EXPLORE DIGITAL SKILLS

Introduction to the Cloud - Part I

A Beginner’s Guide to Using AWS
Elastic Cloud Compute (EC2)

Important Notice - Accessing Amazon Web Services

Get your credentials

Prior to this train, you should have received an email entitled:

‘***AWS EDSA Login Credentials for <your first name>***’

This email contained important information and instructions on how to log into your EXPLORE AWS account using the AWS console.

If you didn’t receive the email (please check your spam folder), contact an academy mentor for assistance.

Ensure you can Login

You will need to have completed the instructions in the email in order to complete this train, and further AWS-related trains.

If you cannot complete the instructions for any reasons, please let us know.

We wish you all the best in this new digital journey of yours!

EXPLORE DIGITAL SKILLS

Topics to be covered

By the end of this train you should be able to:

1. Have a **basic understanding of the cloud** and cloud providers.

2. Login to the AWS console to **launch your own EC2 instance**.

3. Use the command line to **login to your remote instance.**

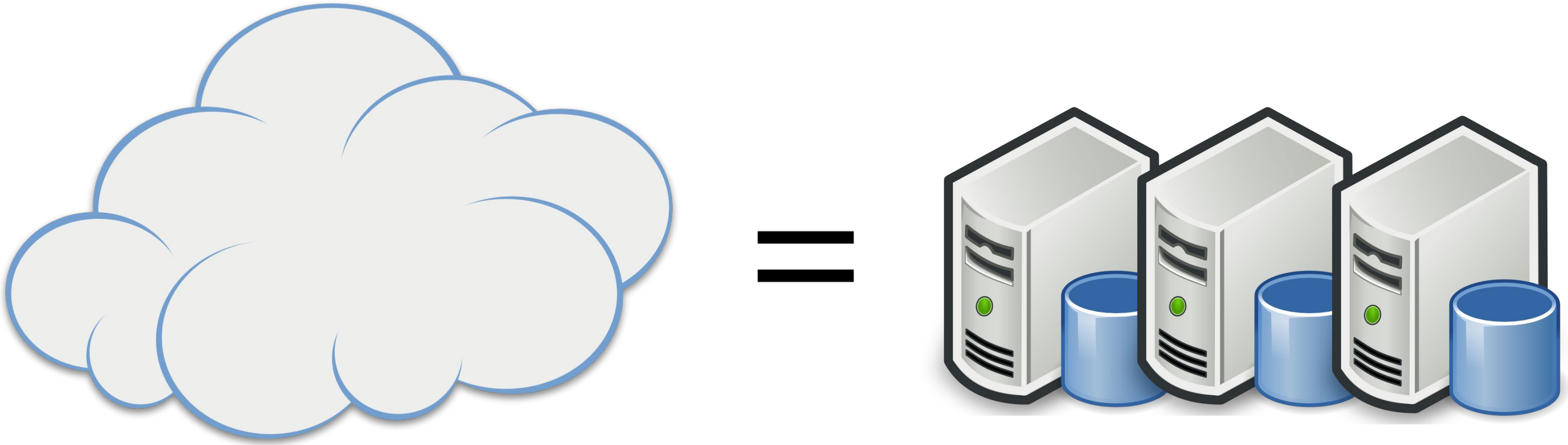
4. **Understand EXPLORE's housekeeping rules** to use your instance responsibly.

Introduction to the Cloud - Part I

Basic Concepts

Photo by: NASA from Unsplash

What is the cloud?



The cloud is actually **just a whole bunch of computers that are kept in massive data centres around the world.** When you are using the cloud, your data are not being stored magically in thin air, they're simply being stored on a server (or rather multiple servers) somewhere else in the world.

To access your data in the cloud, you use the internet, which is a colossal network of servers connected by cables across land and even underneath the sea.

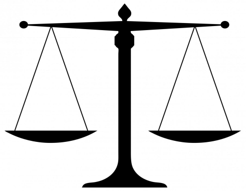
EXPLORE DIGITAL SKILLS

Why use cloud services instead of traditional infrastructure?

Prior to the advent of cloud services, companies could only think of their computing infrastructure in terms of *fixed hardware installations* which, amongst other things, were:

- **Static** - Hardware stacks have fixed storage, compute, and networking capacity. They also have to be bought upfront as a capital expense, which can be a considerable cost!
- **Difficult to maintain** - Managing onsite infrastructure requires many specialised roles including facility management, system administration, network engineering, and system technicians.
- **Ineffiecient** - Unless coordinated perfectly, onsite hardware is either under or over utilised at any time during the day; wasting productivity during peak traffic periods or financial resources during activity lulls.

Using cloud services there are five distinct advantages over traditional fixed hardware installations:

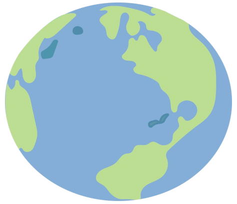


Trade **Capital** (upfront) expenses for **Variable** (as you use) expenses

Significantly **cheaper** per unit of compute power

Dynamic, on demand resources, which **elastically scale** to need.

No maintenance **costs**.



Provide instant **global reach**

EXPLORE DIGITAL SKILLS

Concerns around cloud security

Security concerns associated with cloud computing

Is the cloud safe?

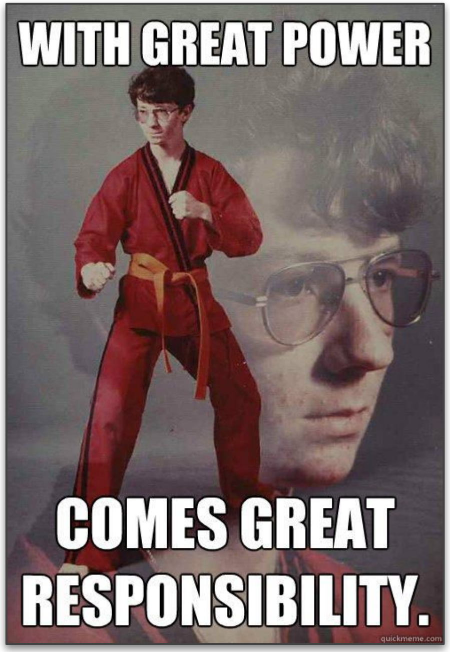
The answer to this is, if you educate yourself with some good practices, this is a definite **Yes**.

The good news is that we have some pretty large companies investing huge amounts of time and money into keeping the cloud safe through things like **encryption** and **multi-factor authentication**.

Some concerns include:

- Misconfigurations of cloud security settings.
- Hijacking of accounts associated with weak password security.
- Malicious insiders who may have access to an organisations network and sensitive information.

You can learn more about cloud security here.



Introduction to the Cloud - Part I

Steps to Launch a Remote Instance

Photo by: NASA from Unsplash

What are Remote Instances?

