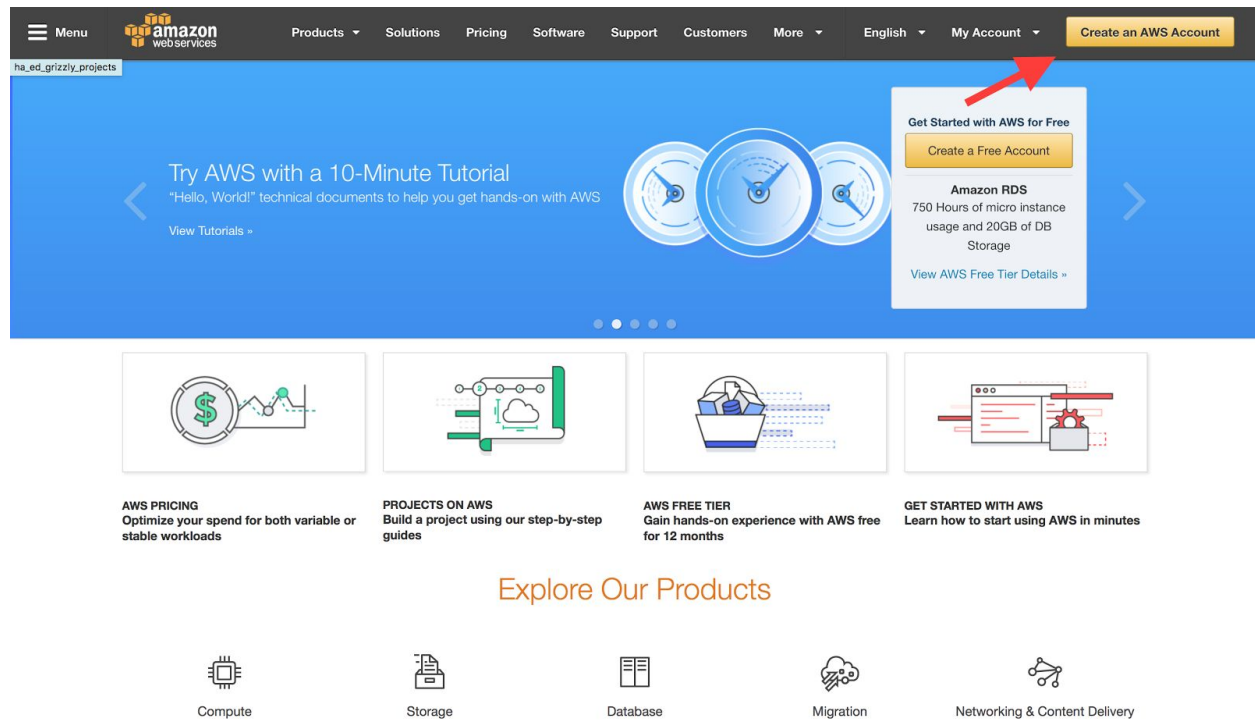


Creating an AWS account

1. Visit the following url: <https://aws.amazon.com/> which will bring you to this page

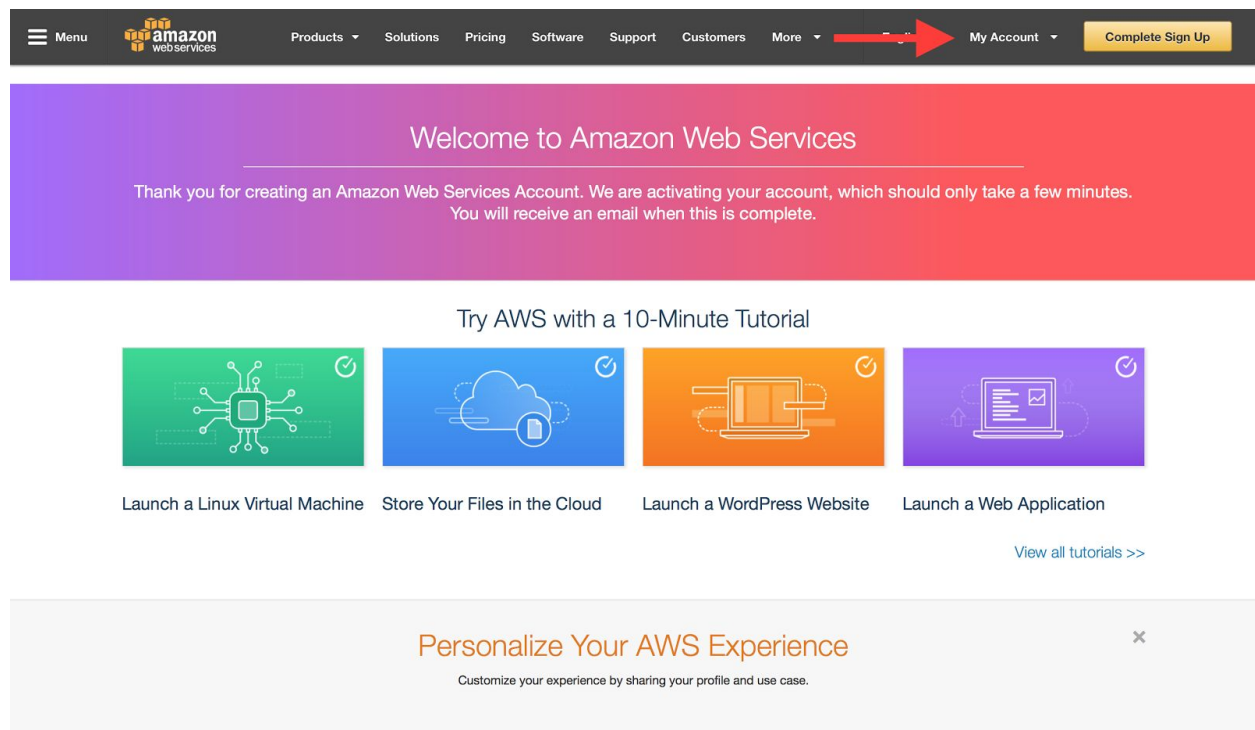


