

Simple Student - Employer Connect Application using AWS

EC2 (Elastic Compute Cloud)

- **Purpose**
 - It provides scalable computing capacity and is designed to make web-scale cloud computing easier for developers.
- **Key-features**
 - Option for virtual computing environments with different OS, known as instances. Multiple configurations for CPU, memory, network and storage
 - Secured logins to instance using key value pair(.pem files or private key file)
 - Scaling option for users to scale up or down the resources as needed.
- **Benefits**
 - Elasticity and scalability, allowing user to scale up or down the resource with option of autoscaling.
 - Pay as you go prices for the resources used.

S3 (Simple Storage Service)

- **Purpose**
 - It is used to store and retrieve any type of data of varying sizes from anywhere on the web.
- **Key-features**
 - Reliable and scalable data storage.
 - Storing in buckets with unique names that are easy to identify.
 - Lifecycle policies and access controls.
- **Benefits**
 - Availability of data anytime and anywhere.
 - secured storage of data with access control mechanisms.
 - Easy to use UI.

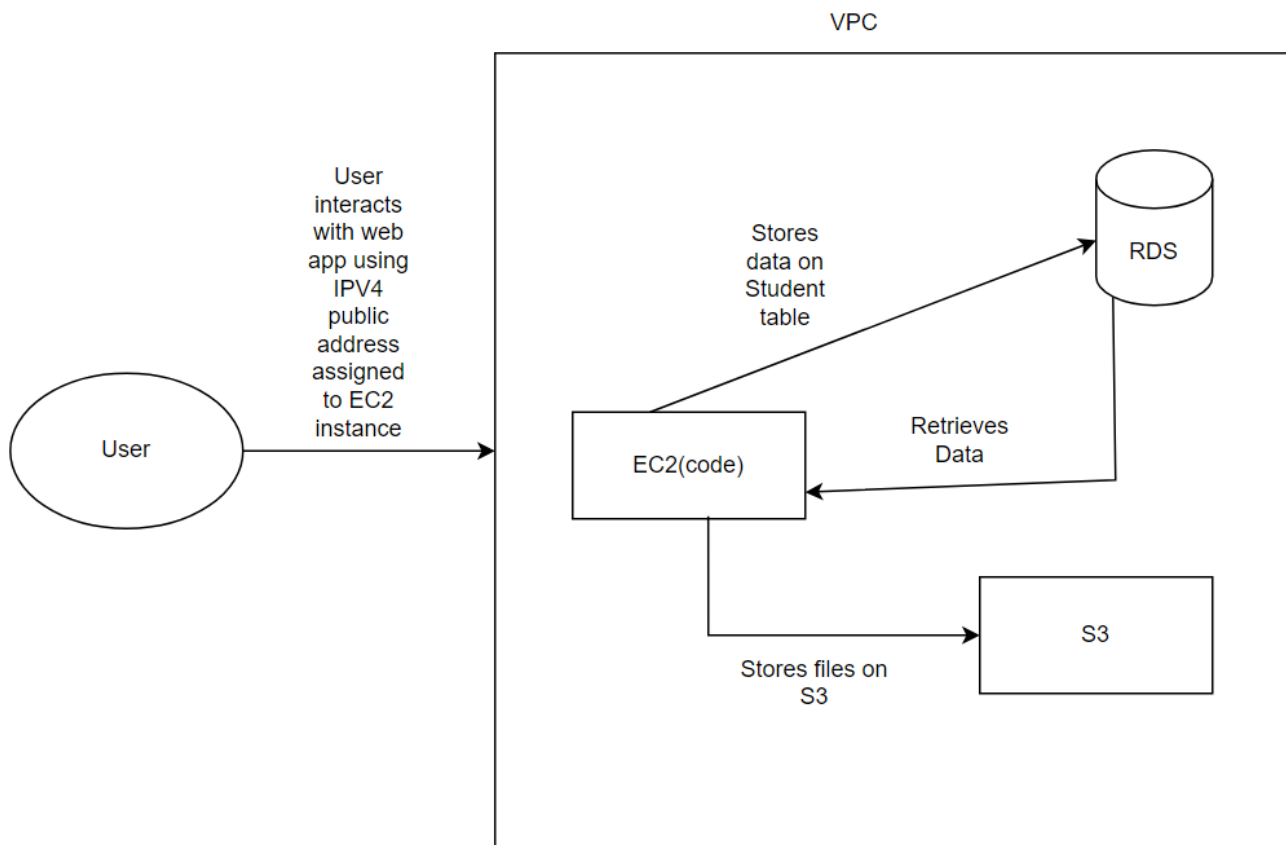
RDS (Relational Database Service)

- **Purpose**
 - It is used to setup, operate, manage various Relational Database management systems
- **Key-features**
 - Support for multiple database instances
 - High availability of data
 - Automated backups
- **Benefits**
 - Easy to setup a database
 - Autoscaling the database size
 - Scalable performance and storage

CloudFormation

- **Purpose**
 - It provides a way to model and set up AWS resources.
- **Key-features**
 - Using YAML or JSON file to provision resources
 - Managing all AWS resources in one place
 - Version control and replication of AWS infrastructure
- **Benefits**
 - User can spend less amount of time worrying about applications.
 - Streamlines the process of deploying and managing AWS resources.
 - Reduces the risk of manual errors.