A remote instance can be thought of as setting up a fully functioning computer system in the cloud. In the case of EC2 we are using Amazon Machine Images as the master image, to create a Virtual Machine (VM) or remote instance in a selected AWS region where we'll run our code, applications, software, etc.

AWS EC2

Amazon EC2 instances created from Windows **Amazon Machine Image (AMI)** enables a user to connect to the instance using Remote Desktop.

Amazon Machine Image

An AMI is a **master image**, preloaded with all the operating system and software configurations needed to quickly spin up an AWS EC2 resource.

Master Image

A master image or golden image is a template used to create a **Virtual Machine**.

Virtual Machine

A VM is a fully-fledged computer system (with networking, compute and storage resources) which is simulated within a physical computer.

1) Using the Right Software

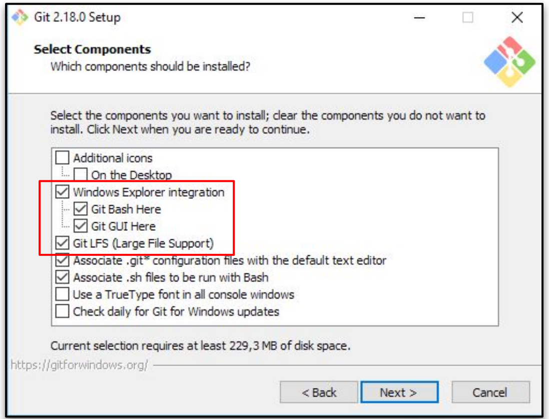
For this train you will be required to **work from the terminal on your machine**. The **set-up might** differ if you are working with a Windows, Mac or Linux Operating System. **However**, once the **correct packages are installed** all the **underlying steps are the same** and if they do differ we will be sure to make mention of the exceptions.

Instructions for Windows

- Download and install Git and Git Bash

Instructions for Mac/Linux

- Ensure that you can open the Terminal



Select Components
Which components should be installed?

Select the components you want to install; clear the components you do not want to install. Click Next when you are ready to continue.

- [ ] Additional icons
  - [ ] On the Desktop
- [✓] Windows Explorer integration
  - [✓] Git Bash Here
  - [✓] Git GUI Here
- [✓] Git LFS (Large File Support)
- [✓] Associate .git* configuration files with the default text editor
- [✓] Associate .sh files to be run with Bash
- [ ] Use a TrueType font in all console windows
- [ ] Check daily for Git for Windows updates

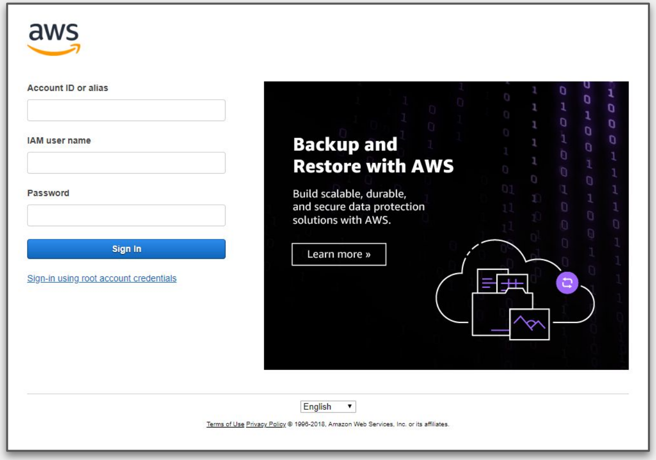
Current selection requires at least 229,3 MB of disk space.

https://gitforwindows.org/

EXPLORE DIGITAL SKILLS

2) Navigating to the AWS Console

The AWS Console is your first point of entry to the various AWS cloud services. This is where you will land after you have signed into your new AWS account from the following URL: https://edsa.signin.aws.amazon.com/console



aws

Account ID or alias

IAM user name

Password

Sign In

Sign-in using root account credentials

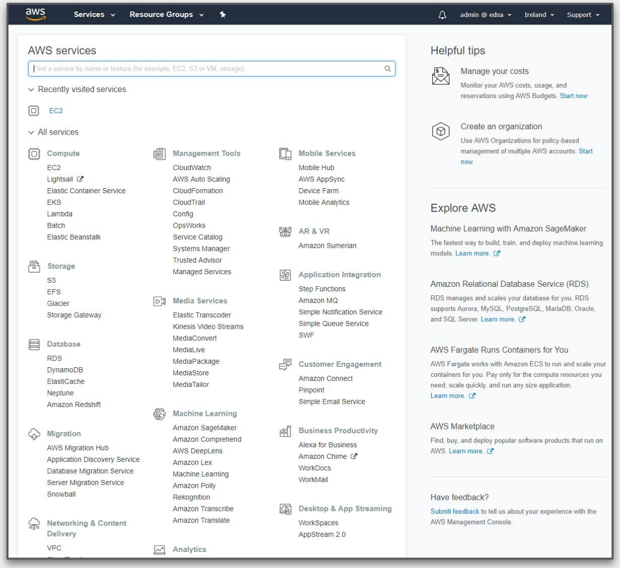
Backup and Restore with AWS

Build scalable, durable, and secure data protection solutions with AWS.

Learn more »

English

Terms of Use Privacy Policy © 1996-2018, Amazon Web Services, Inc. or its affiliates.



aws Services Resource Groups admin @ edsa Ireland Support

AWS services

Find a service by name or feature (for example, EC2, S3 or VM, storage).

Recently visited services

EC2

All services

Compute
EC2
Lightsail
Elastic Container Service
EKS
Lambda
Batch
Elastic Beanstalk

Storage
S3
EFS
Glacier
Storage Gateway

Database
RDS
DynamoDB
ElastiCache
Neptune
Amazon Redshift

Migration
AWS Migration Hub
Application Discovery Service
Database Migration Service
Server Migration Service
Snowball

Networking & Content Delivery
VPC

Management Tools
CloudWatch
AWS Auto Scaling
CloudFormation
CloudTrail
Config
OpsWorks
Service Catalog
Systems Manager
Trusted Advisor
Managed Services

Media Services
Elastic Transcoder
Kinesis Video Streams
MediaConvert
MediaLive
MediaPackage
MediaStore
MediaTailor

Machine Learning
Amazon SageMaker
Amazon Comprehend
AWS DeepLens
Amazon Lex
Machine Learning
Amazon Polly
Rekognition
Amazon Transcribe
Amazon Translate

Analytics

Mobile Services
Mobile Hub
AWS AppSync
Device Farm
Mobile Analytics

AR & VR
Amazon Sumerian

Application Integration
Step Functions
Amazon MQ
Simple Notification Service
Simple Queue Service
SWF

Customer Engagement
Amazon Connect
Pinpoint
Simple Email Service

Business Productivity
Alexa for Business
Amazon Chime
WorkDocs
WorkMail

Desktop & App Streaming
WorkSpaces
AppStream 2.0

Helpful tips

Manage your costs
Monitor your AWS costs, usage, and reservations using AWS Budgets. Start now

Create an organization
Use AWS Organizations for policy-based management of multiple AWS accounts. Start now

Explore AWS

Machine Learning with Amazon SageMaker
The fastest way to build, train, and deploy machine learning models. Learn more.