3. You will need to click the button 'Create an AWS Account' to begin actual account creation.

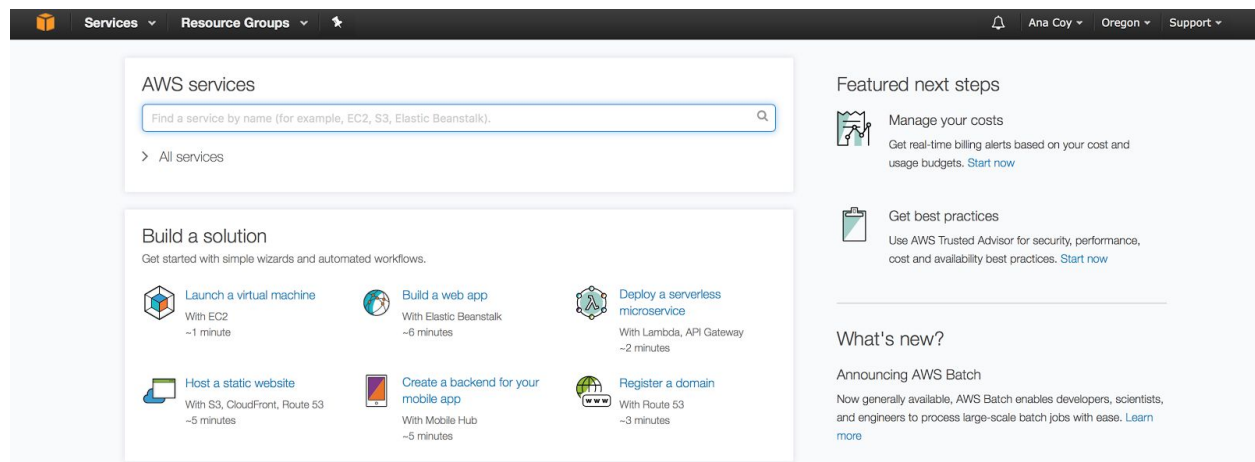
4. To setup an account for Georgia's select 'Company Account' and continue following the sign up instructions from there.