Architecture design:



Configuration details:

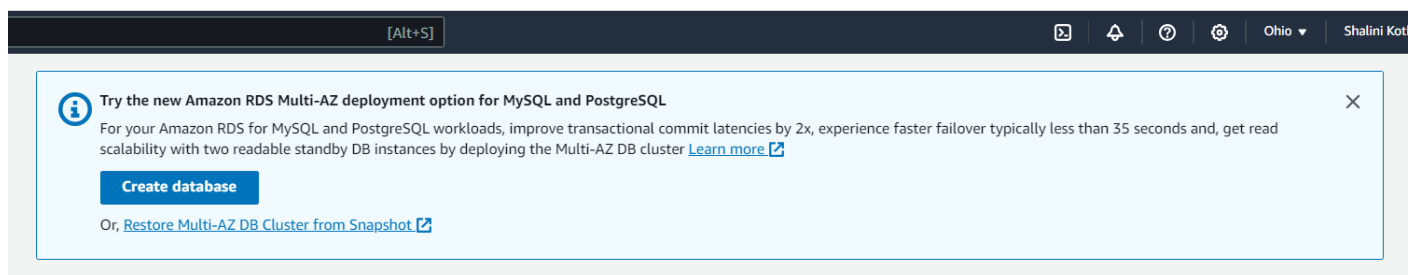
Steps:

- Step 1: Creating RDS database instance.
- Step 2: launching EC2 instance.
- Step 3: Setting up S3 bucket.
- Step 4: ssh to EC2 instance and connect to MYSQL client to create table
- Step 5: Deploying code to EC2 instance and testing the application functionality
- Step 6: Using Cloud Formation to create VPC.

I have explained the details of each step.

Step 1: Database Setup on AWS RDS

First, login to AWS management console and search for RDS. Once you open RDS services click on create database.



Enter details as below:

- Choose standard create in Choose a database creation method and engine options as MYSQL.
- Select engine version and template as free tier. In settings, give name for database instance. Enter master username and password. Later I have **changed my database name to student** as I have changed my application to student. The initial screenshots have reminder but later it was changed to student. I have changed the endpoints in web application code as well.

Region.
remider-database

The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 60 alphanumeric characters or hyphens. First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.

▼ Credentials Settings

Master username [Info](#)
Type a login ID for the master user of your DB instance.
admin
1 to 16 alphanumeric characters. The first character must be a letter.

☐ Manage master credentials in AWS Secrets Manager
Manage master user credentials in Secrets Manager. RDS can generate a password for you and manage it throughout its lifecycle.

[i](#) If you manage the master user credentials in Secrets Manager, some RDS features aren't supported. [Learn more](#)

☐ Auto generate a password
Amazon RDS can generate a password for you, or you can specify your own password.

Master password [Info](#)

Constraints: At least 8 printable ASCII characters. Can't contain any of the following: / (slash), ' (single quote), " (double quote) and @ (at sign).

Confirm master password [Info](#)

Under connectivity, select Connect to an EC2 compute resource and click on create EC2 instance as there is no EC2 instance at present. It will redirect to launch instance page in EC2 service.

Services Search [Alt+S]

Compute resource
Choose whether to set up a connection to a compute resource for this database. Setting up a connection will automatically change connectivity settings so that the compute resource can connect to this database.

☐ Don't connect to an EC2 compute resource
Don't set up a connection to a compute resource for this database. You can manually set up a connection to a compute resource later.

☒ Connect to an EC2 compute resource
Set up a connection to an EC2 compute resource for this database.

EC2 instance [Info](#)
Choose the EC2 instance to add as the compute resource for this database. A VPC security group is added to this EC2 instance. A VPC security group is also added to the database with an inbound rule that allows the EC2 instance to access the database.

Choose an EC2 instance ▼ [Refresh](#)

[i](#) No EC2 instances exist
To connect your database to an EC2 compute resource, first create an EC2 instance. [Create EC2 instance](#)

Virtual private cloud (VPC) [Info](#)
Choose the VPC. The VPC defines the virtual networking environment for this DB instance.

Default VPC (vpc-033834b7fd8a48c66)
3 Subnets, 3 Availability Zones

Only VPCs with a corresponding DB subnet group are listed.

[i](#) After a database is created, you can't change its VPC.