Amazon Relational Database Service (RDS)
RDS manages and scales your database for you. RDS supports Aurora, MySQL, PostgreSQL, MariaDB, Oracle, and SQL Server. Learn more.

AWS Fargate Runs Containers for You
AWS Fargate works with Amazon ECS to run and scale your containers for you. Pay only for the compute resources you need, scale quickly, and run any size application. Learn more.

AWS Marketplace
Find, buy, and deploy popular software products that run on AWS. Learn more.

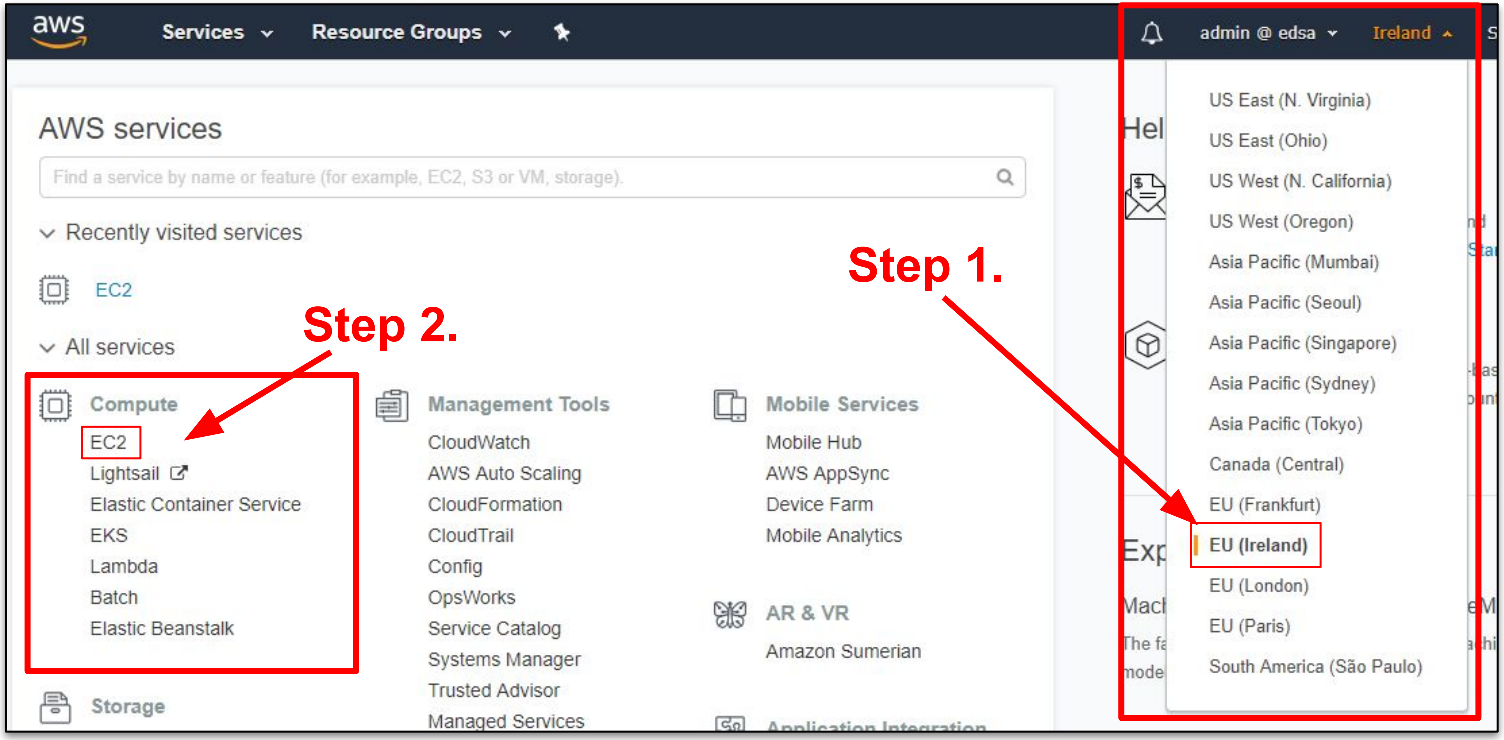
Have feedback?
Submit feedback to tell us about your experience with the AWS Management Console.

The console contains links to all of AWS' cloud services. There are literally hundreds of different cloud services, so we won't go into detail about all of them here. For now, we are just going to make use of the **EC2** service to create our first virtual machine.

EXPLORE DIGITAL SKILLS

3) Using the EC2 Service Console

Start by setting your region to **EU (Ireland)** up in the top right corner. Then click the **EC2** button under the **Compute** heading.



4) Select an AMI

Click the orange “**Launch Instance**” button. You’ll then be taken to a page that looks like the one below:

Choose your Machine Image

The first thing you’ll need to do is choose your **Amazon Machine Image** (or AMI).

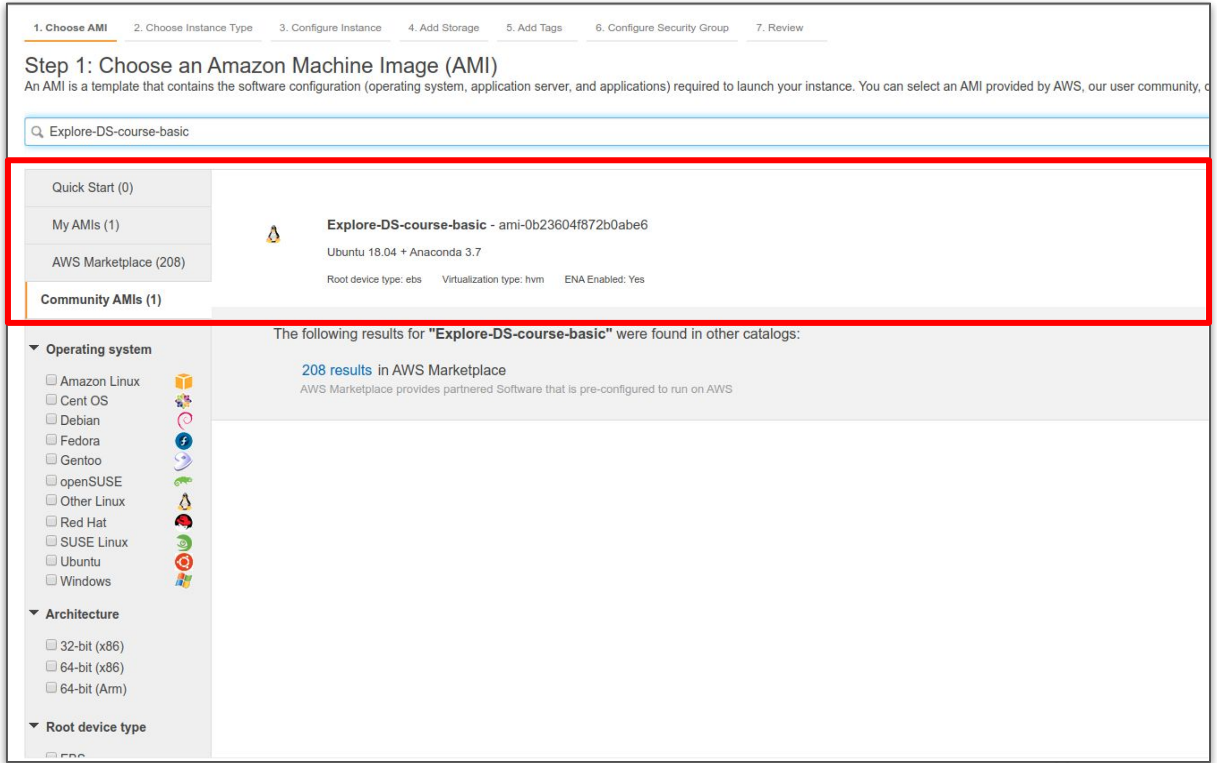
An AMI determines what operating system and software will be installed on your machine once it’s up and running.

