

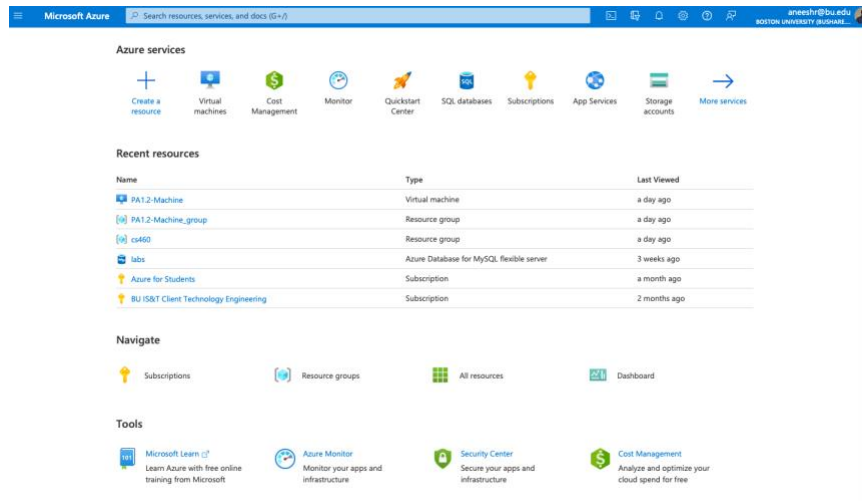
## Setting up Azure

1. Go to the following URL in your browser: <https://azure.microsoft.com/en-us/free/students/>
2. Click on the green **Start free** button and follow on-screen instructions to setup your azure account. **Remember** to use your **BU** email id to register.

## Login to Azure

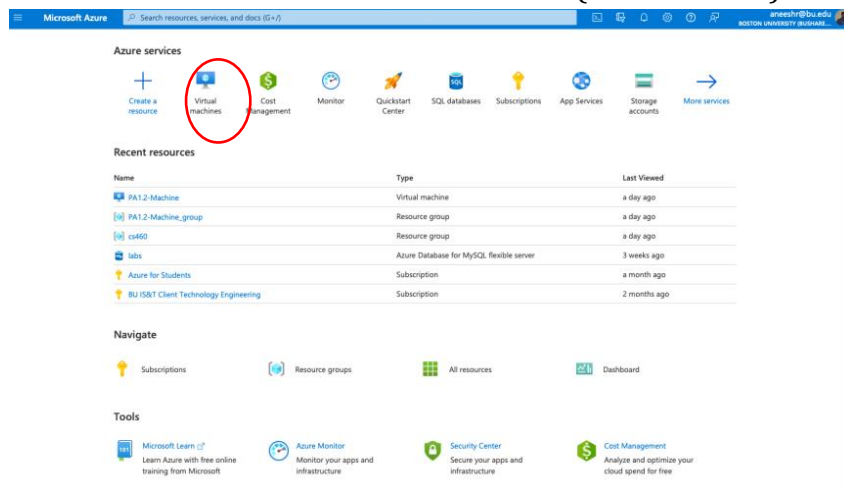
In a web browser of your choice, enter the following URL: <portal.azure.com>

You will see the dashboard as follows:

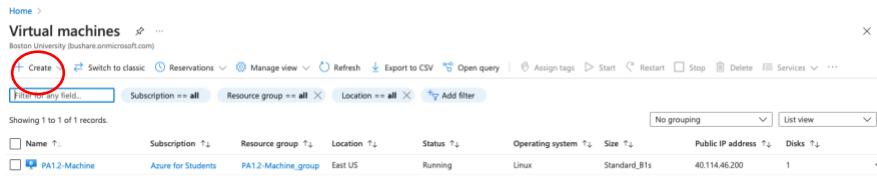


## Creating a Virtual Machine

1. Click on the Virtual Machines icon (shown below)



2. Click on the Create button. Select the **Virtual machine** option



3. You will be taken to a **Create a Virtual Machine Window**. Make sure that you choose **Azure for Students** under the Subscription.
4. Select the **cs561** resource group (you will need to create a new resource, following *these* instructions).
5. Name your virtual machine as **Project1-Machine**.
6. You may keep the defaults for Region, Availability options and Security type.
7. Under **Image**, select **Ubuntu Server 20.04 LTS – Gen2**.
8. Under size, make sure you select **Standard\_B1s – 1vcpu, 1GiB memory (\$7.59/month)**. Note, you will not be charged anything even though Azure will show you a monthly rate for the instance. This is because they maintain the same user interface for students and general users.
9. Under authentication type, select the **Password** option/radio button. You will be prompted for the username and password. For both username and password, enter *exactly* as follows:
  - a. Username: **cs561user**
  - b. Password: **Cs561project1**
10. Keep all other settings set to their defaults.
11. The create window should look as follows:

Create a virtual machine

your resources

Subscription \*

Resource group \*  [Create new](#)

Instance details

Virtual machine name \*

Region \*

Availability options

Security type

Image \*  [See all images](#) | [Configure VM generation](#)

Size \*  [See all sizes](#)

Administrator account

Authentication type ☐ SSH public key ☒ Password

Username \*

Password \*

Confirm password \*

Inbound port rules

Select which virtual machine network ports are accessible from the public internet. You can specify more limited or granular network access on the Networking tab.