A screenshot of the 'Contact Information' section of the AWS sign-up form. At the top left is the AWS logo, and at the top right are 'English' and 'Sign Out' links. The title 'Amazon Web Services Sign Up' is centered. The 'Contact Information' section contains two radio buttons: 'Company Account' (selected) and 'Personal Account'. A red arrow points to the 'Company Account' button. Below the radio buttons is a list of required fields, each with an asterisk: 'Full Name*', 'Company Name*', 'Country*' (with a dropdown menu showing 'United States'), 'Address*' (with two input lines for street and apartment/suite), 'City*', 'State / Province or Region*', 'Postal Code*', and 'Phone Number*'. Each field has a corresponding input box.

5. Once you have completed the sign up process you should end up at this page. Hover over the 'My Account' tab and click 'AWS Management Console' to continue with the system installation



6. You will be redirected to the following page



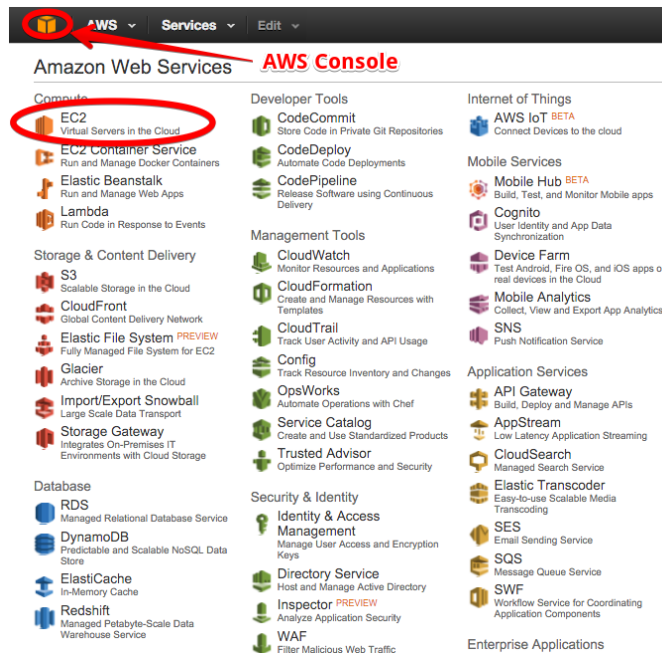
Create a GitHub Account

1. Go to <https://github.com/pricing>
2. Click on "Join Github for free"
3. Under "Create you personal account," type your username, email address, and password, then click "Create an Account"
4. Select the Free account type.

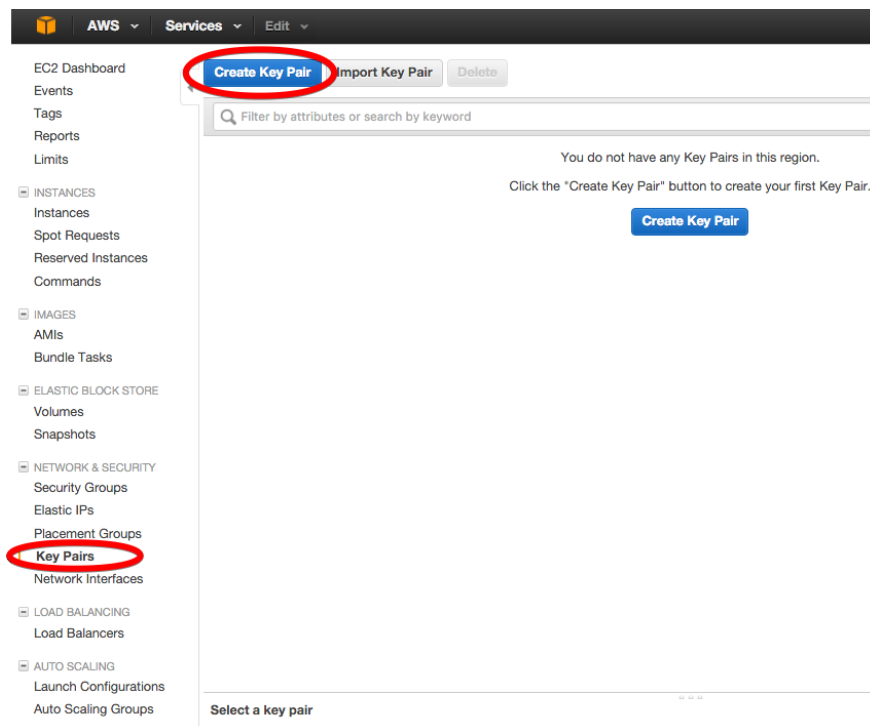
5. Click Finish sign up.