Choosing the right AMI can save you time by not needing to go through installing software yourself.

For this tutorial, we are going to choose the **Explore-DS-course-basic** AMI found under the “Community AMIs” tab.

You can search for the AMI by name to find it easily.

NB: Your region MUST be set to EU(Ireland)



Step 1: Choose an Amazon Machine Image (AMI)

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,

Explore-DS-course-basic

Explore-DS-course-basic - ami-0b23604f872b0abe6

Ubuntu 18.04 + Anaconda 3.7

Root device type: ebs   Virtualization type: hvm   ENA Enabled: Yes

The following results for **"Explore-DS-course-basic"** were found in other catalogs:

208 results in AWS Marketplace

AWS Marketplace provides partnered Software that is pre-configured to run on AWS

5) Choose your Instance Type

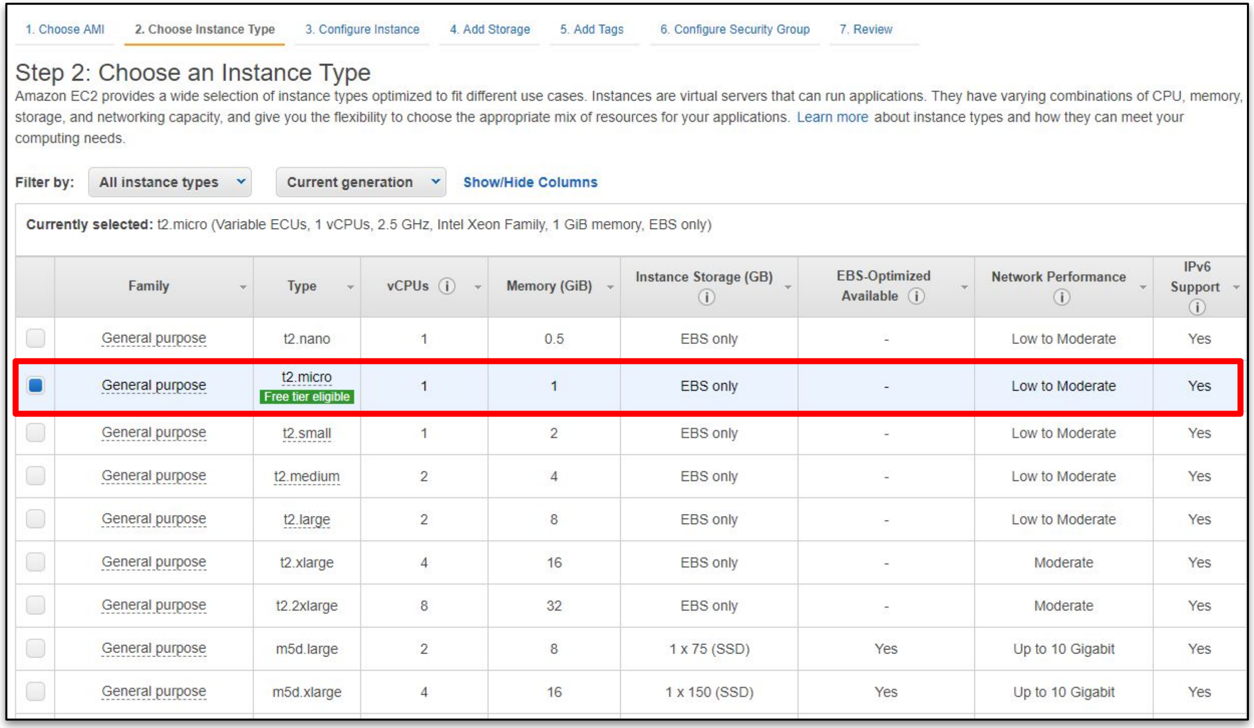
While the AMI determines the operating system and software on your machine, the **instance type determines the hardware** you will have in your virtual machine.

AWS Instance Types

There is a huge range of instances to choose from. For each instance type, you can see the specs shown in the **vCPUs** and **Memory** columns.

Take a quick scroll through all of the instances and you will see the wide range of machines available on AWS.

For now, we are going to use the t2.micro; a modest machine which will be suitable for hosting a basic Jupyter notebook which will allow us to perform some Python programming in the cloud.



Step 2: Choose an Instance Type

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. Learn more about instance types and how they can meet your computing needs.

Filter by: All instance types | Current generation | Show/Hide Columns

Currently selected: t2.micro (Variable ECUs, 1 vCPUs, 2.5 GHz, Intel Xeon Family, 1 GiB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
☐	General purpose	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
■	General purpose	t2.micro Free tier eligible	1	1	EBS only	-	Low to Moderate	Yes
☐	General purpose	t2.small	1	2	EBS only	-	Low to Moderate	Yes
☐	General purpose	t2.medium	2	4	EBS only	-	Low to Moderate	Yes
☐	General purpose	t2.large	2	8	EBS only	-	Low to Moderate	Yes
☐	General purpose	t2.xlarge	4	16	EBS only	-	Moderate	Yes
☐	General purpose	t2.2xlarge	8	32	EBS only	-	Moderate	Yes
☐	General purpose	m5d.large	2	8	1 x 75 (SSD)	Yes	Up to 10 Gigabit	Yes
☐	General purpose	m5d.xlarge	4	16	1 x 150 (SSD)	Yes	Up to 10 Gigabit	Yes

6) Add Tags

Skip forward to "**Step 5: Add Tags**" by selecting the relevant tab at the top of the web page.