☐ None

[Review + create](#) [< Previous](#) [Next: Disks >](#)

12. Click on **Review+Create**.
13. In the review window, click on **Create**. Your instance will be deployed in a few minutes.

14. Once the instance is deployed, you will be given an option for **Go to Resource**. Click on it to view your Virtual Machine instance details.

✓ Your deployment is complete

Deployment name: CreateVm-canonical.0001-com-ubuntu-server-f... Start time: 11/10/2021, 3:09:30 PM  
Subscription: [Azure for Students](#) Correlation ID: 18ba6909-0e4a-47b5-b33f-ecaba2662c11  
Resource group: [PA2-Machine\\_group](#)

Deployment details (Download)

Next steps

[Setup auto-shutdown](#) Recommended

[Monitor VM health, performance and network dependencies](#) Recommended

[Run a script inside the virtual machine](#) Recommended

[Go to resource](#)

Create another VM

## Logging into the Virtual Machine

The resource window for our created instance will look as follows:

The screenshot shows the Azure portal interface for a Virtual Machine named "PA2-Machine". The left sidebar contains navigation options like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Settings, Networking, Connect, Disks, Size, Security, Advisor recommendations, Extensions, Continuous delivery, Availability + scaling, Configuration, Identity, Properties, Locks, Operations, Bastion, Auto-shutdown, and Backup. The main content area displays the VM's status as "Running" and provides various details. The "Public IP address" is highlighted with a red circle and shows the value "13.82.4.126". Other details include the operating system "Linux (ubuntu 20.04)", size "Standard B1s", and subscription "Azure for Students".

Property	Value
Resource group	PA2-Machine_group
Status	Running
Location	East US
Subscription	Azure for Students
Subscription ID	45b23d54-8fda-47ab-a5e5-4e8561f6f222
Tags	Click here to add tags
Operating system	Linux (ubuntu 20.04)
Size	Standard B1s (1 vcpu, 1 GiB memory)
Public IP address	13.82.4.126
Virtual network/subnet	PA2-Machine_group-vnet/default
DNS name	Not configured

Property	Value
Computer name	PA2-Machine
Health state	-
Operating system	Linux (ubuntu 20.04)
Publisher	canonical
Offer	0001-com-ubuntu-server-focal
Plan	20_04-lts-gen2
VM generation	V2
Agent status	Ready
Agent version	2.5.0.2
Host group	None
Host	-
Proximity placement group	-
Colocation status	N/A
Capacity reservation group	-

Property	Value
Public IP address	13.82.4.126
Public IP address (IPv6)	-
Private IP address	10.1.0.4
Private IP address (IPv6)	-
Virtual network/subnet	PA2-Machine_group-vnet/default
DNS name	Configure

Property	Value
Size	Standard B1s
Size	Standard B1s
vCPUs	1
RAM	1 GiB

Property	Value
OS disk	PA2-Machine_OsDisk_1_80107965297d4890ac116e753417dd46
Azure disk encryption	Not enabled
Ephemeral OS disk	N/A
Data disks	0

Here, we note the public ip address that has been allocated for our resource. Note, this address might change after you stop/start the resource again. We will use the IP address listed in the resource details to login to the machine using **ssh** (like we did for login to csa1 machines). If the IP address changes, we will use the latest one.

1. Open a new terminal on a mac/linux system. If you are in a windows machine, please use PuTTY.
2. Login to the remote VM (Virtual machine) using **ssh**. Here, our username will be **cs561user** (that we listed while creating the instance) and password will be **Cs561project1**. Use this command to login from the terminal:

```
ssh cs561user@<ip_address>
```

**Remember to replace <ip\_address> with the public IP address listed for your resource.** For example, in my case, I would type the above command as follows:

```
ssh cs561user@13.82.4.126
```

3. Type the password **Cs561project1** when prompted.
4. You will now be logged into the resource and will see the following window in your terminal:

```
(base) aneeshr@dhcp-acadmin-128-197-10-176 ~ % ssh cs460user@13.82.4.126
cs460user@13.82.4.126's password:
Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.11.0-1021-azure x86_64)

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

System information as of Wed Nov 10 20:29:15 UTC 2021

System load:  0.0               Processes:    104
Usage of /:   5.2% of 28.9GB     Users logged in: 0
Memory usage: 32%              IPv4 address for eth0: 10.1.0.4
Swap usage:   0%

1 update can be applied immediately.
To see these additional updates run: apt list --upgradable

Last login: Wed Nov 10 20:13:01 2021 from 128.197.10.176
cs460user@PA2-Machine:~$
```

## Installing make

Run the following command:

```
sudo apt install make
```

## Installing gcc

Run the following command:

```
sudo apt-get install gcc
```

## PostgreSQL

### Installing PostgreSQL

We will need to install PostgreSQL into our virtual machine. To do so, in the logged in VM terminal (from above), type the following command and follow instructions on-screen:

```
sudo apt-get install postgresql
```

### Starting PostgreSQL

Enter the following commands to start the postgresql service:

1. `sudo service postgresql start`
2. `sudo su`
3. `su - postgres`
4. `createuser --interactive`

You will be prompted for the name of the role to add. Enter

**cs561user**

5. `exit`

Note, command #5 will exit postgres window.

