

How to deploy projects on AWS Ubuntu servers

Click this: [AWS Ubuntu Servers](#)

AWS Ubuntu Server is a 3rd party server developed by Amazon.

1. Logging into the server

1) From mac/linux machine

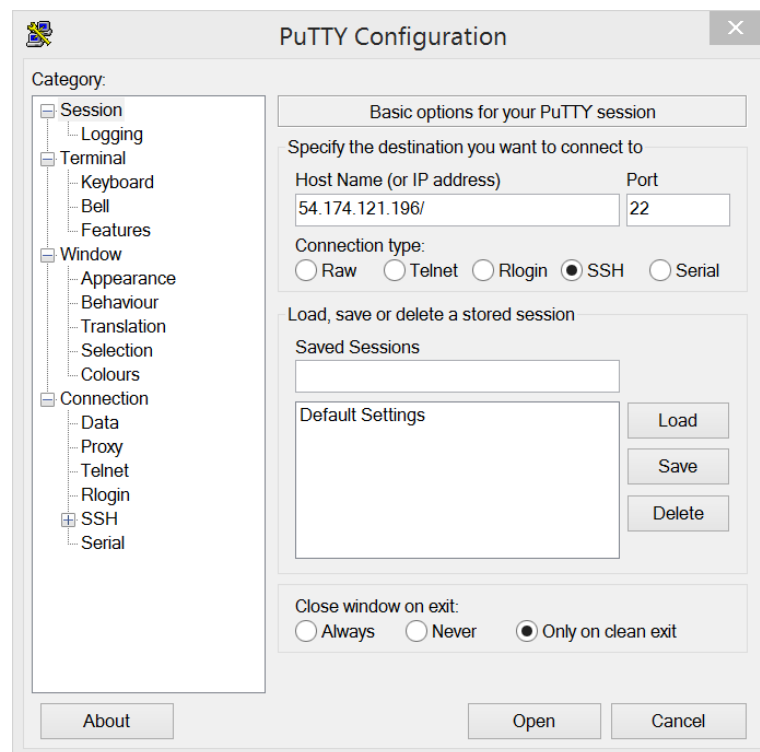
If you have mac or you have some linux OS (e.g. Ubuntu) then you would need to use terminal to connect to the server. Simply open the terminal and type:

```
ssh student@54.174.121.196
```

2) From windows machine

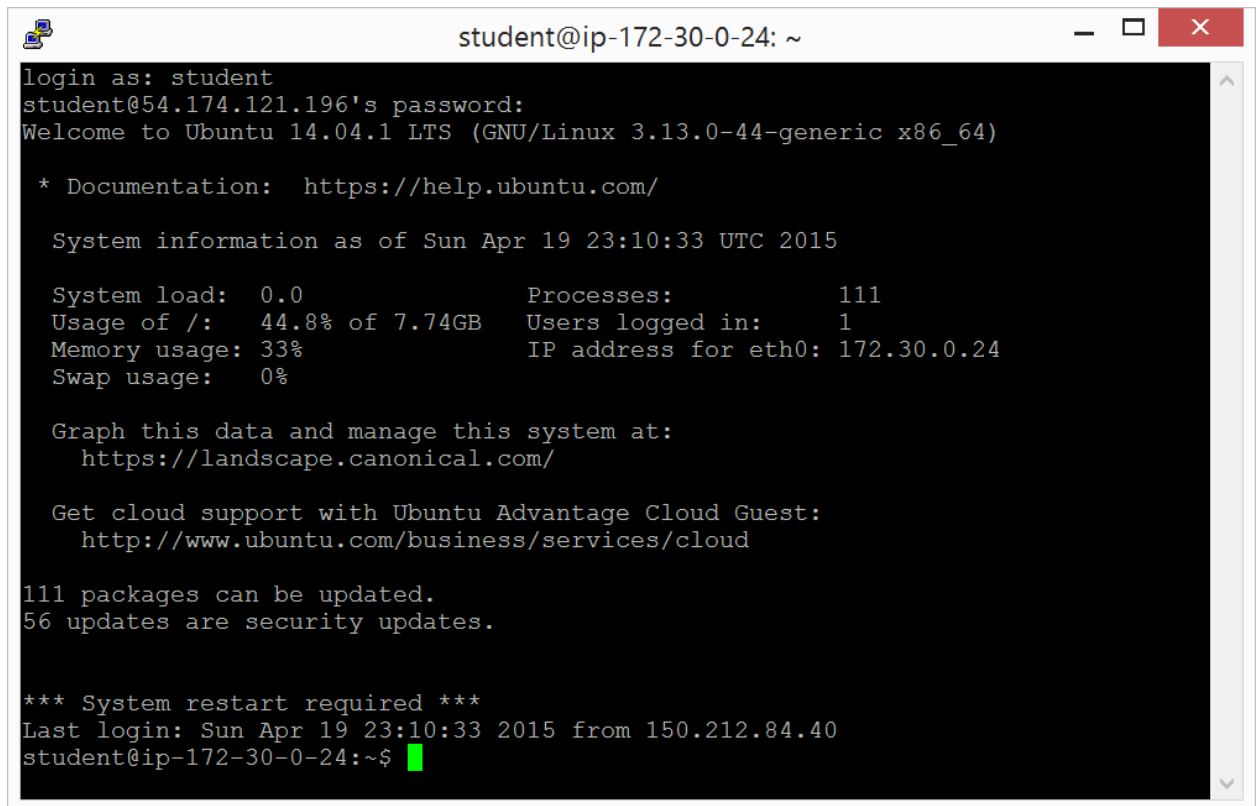
If you have Windows OS, then you need to use [PuTTY](#) software. Simply download it and run it. In the opened window enter server IP in the host area and leave port 22 as default. You can also save the setting for future use if you give it a name and click save button. See the picture below for the example:

[54.174.121.196/](#)



Then click open and in the opened window enter your username, hit enter, and then enter your password.

(username: student password: infsci271141)

A terminal window titled 'student@ip-172-30-0-24: ~' with standard window controls. The terminal output shows a successful login for the 'student' user. It displays system information for Ubuntu 14.04.1 LTS, including system load, memory usage, and processes. It also lists available updates and provides links for documentation and cloud support. The prompt is 'student@ip-172-30-0-24:~\$' with a green cursor.

```
student@ip-172-30-0-24: ~
login as: student
student@54.174.121.196's password:
Welcome to Ubuntu 14.04.1 LTS (GNU/Linux 3.13.0-44-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

System information as of Sun Apr 19 23:10:33 UTC 2015

System load:  0.0                Processes:            111
Usage of /:   44.8% of 7.74GB     Users logged in:     1
Memory usage: 33%                IP address for eth0: 172.30.0.24
Swap usage:  0%

Graph this data and manage this system at:
  https://landscape.canonical.com/

Get cloud support with Ubuntu Advantage Cloud Guest:
  http://www.ubuntu.com/business/services/cloud

111 packages can be updated.
56 updates are security updates.

*** System restart required ***
Last login: Sun Apr 19 23:10:33 2015 from 150.212.84.40
student@ip-172-30-0-24:~$
```

2. Preparing Ubuntu

Keyword.sh, install it

clip: Keyword finishes following steps:

```
cd MultiDBs-Utils
mvn install
cd MultiDBs-KeywordSearchG-Server
mvn install
cd MultiDBsKeywordSearchGServerAPI/target
nohup java -jar multidbskeywordsearchgserverapi-0.1-SNAPSHOT.jar > log.out 2> error.log < /dev/null &
```

```
cd /usr/share/nginx
```

```
ln -s /opt/project/MultiDBs-KeywordSearchG-WebCleint/public_html
```

3. Getting the source code to the server

1) Check project folder

```
student@ip-172-30-0-24:~$ ls
```

```
student@ip-172-30-0-24:~$ ls
2.2.0-M02.tar.gz          neo4j
commands.txt             neo4j-2.2.0-M02
json.jsp                 neo4j-community-2.2.0-RC01-unix.tar.gz
MultiDBs-KeywordSearchG-Server  neo4j.tar.gz
MultiDBs-KeywordSearchG-WebCleint  setup.sh
MultiDBs-Utills
student@ip-172-30-0-24:~$ ^C
student@ip-172-30-0-24:~$ █
```

2) Clone all required Github repos.

```
sudo git clone https://github.com/infsci2711/MultiDBs-KeywordSearchG-Server.git
```

The server code will be served directly from the project folder with Jetty web server, however to server "static" Web UI (Client code) you have several options:

- Create symlink to the client code in the project folder
- Copy client code from the project folder to the nginx folder
- git clone your client code directly in the nginx folder

Good luck :)