Setting Up an AWS EC2 instance:

1. Create a Key Pair
 - a. Navigate to AWS Console, then click on EC2



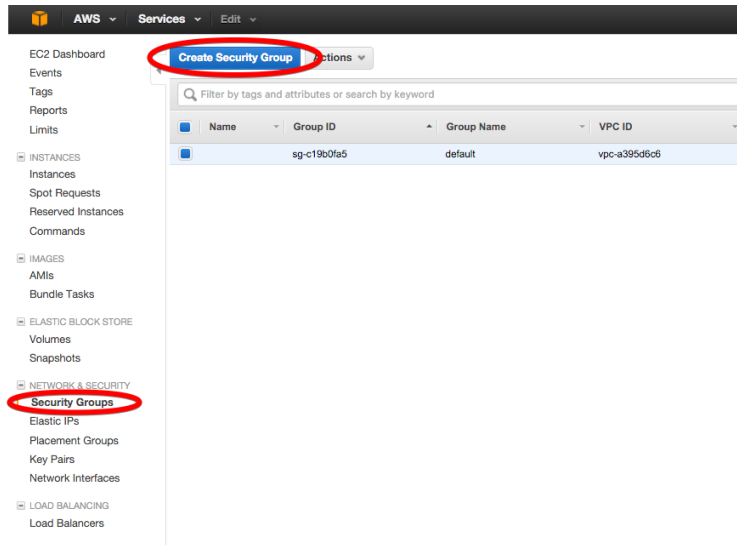
- b. On the left pane, click on Key Pairs, then click on Key Pair.



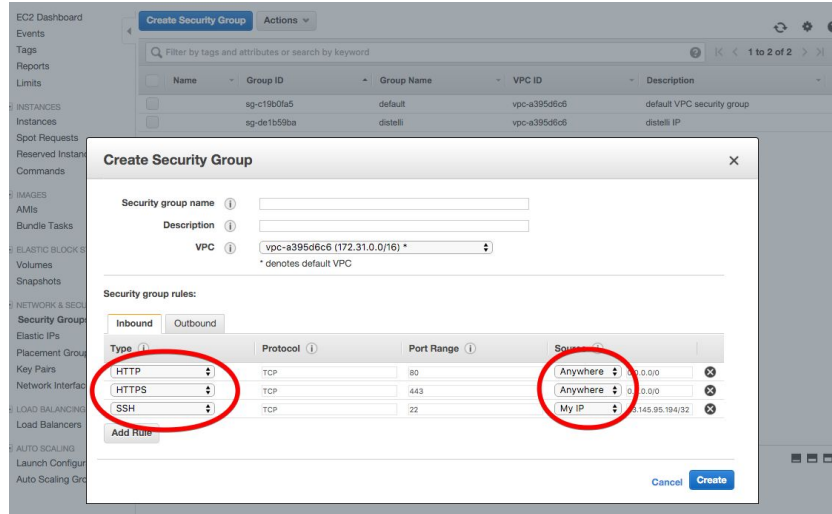
- c. Enter a name for your key, then click Create. The Key Pair will be automatically downloaded.
 - i. Move this key to a permanent location.
 - ii. You will need to change the permissions of this key to read only:
 - 1. Enter the following command: `chmod 400 [key-name].pem`

2. Create a Secure Group

- a. Navigate to the left pane of your EC2, and click on Security Groups, then click Create Security Group.