DB subnet group [Info](#)
Choose the DB subnet group. The DB subnet group defines which subnets and IP ranges the DB instance can use in the VPC that you selected.

☐ Choose existing
Choose existing DB subnet group

☒ Automatic setup
RDS creates a new subnet group for you or reuses an existing subnet group

Step 2: Launching EC2 instance (we will be back to step 1 after creating EC2 instance)

Services

Search

[Alt+S]

[EC2](#) > [Instances](#) > Launch an instance

Launch an instance [Info](#)

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags [Info](#)

Name

Reminder

Add additional tags

▼ Application and OS Images (Amazon Machine Image) [Info](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Search our full catalog including 1000s of application and OS images

Quick Start

Amazon Linux

aws

macOS

Mac

Ubuntu

ubuntu

Windows

Microsoft

Red Hat

Red Hat

SUSE Li

SUS

Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Amazon Linux 2023 AMI

Free tier eligible

ami-06d4b7182ac3480fa (64-bit (x86)) / ami-0090be1905998682a (64-bit (Arm))

Virtualization: hvm ENA enabled: true Root device type: ebs

Description

Amazon Linux 2023 AMI 2023.2.20231113.0 x86_64 HVM kernel-6.1

Architecture

AMI ID

Verified provider

64-bit (x86)

ami-06d4b7182ac3480fa

▼ Instance type [Info](#) | [Get advice](#)

Instance type

t2.micro

Free tier eligible

Family: t2 1 vCPU 1 GiB Memory Current generation: true

On-Demand Linux base pricing: 0.0116 USD per Hour

On-Demand SUSE base pricing: 0.0116 USD per Hour

On-Demand Windows base pricing: 0.0162 USD per Hour

On-Demand RHEL base pricing: 0.0716 USD per Hour

All generations

Compare instance types

Additional costs apply for AMIs with pre-installed software

▼ Key pair (login) [Info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - required

I am using the key value pair that was created last time for this EC2 instance as well.

Services
Search
[Alt+S]

test
Create new key pair

Network settings
Info
Edit

Network
Info

vpc-033834b7fd8a48c66

Subnet
Info

No preference (Default subnet in any availability zone)

Auto-assign public IP
Info

Enable

Firewall (security groups)
Info

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

☐ Create security group
☒ Select existing security group

Common security groups
Info

Select security groups

test_security_group sg-0bf059e9b9352a815 X

VPC: vpc-033834b7fd8a48c66

Compare security group rules

Security groups that you add or remove here will be added to or removed from all your network interfaces.

Configure storage
Info
Advanced

1x 8 GiB gp3 Root volume (Not encrypted)

Summary

Number of instances
Info

1

Software Image (AMI)

Amazon Linux 2023 AMI 2023.2.20231113.0
x86_64 HVM kernel-6.1
ami-06d4b7182ac3480fa

Virtual server type (instance type)

t2.micro

Firewall (security group)

test_security_group

Storage (volumes)

1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of

Cancel Launch instance Review commands

I used existing security group and key value pair. Now click on launch instance. Its launched successfully.

Services
Search
[Alt+S]

EC2 > Instances > Launch an instance

Success

Successfully initiated launch of instance (i-0085c4f19ccb9a679)

Launch log

Initializing requests
Succeeded

Launch initiation
Succeeded

Back to Step 1: Creating RDS database instance.
Select the EC2 instance that was created recently.

Services
Search
[Alt+S]

Connectivity
Info

Compute resource

Choose whether to set up a connection to a compute resource for this database. Setting up a connection will automatically change connectivity settings so that the compute resource can connect to this database.

☐ Don't connect to an EC2 compute resource

Don't set up a connection to a compute resource for this database. You can manually set up a connection to a compute resource later.

☒ Connect to an EC2 compute resource

Set up a connection to an EC2 compute resource for this database.

EC2 instance

Choose the EC2 instance to add as the compute resource for this database. A VPC security group is added to this EC2 instance. A VPC security group is also added to the database with an inbound rule that allows the EC2 instance to access the database.

i-0085c4f19ccb9a679
Reminder

Some VPC settings can't be changed when a compute resource is added

Adding an EC2 compute resource automatically selects the VPC, DB subnet group, and public access settings for this database. To allow the EC2 instance to access the database, a VPC security group rds-ec2-X is added to the database and another called ec2-rds-X to the EC2 instance. You can remove the new security group for the database only by removing the compute resource.

Virtual private cloud (VPC)

Choose the VPC. The VPC defines the virtual networking environment for this DB instance.

Default VPC (vpc-033834b7fd8a48c66)
3 Subnets, 3 Availability Zones

Only VPCs with a corresponding DB subnet group are listed.

After a database is created, you can't change its VPC.

