## Instructions for setting up your web application on Amazon Web Services ( Steps include JAVA, Tomcat & MYSQL Setup)

## Set-up on Amazon AWS

* Activate the credit card account
* Create EC2 Instance (Free tier allowed)
* Check Public IP details which include
  + Server name: <<XX>>.compute.amazonaws.com
  + IP: AA.BB.CC.DD
* Download key pair to the local system. This pem file will be used for connecting to the Amazon application server instance

## CyberDuck setup (Cyberduck for Mac, Windows user can use winscp)

* To connect using Cyberduck

Public DNS: Servername/PublicIP (created above)

* User name: ec2-user
* Password: empty
* Pass the key of pem address

## Install JAVA on Amazon Application server instance

* Goto the folder where your pem key is located
* Steps for connecting

>>ssh -i "xx.pem" server-naeme

* Install the latest Java JDK 11 by running below on SSH client

>>sudo amazon-linux-extras install java-openjdk11

* Check whether java is installed by typing
  + >> java -version
* Set JAVA\_HOME by adding below lines in /etc/profile
  + export JAVA\_HOME=/usr/lib/jvm/java-11-openjdk-11.0.5.10-0.amzn2.x86\_64
  + export JRE\_HOME=/usr/lib/jvm/jre-11-openjdk-11.0.5.10-0.amzn2.x86\_64
  + export PATH=$PATH:$JAVA\_HOME/bin:$JRE\_HOME/bin
* Switch to root user before adding below lines in profile. To switch follow below lines
  + sudo su
  + passwd root
  + create a new password for root

## Tomcat setup

* Download latest version of tomcat:- <https://tomcat.apache.org/download-90.cgi>
* Copy the file through putty/Cyberduck
* Transfer the tar file to Linux EC2 server and unzip
* Use tax -zxvf <<>>.jar. If there is error in gzip use, then just use command tar xvf <<>>.tar
* Goto tomcat bin folder to start and shutdown the instance
* Try accessing on public IP at 8080 port
* If this does not work, check inbound traffic on the EC2 aws console. Add port 8080 to inbound traffic (Goto security groups option on AWS console)
* To change port goto server.xml at /home/ec2-user/apache-tomcat-9.0.27/conf
* To restart tomcat server automatically when the ec2 restarts, create following script in /etc/rc.d/init.d/
  + - !/bin/sh
    - # Tomcat init script for Linux.
    - # chkconfig: 234 20 80﻿﻿
    - # description: The Apache Tomcat servlet/JSP container.
    - JAVA\_HOME=/usr/lib/jvm/java-11-openjdk-11.0.5.10-0.amzn2.x86\_64
    - CATALINA\_HOME=/home/ec2-user/apache-tomcat-9.0.27
    - export JAVA\_HOME CATALINA\_HOME
    - exec $CATALINA\_HOME/bin/catalina.sh $\*
* Launch below commands for enabling tomcat restart automatically
  + **chmod 755 /etc/rc.d/init.d/tomcat**
  + **chkconfig --level 2345 tomcat on**
* Setting up tomcat users by adding below lines in tomcat-users.xml
  + <role rolename="manager-gui"/>
  + <role rolename="manager-script"/>
  + <role rolename="manager-jmx"/>
  + <role rolename="manager-status"/>
  + <role rolename="admin-gui"/>
  + <user username="admin" password="admin"
  + roles="manager-gui,manager-status,admin-gui"/>
  + <user username="tomcattools" password="tomcat"/>
* If access is still denied, try commenting value in context.xml at /home/ec2-user/apache-tomcat-9.0.27/webapps/manager/META-INF
* In the brew set-up, if you want to check the folder where tomcat is installed just use brew ls tomcat. For us, /usr/local/Cellar/tomcat/9.0.27/libexec/webapps

## MySQL Setup

* Goto Amazon console🡪Services🡪Storage🡪RDS
* Create mysql8.0+ database. Database name: <<>> Username: <<>> Password: <<>>
* Connecting to it
* mysql -h *<<servername>>* -P <<port>> -u *<<username>>* -p
* Cannot connect to DB Server. Try ping to *<<DB Server name>>*
* Try adding your public ip in the inbound and outbound traffic of the database instance. Specifying TCP, 3306 and MYSQL Type.
* Connect using Mysqlworkbench
  + Start server
* Creating users
  + create user '<<username>>'@'*server-name*' identified by 'schemaname';
  + GRANT ALL PRIVILEGES ON schemaname.\* TO 'username'@'*servername*';
* Connecting to mysql through terminal
  + mysql -h *servername -u <<username>> -p*