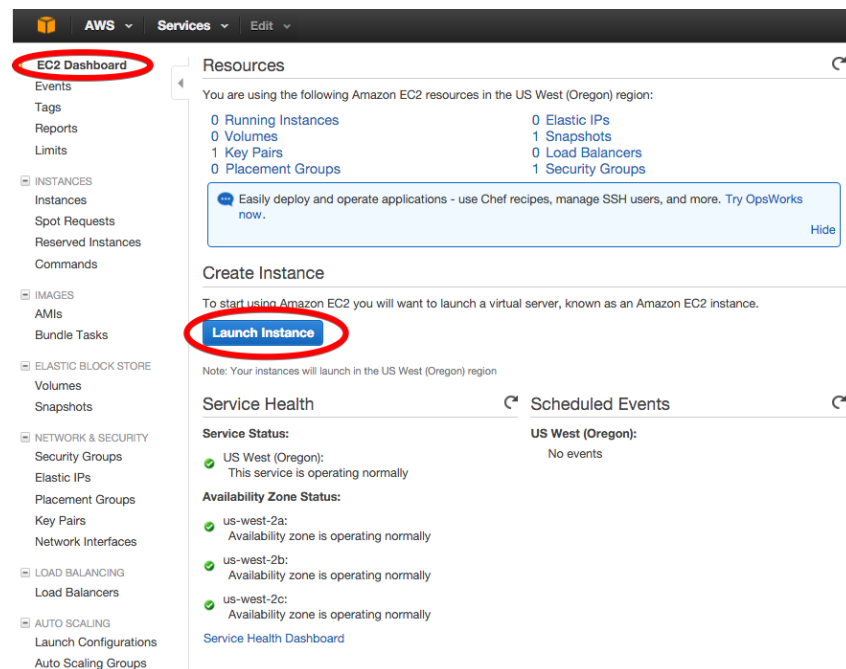


- b. Fill out the Security Group Name and give it a Description
 - i. On the Inbound tab, click Add Rules create the following 3 rules
 - 1. Select HTTP under Type.
 - a. Make sure Source is set to Anywhere.
 - 2. Select HTTPS under Type.
 - a. Make sure Source is set to Anywhere.
 - 3. Select SSH under Type.
 - a. Make sure Source is set to My IP.



3. Launch EC2 Instance

- One the left pane, click on EC2 Dashboard, then click on Launch Instance



- Choose your server OS and proceed.

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Tag Instance 6. Configure Security Group 7. Review

Step 1: Choose an Amazon Machine Image (AMI)

[Cancel and Exit](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, our user community, or the AWS Marketplace; or you can select one of your own AMIs.

Quick Start 1 to 22 of 22 AMIs

My AMIs

AWS Marketplace

Community AMIs

☐ Free tier only

Amazon Linux
Free tier eligible

Amazon Linux AMI 2015.09.1 (HVM), SSD Volume Type - ami-f0091d91

The Amazon Linux AMI is an EBS-backed, AWS-supported image. The default image includes AWS command line tools, Python, Ruby, Perl, and Java. The repositories include Docker, PHP, MySQL, PostgreSQL, and other packages.

Root device type: ebs Virtualization type: hvm

Select

Red Hat
Free tier eligible

Red Hat Enterprise Linux 7.1 (HVM), SSD Volume Type - ami-4dbf9e7d

Red Hat Enterprise Linux version 7.1 (HVM), EBS General Purpose (SSD) Volume Type

Root device type: ebs Virtualization type: hvm

Select

SUSE Linux
Free tier eligible

SUSE Linux Enterprise Server 12 (HVM), SSD Volume Type - ami-d7450be7

SUSE Linux Enterprise Server 12 (HVM), EBS General Purpose (SSD) Volume Type. Public Cloud, Advanced Systems Management, Web and Scripting, and Legacy modules enabled.