DB subnet group

Choose the DB subnet group. The DB subnet group defines which subnets and IP ranges the DB instance can use in the VPC that you selected.

I have change public access to yes later while working on deploying website.

DB subnet group

Choose the DB subnet group. The DB subnet group defines which subnets and IP ranges the DB instance can use in the VPC that you selected.

☐ Choose existing

Choose existing DB subnet group

☒ Automatic setup

RDS creates a new subnet group for you or reuses an existing subnet group

DB subnet group name

rds-ec2-db-subnet-group-1

New DB subnet group created.

Public access

☒ Yes

RDS assigns a public IP address to the database. Amazon EC2 instances and other resources outside of the VPC can connect to your database. Resources inside the VPC can also connect to the database. Choose one or more VPC security groups that specify which resources can connect to the database.

☐ No

RDS doesn't assign a public IP address to the database. Only Amazon EC2 instances and other resources inside the VPC can connect to your database. Choose one or more VPC security groups that specify which resources can connect to the database.

VPC security group (firewall)

Choose one or more VPC security groups to allow access to your database. Make sure that the security group rules allow the appropriate incoming traffic.

☒ Choose existing

Choose existing VPC security groups

☐ Create new

Create new VPC security group

Additional VPC security group

Choose one or more options

test_security_group

Amazon RDS will add a new VPC security group rds-ec2-1 to allow connectivity with your compute resource.

Availability Zone [Info](#)

us-east-2a ▼

Certificate authority - *optional* [Info](#)

Using a server certificate provides an extra layer of security by validating that the connection is being made to an Amazon database. It does so by checking the server certificate that is automatically installed on all databases that you provision.

rds-ca-2019 (default)
Expiry: Aug 22, 2024 ▼

If you don't select a certificate authority, RDS chooses one for you.

► Additional configuration

Database authentication

Database authentication options [Info](#)

☒ Password authentication
Authenticates using database passwords.

☐ Password and IAM database authentication
Authenticates using the database password and user credentials through AWS IAM users and roles.

☐ Password and Kerberos authentication
Choose a directory in which you want to allow authorized users to authenticate with this DB instance using Kerberos Authentication.

Monitoring

☐ Enable Enhanced monitoring
Enabling Enhanced monitoring metrics are useful when you want to see how different processes or threads use the CPU.

After selecting options as above click on create database at the bottom right corner of current webpage which creates RDS database instance.

☑ Successfully set up a connection between reminder-database and EC2 instance i-0085c4f19ccb9a679 [🔗](#)

☑ Successfully created database reminder-database [View connection details](#)

You can use settings from reminder-database to simplify configuration of [suggested database add-ons](#) while we finish creating your DB for you.

[RDS](#) > Databases

Consider creating a Blue/Green Deployment to minimize downtime during upgrades
You may want to consider using Amazon RDS Blue/Green Deployments and minimize your downtime during upgrades. A Blue/Green Deployment provides a staging environment for changes to production databases. [RDS User Guide](#) [Aurora User Guide](#)

Databases (1) Group resources Modify Actions ▼ Restore from S3 Create database

Filter by databases

DB Identifier ▲	Status ▼	Role ▼	Engine ▼	Region & AZ ▼	Size ▼	Actions ▼	CPU ▼	Current activity ▼	Maintenance ▼	VPC
reminder-database	Back-up	Instance	MySQL Community	us-east-2a	db.t3.micro	-	-		none	vpc-033834b7fd8a4t

My database connection details are as below.

Endpoint: reminder-database.ctdtdzret8pt.us-east-2.rds.amazonaws.com.

Later It was changed to student.ctdtdzret8pt.us-east-2.rds.amazonaws.com as I changed my database name.

Connection details to your database reminder-database

×

This is the only time you can view this password. Copy and save the password for your reference. If you lose the password, you must modify your database to change it. You can use a SQL client application or utility to connect to your database.

[Learn about connecting to your database](#)

Master username
admin

Master password

Copy

Endpoint
reminder-database.ctdtdzret8pt.us-east-2.rds.amazonaws.com

Copy

Close

Now I am creating a new IAM role for my EC2 instance. Go to role in AWS management console and create role. I created new IAM role 'EC2_S3_AccessRole' as below

Name, review, and create

Role details

Role name

Enter a meaningful name to identify this role.

EC2_S3_AccessRole

Maximum 64 characters. Use alphanumeric and '+', '=', '@', '-', '_' characters.

Description

Add a short explanation for this role.

Allows EC2 instances to call AWS services on your behalf.

Maximum 1000 characters. Use alphanumeric and '+', '=', '@', '-', '_' characters.