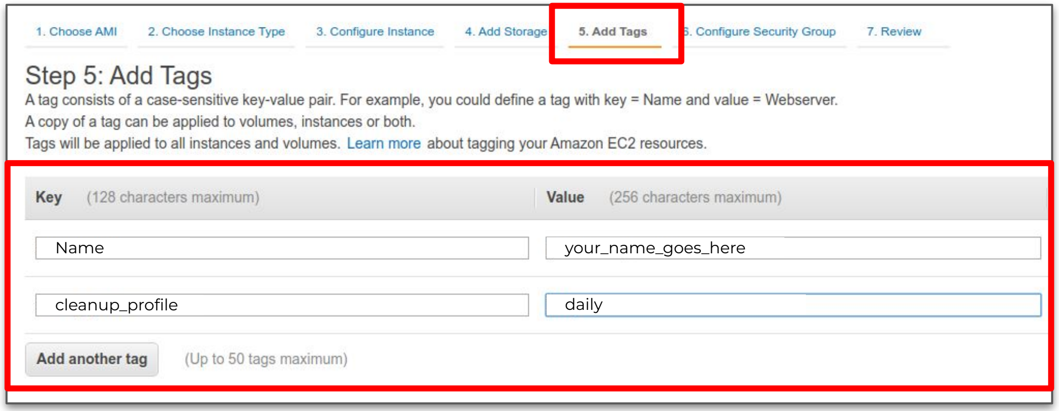
Add a **Name** tag:
- Use '***Name'*** as the Key
- Use ***your name*** as the Value

[NB] Add a **Housekeeping** tag:
- Use '**cleanup_profile**' as the Key
- Use '**daily**' as the Value

The **name tag** will help you find your instance more easily in the future.

The **name and housekeeping tags** are used by Explore to control spawned EC2 instances. **If you don't create these tags, your instance will be terminated** overnight without your consent.

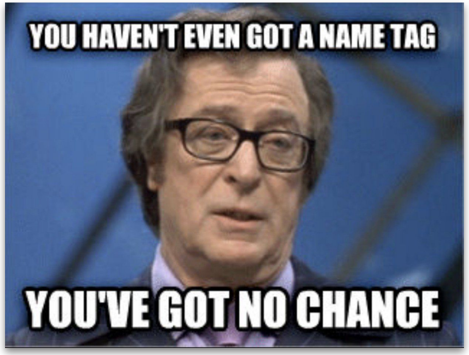
We will speak more about housekeeping at the end of this train.



Step 5: Add Tags

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver.
A copy of a tag can be applied to volumes, instances or both.
Tags will be applied to all instances and volumes. Learn more about tagging your Amazon EC2 resources.

Key (128 characters maximum)	Value (256 characters maximum)
Name	your_name_goes_here
cleanup_profile	daily



7) Choose a Security Group

You can now skip forward to step 6 to configure your **Security Group**. Choose “select an **existing** security group” and select the “**edsa-student-ec2-vpc-group**” group.

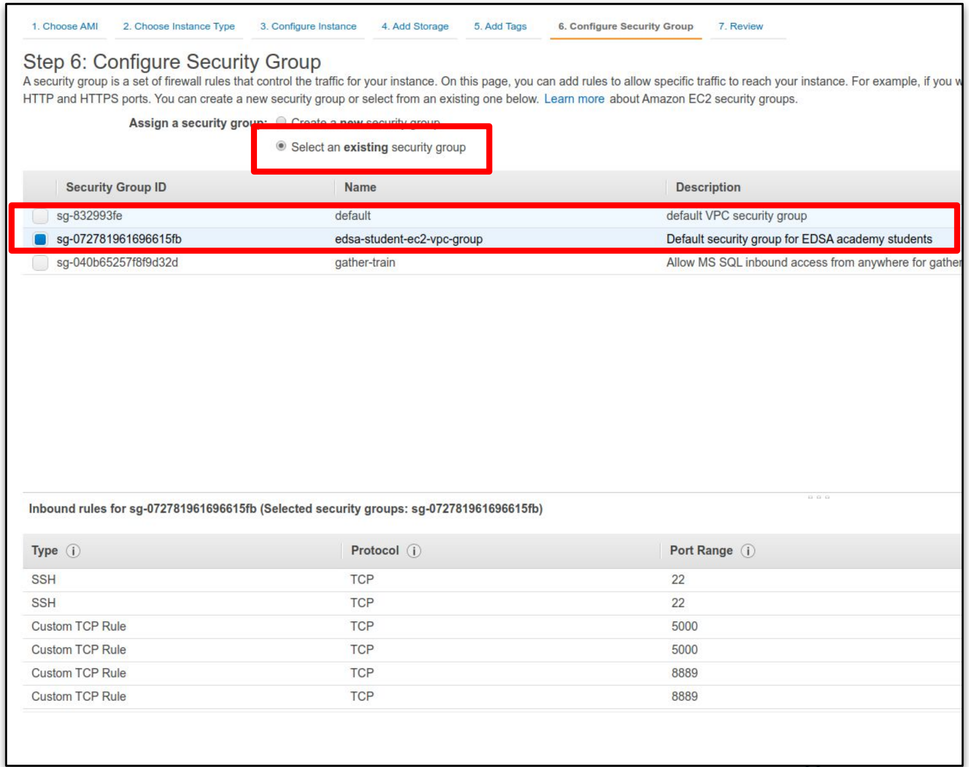
AWS Security Groups

In AWS, everything needs to be assigned to at least one security group.

A security group is a set of rules. These rules define *how* we would like our instance to be able to communicate with other servers over the internet.

We have configured the default security group to allow your local web browser (the one on the computer you are using to view these slides) to communicate with your instance’s port 8889.

Think of your computer like an apartment block. **Its IP address is like its street address**. The **port numbers are like the apartment numbers** - each ‘apartment’ is a different port on your computer. We’re going to host a Jupyter notebook on this port, so we will need to allow access to it so that we can ‘visit’ it, and display it in your personal browser window.



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 instance. For example, if you w
HTTP and HTTPS ports. You can create a new security group or select from an existing one below. Learn more about Amazon EC2 security groups.

Assign a security group: ○ Create a new security group [✓ Select an **existing** security group]

	Security Group ID	Name	Description
☐	sg-832993fe	default	default VPC security group
☑	sg-072781961696615fb	edsa-student-ec2-vpc-group	Default security group for EDSA academy students
☐	sg-040b65257f8f9d32d	gather-train	Allow MS SQL inbound access from anywhere for gather

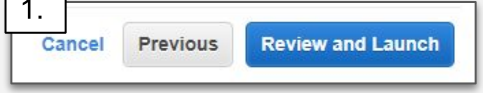
Inbound rules for sg-072781961696615fb (Selected security groups: sg-072781961696615fb)

Type	Protocol	Port Range
SSH	TCP	22
SSH	TCP	22
Custom TCP Rule	TCP	5000
Custom TCP Rule	TCP	5000
Custom TCP Rule	TCP	8889
Custom TCP Rule	TCP	8889

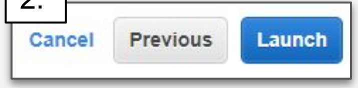
8) Launch Instance

You are now ready to launch your instance! Simply click the blue **Launch** buttons (Box 1 & 2). When prompted to select or create a key pair, choose the option to **proceed without a key pair** (Box 3).