Root device type: ebs Virtualization type: hvm

Select

Ubuntu
Free tier eligible

Ubuntu Server 14.04 LTS (HVM), SSD Volume Type - ami-5189a661

Ubuntu Server 14.04 LTS (HVM), EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).

Root device type: ebs Virtualization type: hvm

Select

Windows
Free tier eligible

Microsoft Windows Server 2012 R2 Base - ami-f8f715cb

Microsoft Windows 2012 R2 Standard edition with 64-bit architecture. (English)

Root device type: ebs Virtualization type: hvm

Select

- Select Configure Security Group on top of the page.
- Check the option Select an existing security group, then select the name of your group.

AWS Services Edit

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Tag Instance 6. Configure Security Group 7. Review

Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select an existing security group.

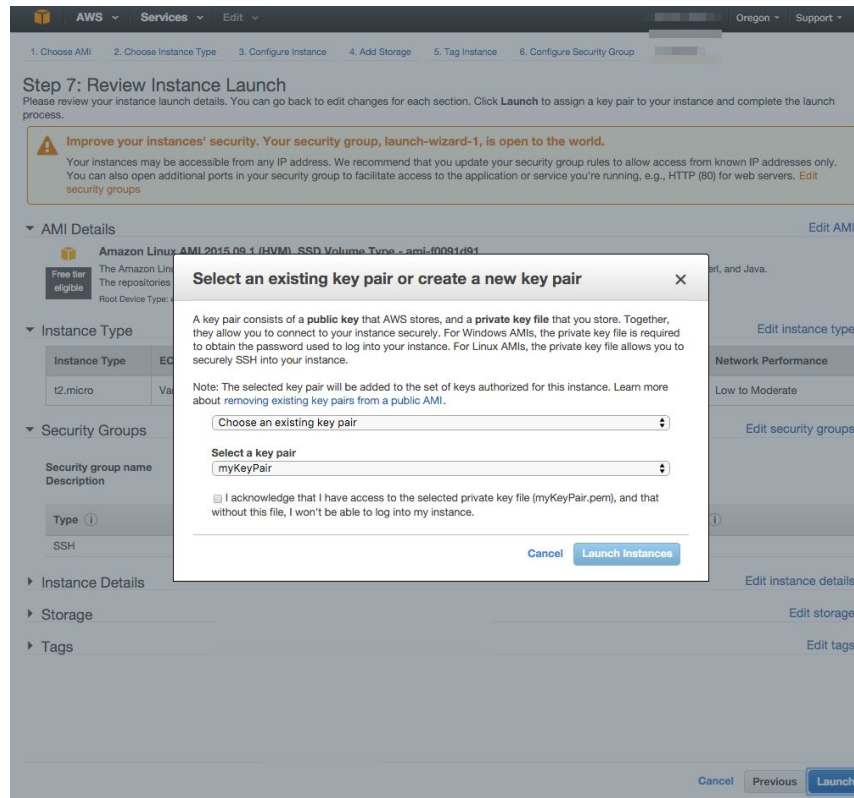
Assign a security group: ☐ Create a new security group ☒ Select an existing security group

Security Group ID	Name	Description
sg-c19b0fa5	default	default VPC security group
sg-531d5937	myCustomGroup	hello