Step 1: Select trusted entities

Trust policy

```

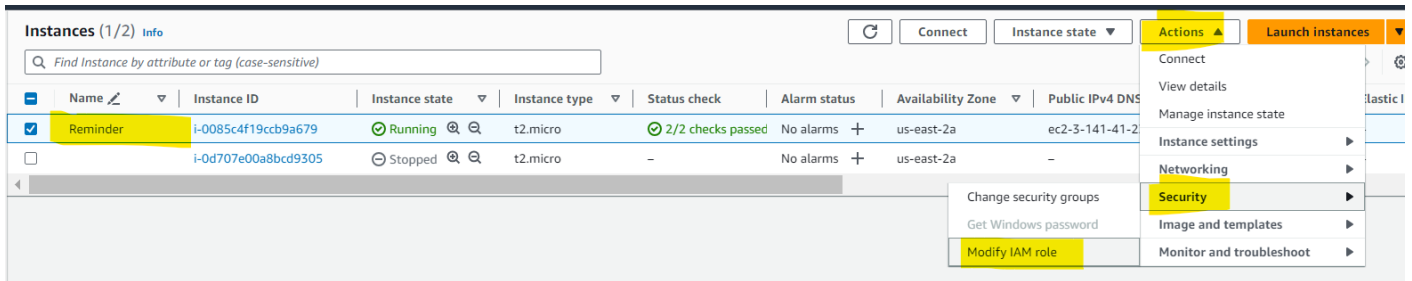
1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Effect": "Allow",
6       "Action": [
7         "sts:AssumeRole"
8       ],
9       "Principal": {
10        "Service": [
11          "ec2.amazonaws.com"
12        ]
13      }
14    }
15  ]
16 }
```

Step 2: Add permissions

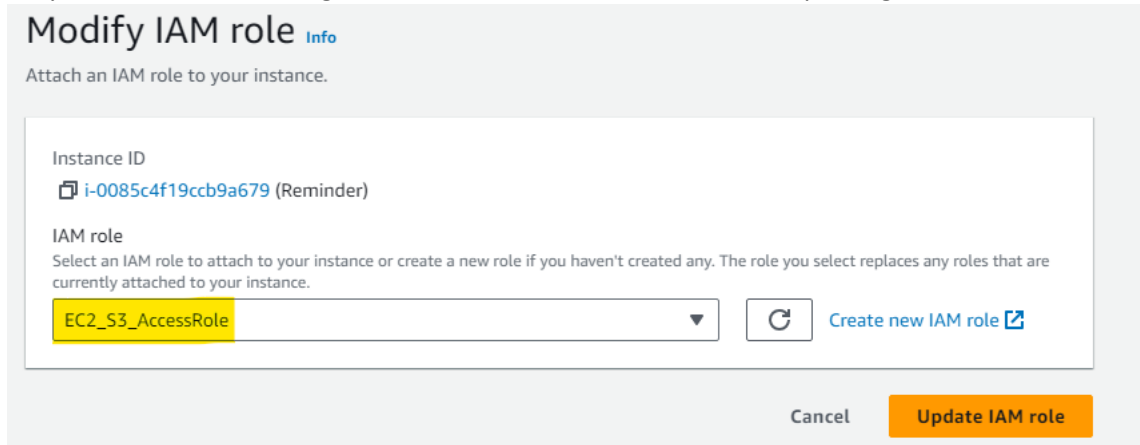
Permissions policy summary

Policy name	Type	Attached as
AmazonS3FullAccess	AWS managed	Permissions policy

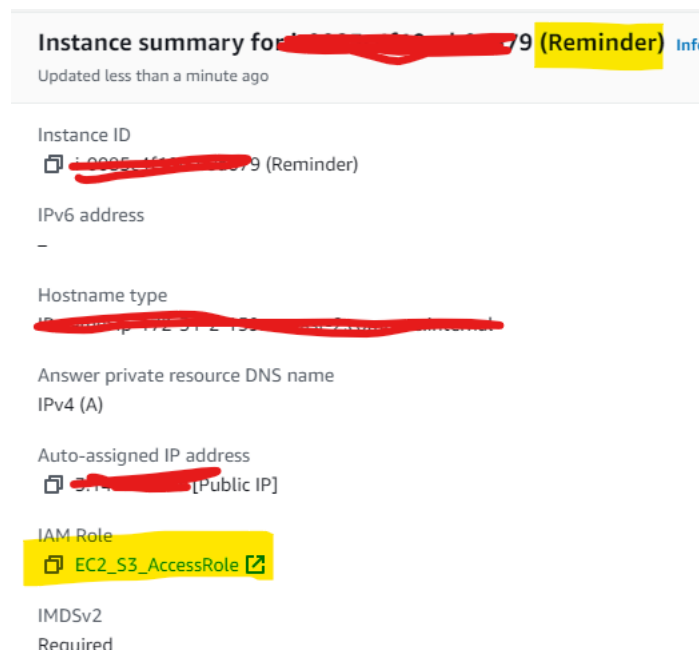
Now, lets attach role to EC2 instance. Go select EC2 instance and click on Action tab> security> Modify IAM Role.