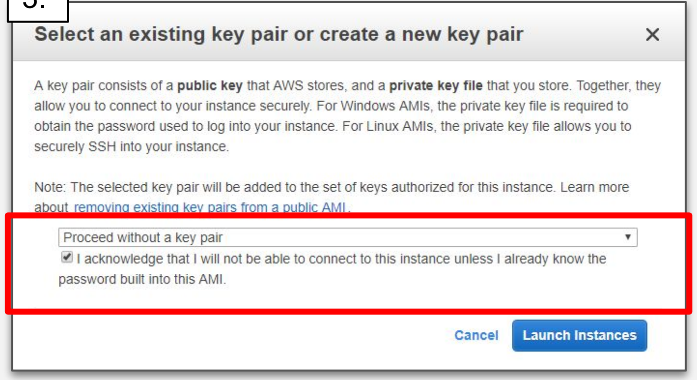
1.



2.



3.



Select an existing key pair or create a new key pair

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

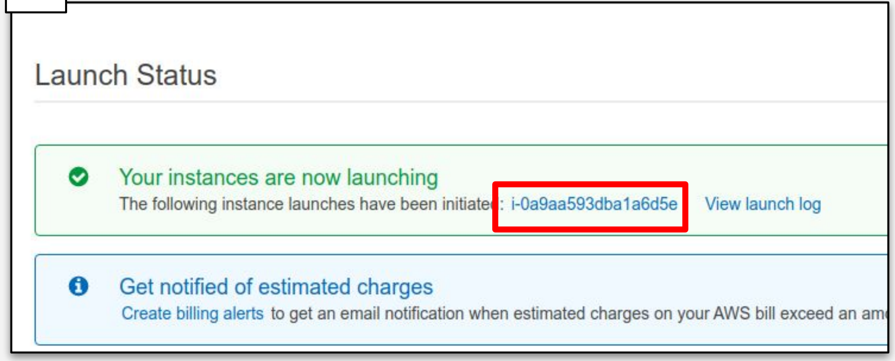
Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about removing existing key pairs from a public AMI.

Proceed without a key pair

[✓] I acknowledge that I will not be able to connect to this instance unless I already know the password built into this AMI.

After you launch your instance, you should get a **success message** confirming that your instance is launching (Box 4). **Click on the instance-id** provided within this message to navigate to your launched instance within the 'Instance View' page of the EC2 Dashboard.

4.



Launch Status

Your instances are now launching

The following instance launches have been initiated: i-0a9aa593dba1a6d5e View launch log

Get notified of estimated charges

Create billing alerts to get an email notification when estimated charges on your AWS bill exceed an am

9) Connect to your Instance: Preliminaries

Your virtual machine is now being started. You can always find it again within the instance view page using the name tag you setup in Step 6. Alternatively, please **name your instance**! You can do this by hovering over the name field and clicking on the pencil that appears. Use your first name followed by _edsa, i.e. <first_name>_edsa.

In preparation to connect, select your instance using the check box on the left and then look for the **IPv4 Public IP** address in the “Description” tab at the bottom of the page. **Copy this value** onto your system’s clipboard.

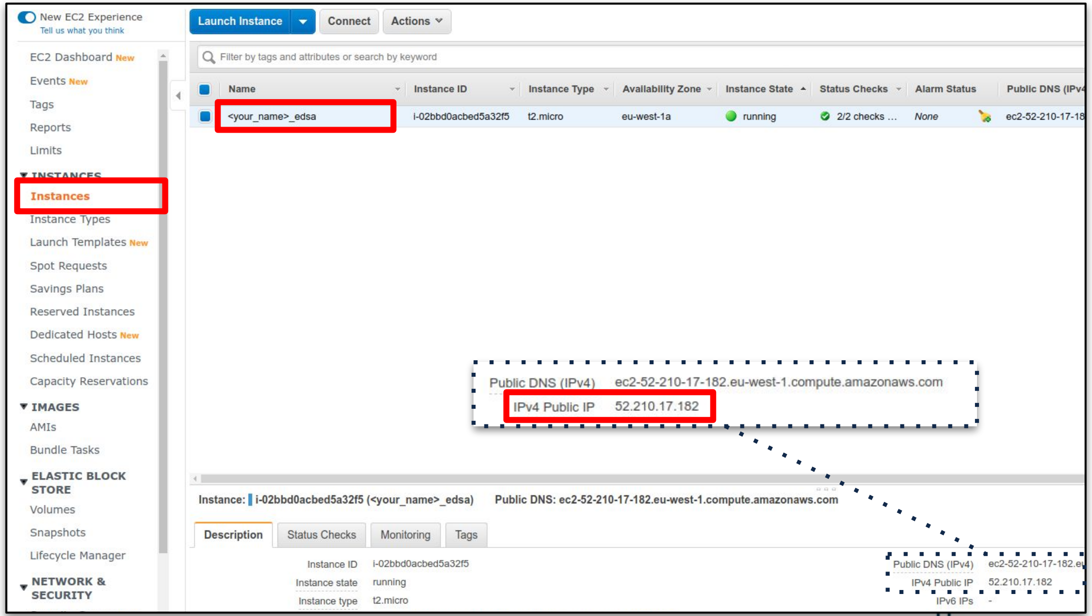
Now have a remote machine, but the only problem is that it’s in Ireland!

So we are going to use our local computer’s screen and keyboard to work on the remote machine. You’ll also hear instances referred to as remote machines, due to the fact that you access them remotely.

Now you will need to **open a terminal to connect remotely.**

[For Windows users]
Open GitBash

[For Mac/Linux users]
Open a normal terminal



Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4
<your_name>_edsa	i-02bbd0acbed5a32f5	t2.micro	eu-west-1a	running	2/2 checks ...	None	ec2-52-210-17-18

Public DNS (IPv4) ec2-52-210-17-182.eu-west-1.compute.amazonaws.com
IPv4 Public IP 52.210.17.182

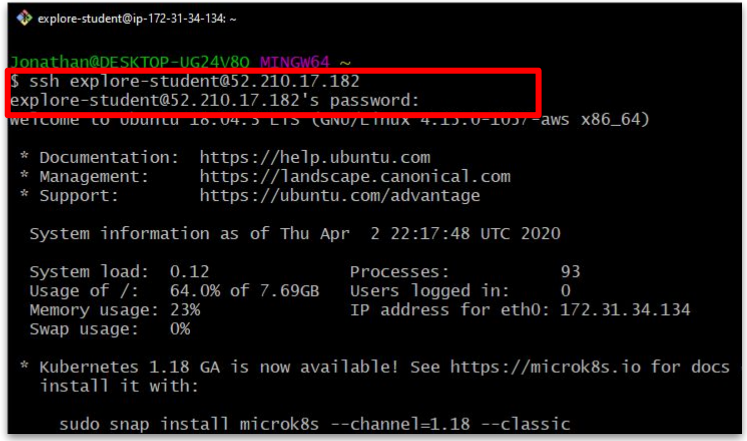
Instance: i-02bbd0acbed5a32f5 (<your_name>_edsa) Public DNS: ec2-52-210-17-182.eu-west-1.compute.amazonaws.com

Instance ID	i-02bbd0acbed5a32f5	Public DNS (IPv4)	ec2-52-210-17-182.e
Instance state	running	IPv4 Public IP	52.210.17.182
Instance type	t2.micro	IPv6 IPs	-

10) Connect to your Instance: Going Live

Remember to copy the **IPv4 Public IP** address in the Description tab of your instance.