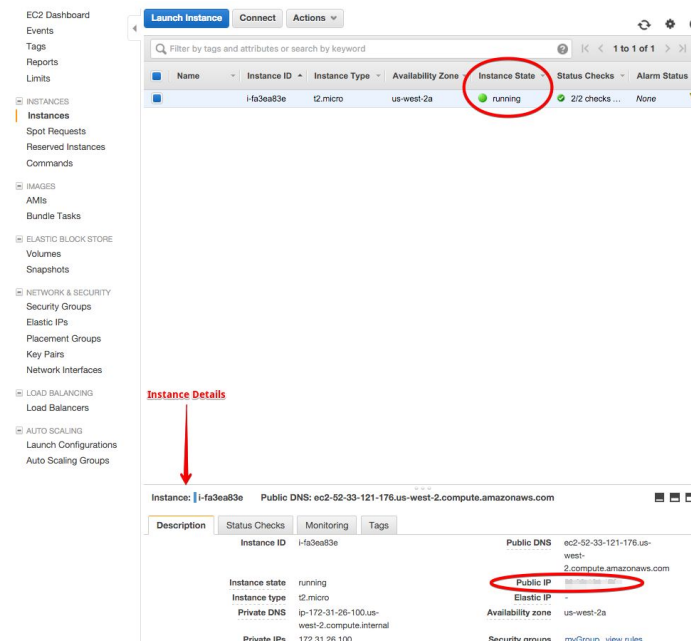
Inbound rules for sg-531d5937 (Selected security groups: sg-531d5937)

Type	Protocol	Port Range
This security group has no rules		

- Click Review and Launch.
- Click Launch.
 - Select the key pair you created, then select Launch instances.



- g. Navigate to the left pane of the EC2, and click on Instances.
 - i. Check out the Instance State, and make sure it is running
 - ii. You will need your instance IP address to connect your instance
 - iii. Your instance IP address can be found in the instance's details



4. Connect to Instance

- a. Open your terminal.
 - i. Navigate to the directory where your Key Pair is saved
 - ii. Enter the following commands:
 1. `chmod 700 [your key name pair].pem`
 2. `ssh -i "[your key name pair].pem" ubuntu@[instance IP address]`
- b. Enter the following command to update your instance:
 - i. `sudo apt-get update`
 - ii. `sudo apt-get install apache2 libapache2-mod-wsgi python-pip python-dev`
 - iii. `sudo pip install django`
 - iv. `sudo apt-get install mysql-server python-mysqldb`
 - v. `sudo apt-get install git`
- c. Set up database - open your terminal
 - i. Enter 'mysql'
 - ii. You should see 'mysql>' on the left hand side
 - iii. Enter the following commands:
 1. `create database georgias;`
 2. `grant all on georgias.* to 'georgias' identified by 'georgiaspassword';`
 3. `exit;`
 - iv. Enter `'mysql -u georgias -pgeorgiaspassword georgias < georgias.sql'`
 - v. Enter 'mysql'
 - vi. Enter `'INSERT INTO auth_users VALUES ('admin_user', 'email@email.com', 'password');'`
 - vii. Close the terminal
- d. Pull your code files
 - i. If from github, enter `'git clone https://github.com/uva-slp/georgias.git'`
 - ii. Otherwise, through scp terminal commands
- e. Migrate database
 - i. Enter `'cd georgias'`
 - ii. `'python manage.py makemigrations'`
 - iii. `'python manage.py migrate'`
- f. Configure apache2
 - i. `'cd /etc/apache2/sites-enabled'`
 - ii. `'vi georgias.conf'`
 - iii. Paste or type this:
 1. **WSGIScriptAlias / /home/ubuntu/georgias/georgias/wsgi.py**
WSGI PythonPath /home/ubuntu/georgias/
<Directory /home/ubuntu/georgias/georgias>
<Files wsgi.py>
Require all granted
</Files>
</Directory>

**Alias /media/ /home/ubuntu/georgias/media/
Alias /static/ /home/ubuntu/georgias/static/**

**<Directory /home/ubuntu/georgias/static/>
Require all granted
</Directory>**

**<Directory /home/ubuntu/georgias/media/>
Require all granted
</Directory>**

- iv. Enter ':wq'
- v. Enter 'sudo service apache2 restart'
- g. Navigate to the public IP address and the application should be usable