Click on modify IAM role and selecting the role that was created for EC2 and updating the IAM role.



Now, this role is attached to EC2 instance Reminder



Step 3: Setting up S3 bucket

I am using already existing s3 bucket s3bucketshalini which was created as part of last AWS assignment. But I have to update permissions. Select the S3 bucket and under permissions go to bucket policy and update policy as below and save changes. This provides the necessary permissions for my application to read from and write to the S3 bucket.

```
arn:aws:s3:::s3bucketshalini
```

```
1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Effect": "Allow",
6       "Principal": {
7         "AWS": "arn:aws:iam::274082713924:role/EC2_S3_AccessRole"
8       },
9       "Action": [
10        "s3:GetObject",
11        "s3:PutObject",
12        "s3:DeleteObject"
13      ],
14       "Resource": "arn:aws:s3:::s3bucketshalini/*"
15     }
16   ]
17 }
```

Now, connect to EC2 instance using putty and .pem file that was downloaded when creating key-value pair.



```

[ec2-user@ip-172-31-2-159 ~]$ curl -O https://bootstrap.pypa.io/get-pip.py
% Total    % Received % Xferd Average Speed   Time    Time     Time  Current
           Dload  Upload   Total   Spent    Left   Speed
100 2570k  100 2570k    0     0  16.8M    0 --:--:-- --:--:-- --:--:-- 16.9M
[ec2-user@ip-172-31-2-159 ~]$ sudo python3 get-pip.py
Collecting pip
  Downloading pip-23.3.1-py3-none-any.whl.metadata (3.5 kB)
Collecting wheel
  Downloading wheel-0.42.0-py3-none-any.whl.metadata (2.2 kB)
Downloading pip-23.3.1-py3-none-any.whl (2.1 MB)
----- 2.1/2.1 MB 21.8 MB/s eta 0:00:00
Downloading wheel-0.42.0-py3-none-any.whl (65 kB)
----- 65.4/65.4 kB 9.5 MB/s eta 0:00:00
Installing collected packages: wheel, pip
Successfully installed pip-23.3.1 wheel-0.42.0
WARNING: Running pip as the 'root' user can result in broken permissions and con
flicting behaviour with the system package manager. It is recommended to use a v
irtual environment instead: https://pip.pypa.io/warnings/venv
[ec2-user@ip-172-31-2-159 ~]$ pip3 --version
pip 23.3.1 from /usr/local/lib/python3.9/site-packages/pip (python 3.9)
[ec2-user@ip-172-31-2-159 ~]$ python3 --version
Python 3.9.16
[ec2-user@ip-172-31-2-159 ~]$ pip3 install flask
Defaulting to user installation because normal site-packages is not writeable
Collecting flask
  Downloading flask-3.0.0-py3-none-any.whl.metadata (3.6 kB)
Collecting Werkzeug>=3.0.0 (from flask)
  Downloading werkzeug-3.0.1-py3-none-any.whl.metadata (4.1 kB)
Collecting Jinja2>=3.1.2 (from flask)
  Downloading Jinja2-3.1.2-py3-none-any.whl (133 kB)
----- 133.1/133.1 kB 3.3 MB/s eta 0:00:00
Collecting itsdangerous>=2.1.2 (from flask)
  Downloading itsdangerous-2.1.2-py3-none-any.whl (15 kB)
Collecting click>=8.1.3 (from flask)
  Downloading click-8.1.7-py3-none-any.whl.metadata (3.0 kB)
Collecting blinker>=1.6.2 (from flask)
  Downloading blinker-1.7.0-py3-none-any.whl.metadata (1.9 kB)
Collecting importlib-metadata>=3.6.0 (from flask)
  Downloading importlib_metadata-7.0.0-py3-none-any.whl.metadata (4.9 kB)
Collecting zipp>=0.5 (from importlib-metadata>=3.6.0->flask)
  Downloading zipp-3.17.0-py3-none-any.whl.metadata (3.7 kB)
Collecting MarkupSafe>=2.0 (from Jinja2>=3.1.2->flask)
  Downloading MarkupSafe-2.1.3-cp39-cp39-manylinux_2_17_x86_64.manylinux2014_x86

```

Another way connect to EC2 instance using connect button on the details page of EC2 instance. Both ways works using putty and using connect in ec2 instance.

The screenshot shows the AWS Management Console interface for an EC2 instance. The instance ID is i-0085c4f19ccb9a679. The 'Connect' button is highlighted in yellow. The instance state is 'Available'.