- Type “`ssh explore-student@<IP-address>`” into the terminal, where <IP-address> is the public IP address you’ve just copied.
- Answer “**yes**” to the prompt.
- Use *“explore”* as your **password** to sign into your machine (beware; your cursor won’t react to you entering your password, so type carefully)

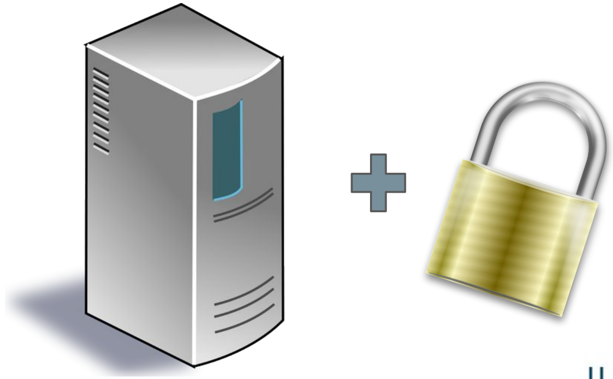


```
explore-student@ip-172-31-34-134: ~

Jonathan@DESKTOP-UG24V80 MINGW64 ~
$ ssh explore-student@52.210.17.182
explore-student@52.210.17.182's password:
Welcome to Ubuntu 18.04.3 LTS (GNU/Linux 4.15.0-1057-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Thu Apr  2 22:17:48 UTC 2020

  System load:  0.12              Processes:           93
  Usage of /:   64.0% of 7.69GB   Users logged in:     0
  Memory usage: 23%               IP address for eth0: 172.31.34.134
  Swap usage:   0%

 * Kubernetes 1.18 GA is now available! See https://microk8s.io for docs
   install it with:

     sudo snap install microk8s --channel=1.18 --classic
```

At this point, you are on the **command line** of your very own remote machine, congratulations!

[NB] The first thing you should do is change your password to something more secure*.

Type “**`passwd`**” and follow the prompts to change your password (just don’t forget it!). You will use this new password to login to your remote instance in the future.

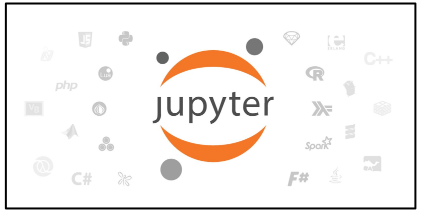


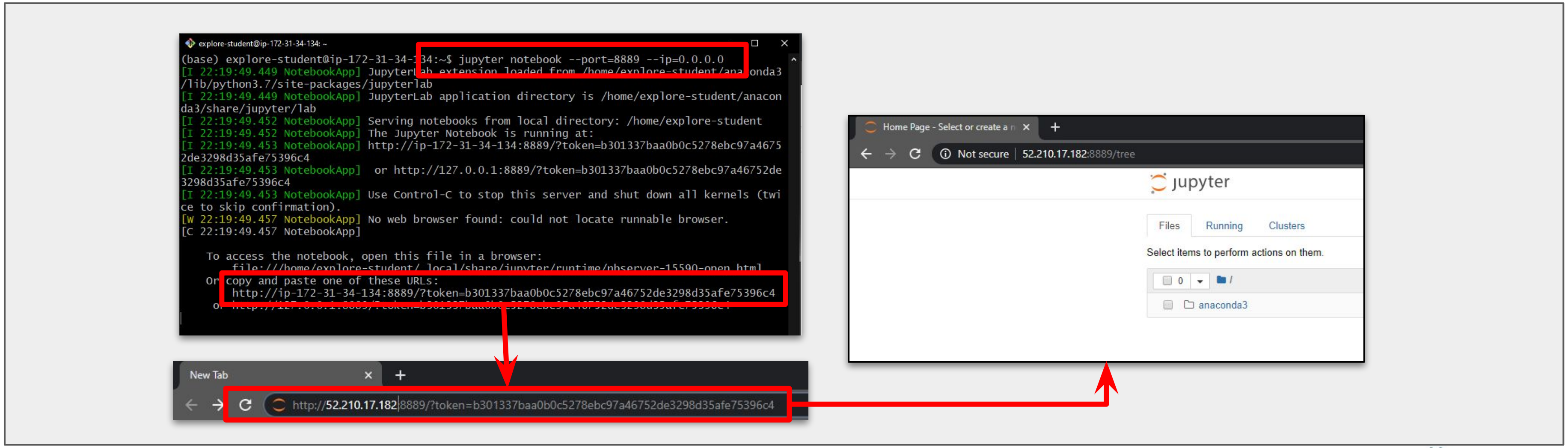
*For some password inspiration, see: https://imgs.xkcd.com/comics/password_strength.png

11) Launch an Application: Jupyter Notebook

Now we're ready to launch our first application on our remote instance. To do this, we'll **start up a *Jupyter Notebook* server**, which we'll be able to connect to from your own browser on your local machine. In your terminal, do the following:

- Type "**`jupyter notebook --port=8889 --ip=0.0.0.0`**"
- Copy/paste the URL provided by Jupyter into your local browser
- Replace the given IP value with your machine's (EC2's) public IP address





```
explore-student@ip-172-31-34-134: ~
(base) explore-student@ip-172-31-34-134:~$ jupyter notebook --port=8889 --ip=0.0.0.0
[I 22:19:49.449 NotebookApp] JupyterLab extension loaded from /home/explore-student/anaconda3
/lib/python3.7/site-packages/jupyterlab
[I 22:19:49.449 NotebookApp] JupyterLab application directory is /home/explore-student/anacon
da3/share/jupyter/lab
[I 22:19:49.452 NotebookApp] Serving notebooks from local directory: /home/explore-student
[I 22:19:49.452 NotebookApp] The Jupyter Notebook is running at:
[I 22:19:49.453 NotebookApp] http://ip-172-31-34-134:8889/?token=b301337baa0b0c5278ebc97a4675
2de3298d35afe75396c4
[I 22:19:49.453 NotebookApp]  or http://127.0.0.1:8889/?token=b301337baa0b0c5278ebc97a46752de
3298d35afe75396c4
[I 22:19:49.453 NotebookApp] Use Control-C to stop this server and shut down all kernels (twi
ce to skip confirmation).
[W 22:19:49.457 NotebookApp] No web browser found: could not locate runnable browser.
[C 22:19:49.457 NotebookApp]

    To access the notebook, open this file in a browser:
        file:///home/explore-student/.local/share/jupyter/runtime/nbserver-15590-open.html
    Or copy and paste one of these URLs:
        http://ip-172-31-34-134:8889/?token=b301337baa0b0c5278ebc97a46752de3298d35afe75396c4
     or http://127.0.0.1:8889/?token=b301337baa0b0c5278ebc97a46752de3298d35afe75396c4
```