### Connect to Postgres

Enter the following command to connect to postgres:

```
psql postgres
```

You will be taken to the Postgres terminal (like the MariaDB in the CSA machine).

### Creating a database

To create a database (called task1database), enter the following command (**remember to first connect to postgres**):

```
CREATE DATABASE task2database_psql;
```

### Exiting Postgres

Simply type the exit command as follows:

```
exit;
```

### Transferring files from Local to Remote Machine

To import files, download the [following](#) zip file and unzip it in your local system (download it in your desktop). Then, open a new terminal in **your local machine** and enter the following command:

```
scp -r file_path/project1 cs561user@<ip_address>:~/
```

For example, I would use the above command as follows since I have placed the unzipped folder in my desktop:

```
scp -r ~/Desktop/project1 cs561user@13.82.4.126:~/
```

### Switching to the right database

First, connect to postgres as described above. **Remember**, we now need to switch to the task2database\_psql (replace databasename with this). Use this command:

```
\c databasename;
```

### Sanity check

**Do this step only after setting up monetdb and running the setup script.**

Now type the following query after connecting to postgres and switching to task2database\_psql:

```
select count(*) from lineitem;
```

**You should see 600,000 as the count returned.**

### Listing Tuning knobs

In the postgres terminal (use command psql postgres), enter this command:

```
show all;
```

## MonetDB

**Login** to your virtual machine following instructions from above.

### Installing MonetDB

We will first need to create a file with a few links to download the monetdb source code.

- a. Use this command to create the file:  

```
sudo touch /etc/apt/sources.list.d/monetdb.list
```
- b. Edit the file using the **nano** command (you may also use **vim** to edit the files):  

```
sudo nano /etc/apt/sources.list.d/monetdb.list
```
- c. Paste the following contents into the file **as is**:  

```
deb https://dev.monetdb.org/downloads/deb/ focal monetdb
deb-src https://dev.monetdb.org/downloads/deb/ focal monetdb
```
- d. Save the file with the above contents following on-screen instructions (use the key combination **ctrl+o** and hit enter to save the file, **ctrl+x** to exit the editor).
2. Issue the following command to install the MonetDB GPG public key:  

```
sudo wget --output-document=/etc/apt/trusted.gpg.d/monetdb.gpg
https://www.monetdb.org/downloads/MonetDB-GPG-KEY.gpg
```
3. Check that you installed the correct file using:  

```
sudo apt-key finger
```
4. Run this command to install necessary packages:  

```
sudo apt update
```
5. Run this command to install MonetDB/SQL:  

```
sudo apt install monetdb5-sql monetdb-client
```
6. Run these commands to have system manage the service:  

```
sudo systemctl enable monetdb
sudo systemctl start monetdb
```
7. Add users to the monetdb group:  

```
sudo usermod -a -G monetdb $USER
```

### Creating a Monetdb Database

1. First, we will create a db farm. Use this command:  

```
monetdbd create cs561dbfarm
```
2. Next, we will start the db farm using this command:  

```
monetdbd start cs561dbfarm
```
3. Now, we will create the database, start and release it using these commands:  

```
monetdb create task2database_monet
monetdb start task2database_monet
monetdb release task2database_monet
```
4. Check status if running using this command:  

```
monetdb status
```

## Connecting to a Monetdb Database

1. Create a file that will setup the monetdb environment variables:
  - a. Create a file called **.monetdb** using this command:

```
touch .monetdb
```
  - b. Using a text editor of your choice (**nano or vim**), enter the following contents into this file:

```
user=monetdb
password=monetdb
```
  - c. Save the file and exit the text editor.
2. You can go into the monetdb database terminal using this command:

```
mclient -d task2database_monet
```

3. To run a query from a file, use this command (remember to replace **input\_file** and **output\_file** with actual file names):

```
mclient -d task2database_monet < input_file > output_file
```

## Setting up both databases with Data

1. Navigate to your project1 folder:

```
cd project1/
```
2. Run the following commands one after the other:

```
make clear
./setup.sh
```

## MonetDB Data Loading Issue

You may see that the lineitem table does not load correctly in Monetdb. This is because of insufficient memory (as we use the free tier in Azure for the VM). If you face this issue, follow these steps:

1. Stop the postgresql service using this command: 

```
sudo service postgresql stop
```
2. Restart the instance on the Azure VM portal (in [portal.azure.com](https://portal.azure.com))
3. Login to the VM
4. Start the Monetdb data farm using this command: 

```
monetdbd start cs561dbfarm
```
5. Navigate to the project1 folder
6. Login to the monetdb terminal using this command: 

```
mclient -d task2database_monet
```
7. Run this command: 

```
COPY INTO lineitem FROM
'/home/cs561user/project1/data/lineitem.csv' USING
DELIMITERS '|';
```
8. Exit using this command: 

```
exit
```