I have uploaded my web app files to github. Now I am going to use that repository to clone files on EC2 instance.

```

[ec2-user@ip-172-31-2-159 ~]$ git clone https://github.iu.edu/skothuru/bda_simple_web_app.git
Cloning into 'bda_simple_web_app'...
Username for 'https://github.iu.edu': skothuru
Password for 'https://skothuru@github.iu.edu':
remote: Enumerating objects: 11, done.
remote: Counting objects: 100% (11/11), done.
remote: Compressing objects: 100% (9/9), done.
remote: Total 11 (delta 0), reused 8 (delta 0), pack-reused 0
Receiving objects: 100% (11/11), done.
[ec2-user@ip-172-31-2-159 ~]$ cd bda_simple_web_app/

```

I renamed by database instance from reminder-database to student since I changed my application.

The screenshot shows the AWS RDS console interface. The 'Databases (1)' section is visible. A search filter 'Filter by databases' is present. The table below shows the database details:

DB identifier	Status	Region
student	Available	us-east-1

I have installed MySQL command line client using following command 'sudo dnf update -y' and I have connected to database instance using command. Next step is to create a database student

```
[ec2-user@ip-172-31-2-159 ~]$ mysql -h student.ctdtdzret8pt.us-east-2.rds.amazonaws.com -P 3306 -u admin -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 97
Server version: 8.0.33 Source distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]> create database student;
Query OK, 1 row affected (0.017 sec)

MySQL [(none)]> use student
Database changed
```

I have created table student in database student as show in below picture.

```
mysql> create table student(
-> email varchar(255) primary key,
-> first_name varchar(255),last_name varchar(255),mobile varchar(10),college varchar(255),major varchar(255),graduation_year YEAR,skills TEXT,
-> area_of_interest TEXT);
Query OK, 0 rows affected (0.03 sec)

mysql> show tables
-> ;
+-----+
| Tables_in_student |
+-----+
| student           |
+-----+
1 row in set (0.00 sec)
```

Step 5: Deploying code to EC2 instance and testing the application functionality

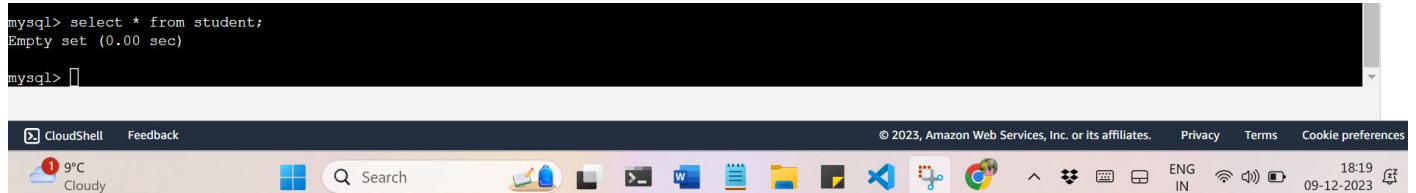
Next step is deploying the code in EC2 instance. I have developed code locally and pushed that code to GitHub and later cloned it on EC2 instance.

Testing functionality of my web application:

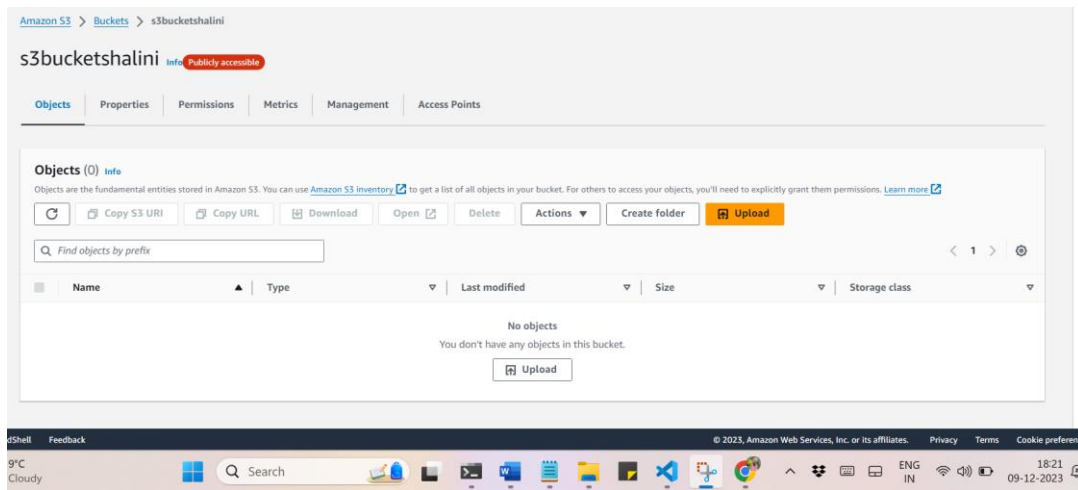
I have provided screenshots with timestamps for reference

Initially I don't have any records in my table student and no files in S3 bucket s3bucketshalini. Please refer screenshot with timestamp

Tables with 0 rows



S3 bucket with no files



This is my public IP address of EC2 instance. I am going to connect to using <http://34.238.169.24/> when I run my app.py file using python3 app.py

Below screen is my main portal



Welcome to student portal!!

