

Elastic Beanstalk My First Elastic Beanstalk Application [Create New Environment](#)

New Environment  
Environment Type  
Application Version  
Environment Info  
**Additional Resources**  
Configuration Details  
Environment Tags  
Permissions  
Review Information

### Additional Resources

Select additional resources for this environment.

☐ Create an RDS DB Instance with this environment [Learn more](#)

☐ Create this environment inside a VPC [Learn more](#)

[Cancel](#) [Previous](#) [Next](#)

11. In Configuration Details page, for instance select 't1.micro'. Select key pair name provided to you. Put your email address. In Health reporting system type put 'Basic'. For root volume type select magnetic. Type 8 in root volume size. Click Next.

Environment Info      Modify the following settings or click Next to accept the default configuration. [Learn more.](#)

Additional Resources

**Configuration Details**

Environment Tags

Permissions

Review Information

Instance type:    
 Determines the processing power of the servers in your environment.

EC2 key pair:  [Refresh](#)   
 Optional: Enables remote login to your instances.

Email address:    
 Optional: Get notified about any major changes to your environment.

**Health Reporting**

System type:    
 Determines the health reporting type.

**Root Volume (Boot Device)**

Root volume type:    
 Determines the type of storage volume to attach to instances.

Root volume size: ☒ Enables you to specify the size of the root volume.   
     
 Number of gibibytes of the root volume attached to each instance. Must be between 10 and 16384 for Provisioned IOPS (SSD) and General Purpose (SSD) root volumes and between 8 and 1024 for other root volumes.

[Cancel](#) [Previous](#) [Next](#)

12. In Environment Tags page, put UserId and your WiproID for key and value. Click Next

Elastic Beanstalk      My First Elastic Beanstalk Application      [Create New Environment](#)

New Environment

Environment Type

Application Version

Environment Info

Additional Resources

Configuration Details

**Environment Tags**

Permissions

Review Information

**Environment Tags**

You can specify tags (key-value pairs) for your Environment. You can add up to 7 unique key-value pairs for each Environment.

	Key (128 characters maximum)	Value (256 characters maximum)
1.	<input type="text" value="UserId"/>	<input type="text" value="SA345476"/>
2.	<input type="text"/>	<input type="text"/>

6 remaining

[Cancel](#) [Previous](#) [Next](#)

13. In Permission page, don't change anything. Click Next

Elastic Beanstalk      My First Elastic Beanstalk Application      [Create New Environment](#)

New Environment

Environment Type

Application Version

Environment Info

Additional Resources

Configuration Details

Environment Tags

**Permissions**

Review Information

**Permissions**

Select an instance profile and service role for your AWS Elastic Beanstalk environment.

An instance profile is an IAM role configured for use with EC2 instances. The instances in your Elastic Beanstalk use the credentials provided by the instance profile to communicate with AWS.

A service role allows the Elastic Beanstalk service to monitor environment resources on your behalf. See [Roles](#) and [Instance Profiles](#) in the Elastic Beanstalk developer guide for details.

Instance profile:

Service role:

[Cancel](#) [Previous](#) [Next](#)

14. In Review page, review details and click on Launch. Wait for the environment to be created. You can see all the task of creating server, installing software etc. are being done by AWS.

i

**Info**  
 Elastic Beanstalk is now creating your environment. When it has finished it will be running Sample Application.

×

[All Applications](#) > [SA345476](#) > [sa345476-env](#) ( Environment ID: e-rp28uxup7, URL: sa345476-env-us-east-1.elasticbeanstalk.com )

Actions ▾

Dashboard

Configuration

Logs

Health

Monitoring

Alarms

Managed Updates NEW

Events

Tags

Overview

Refresh ↻

✓

**Health**  
Green

Causes

Running Version

Sample Application

Upload and Deploy

Configuration

64bit Amazon Linux 2016.03  
v2.1.1 running Tomcat 8 Java 8

Change

Recent Events

Show All

Time	Type	Details
2016-06-01 15:55:51 UTC+0530	INFO	Successfully rebuilt environment: sa345476-env
2016-06-01 15:54:53 UTC+0530	INFO	Environment health has been set to GREEN

15. You can see the url for accessing the application starting with you Wiproid. Click on that URL to see your application running on AWS environment.
16. To modify the application you have deployed click on Upload and Deploy button. Update you application in local eclipse. Export new war file. Upload it. Put new version name. Click Deploy. Wait for it to get deployed.

Upload and Deploy

×

i
 To deploy a previous version, go to the [Application Versions page](#).

Upload application: Choose File AWSSampleWeb2.war

Version label: AWSSampleWeb2

Cancel
Deploy

17. Access the URL again. You can see your changes in the application getting reflected.
18. Delete the Application after you have tested your changes
19. Congratulations!! You have successfully deployed, and tested and removed your application in AWS platform.