New Tab
http://52.210.17.182:8889/?token=b301337baa0b0c5278ebc97a46752de3298d35afe75396c4
Home Page - Select or create a n
Not secure | 52.210.17.182:8889/tree
jupyter
Files
Running
Clusters
Select items to perform actions on them.
0
/
anaconda3

EXPLORE DIGITAL SKILLS

12) Stop your Instance when Completed

When you are finished working on your machine, remember to **stop your instance.**

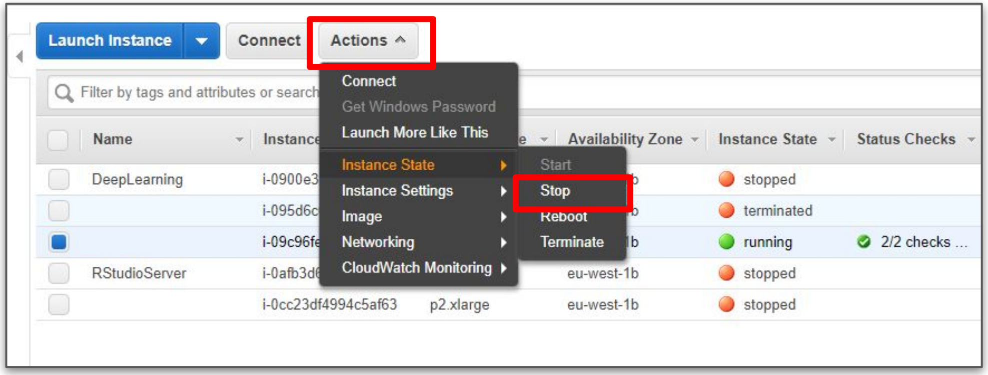
- To exit your remote session, simply type "`exit`" within the terminal.

Note, however, that this will not stop your remote machine. Instead, this action is like walking away from your computer while leaving it on. To stop it correctly, we need to use the AWS console again.

- In the EC2 Management Console, select your instance and click on "**Actions**"

- Under "**Instance State**" select "**Stop**"

- If you **do not want to use the same instance in the future**, select "**Terminate**" instead



Name	Instance	e	Availability Zone	Instance State	Status Checks
DeepLearning	i-0900e3		b	stopped	
	i-095d6c		b	terminated	
	i-09c96fe		1b	running	2/2 checks ...
RStudioServer	i-0afb3d6		eu-west-1b	stopped	
	i-0cc23df4994c5af63	p2.xlarge	eu-west-1b	stopped	

Introduction to the Cloud - Part I

Housekeeping and Review

Photo by: NASA from Unsplash

AWS Housekeeping Rules

Nothing in this world is truly free. While you're part of the Explore Academy, **you won't be charged** for any of the compute resources you use - in fact we **encourage you to be curious** and become more familiar with remote technologies.

However, because these resources are often charged on a per-hour (or even per-minute) basis, **companies you work for will often have strict rules** around the usage of cloud computing resources to avoid significant costs.

As such, we encourage you to **start developing good habits** when using remote instances for future Academy work.

These habits include:

- **Stopping your instances** when they are not in use (such as overnight or on the weekends).
- Spawning **no more than one instance at a time**.
- **Choosing an appropriate instance type** for your needs (using a big instance only if needed).
- **Correctly tagging your instance**, as motivated earlier on in the train.

AWS Housekeeping Rules - Automated Enforcement

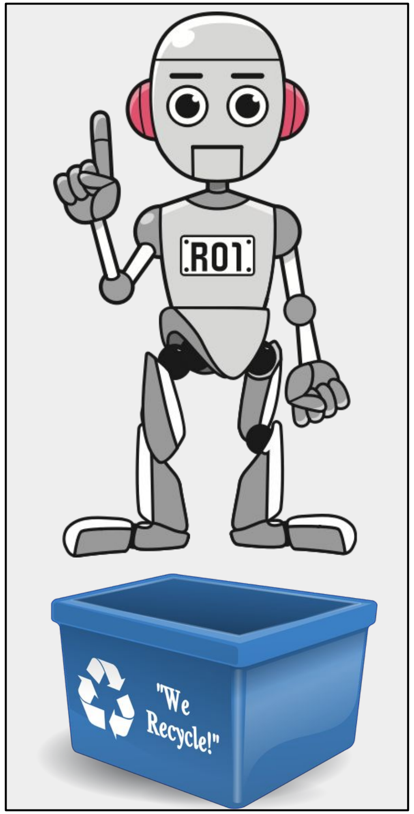
As we're all learning in the Academy, **we've created a Housecleaning bot** which will tidy up after you in AWS.

Here's how it works:

- **Every evening** the bot will check each running remote instance within the Academy.
- If it finds an **untagged instance**, it will **instantly be terminated**. (All your work is gone)
- If it finds an instance with **no housekeeping tag, it will instantly be stopped**.
- **If you have added a housekeeping tag** to your instance*, then depending on the key-value specified the **bot will respond in different ways**:

Key	Value	Description
cleanup-profile**	sandpit	Your instance will be stopped. It will then be terminated if not used for 7 consecutive days. (You are free to start it up once it's been stopped)
	daily	You are responsible for starting and stopping your EC2 instance as and when you need to make use of the service.

****Note:** It is possible to have an instance that runs overnight but you will have to get permission first. Upon getting the permission, you can be asked to change the **'cleanup-profile' tag from 'daily' to 'persistent'.**



*See step (3) for a reminder on how to do this

EXPLORE DIGITAL SKILLS

Conclusion

Phew, that was a lot!

Lets **recap** what we've learnt:

- We know that '**the cloud**' is **simply remote computing infrastructure** we can access via the internet. This helps us understand some **advantages of the cloud over traditional fixed hardware** installations.
- We have **used the AWS Console and EC2 service** to setup and **launch a remote instance**.
- We have used a **terminal to login to our remote instance**, and have launched a running Jupyter Notebook server from it.
- We've seen some **good practices and housekeeping rules** for our launched instances

EXPLORE DIGITAL SKILLS