Add Student Details

View Student Details



On clicking add student details and user get a chance to enter student details

Not secure | 34.238.169.24/add_student

Add Student Details

Email
skothuru@iu.edu

First Name
Shalini

Last Name
Kothuru

Mobile
8123456789

College
Indiana University

Major
Computer Science

Graduation Year
2023

Skills
Python, C, HL

Area of Interest
Data Science

Resume
Choose File Final list_CHC.pdf

Add Student

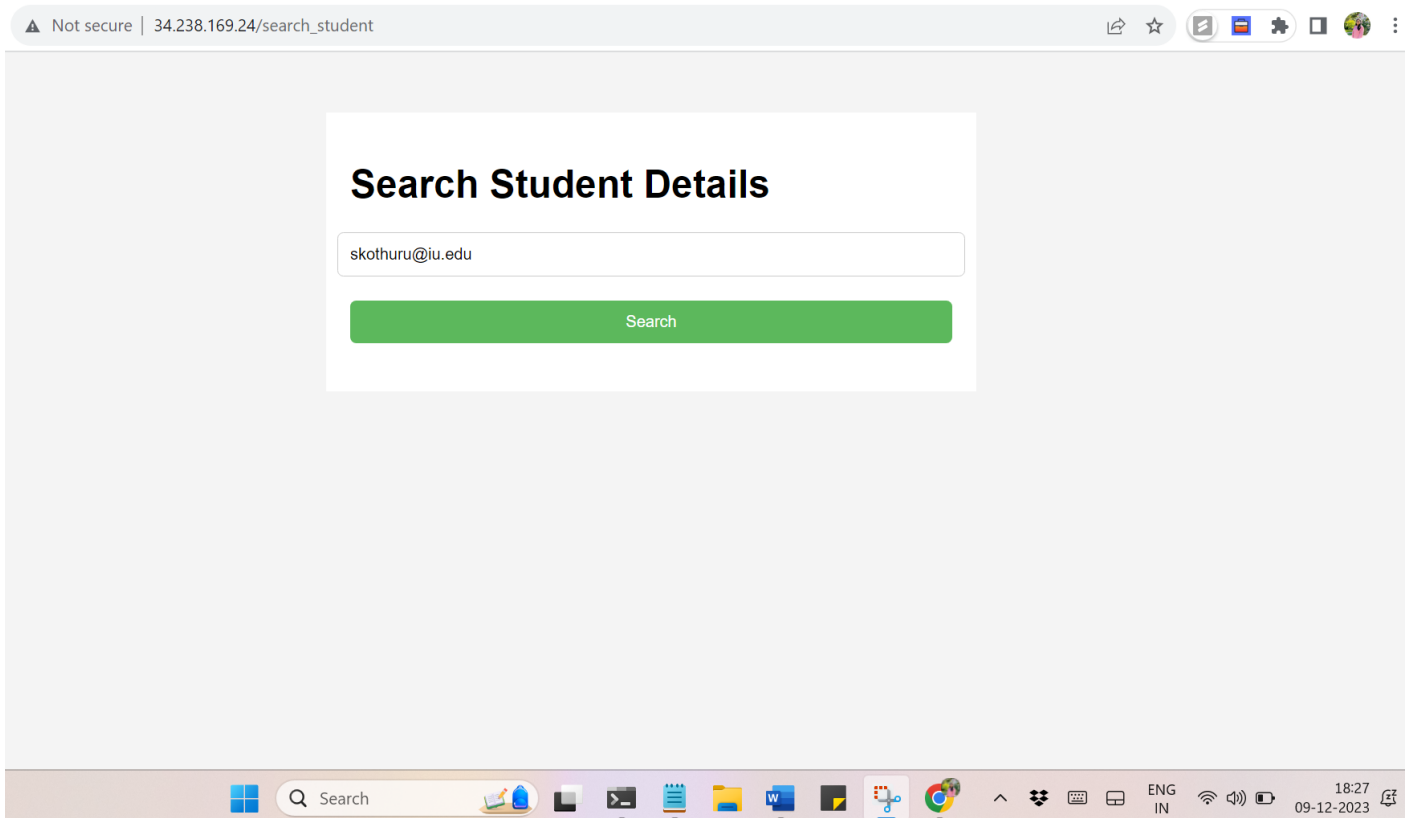
Windows taskbar: Search, File Explorer, Microsoft Word, Google Chrome, 18:25 09-12-2023

Once user completes filling details, on successful entry to database, it gives firstname's details entered in our database successfully. Since firstname was Shalini, it mentioned Shalini

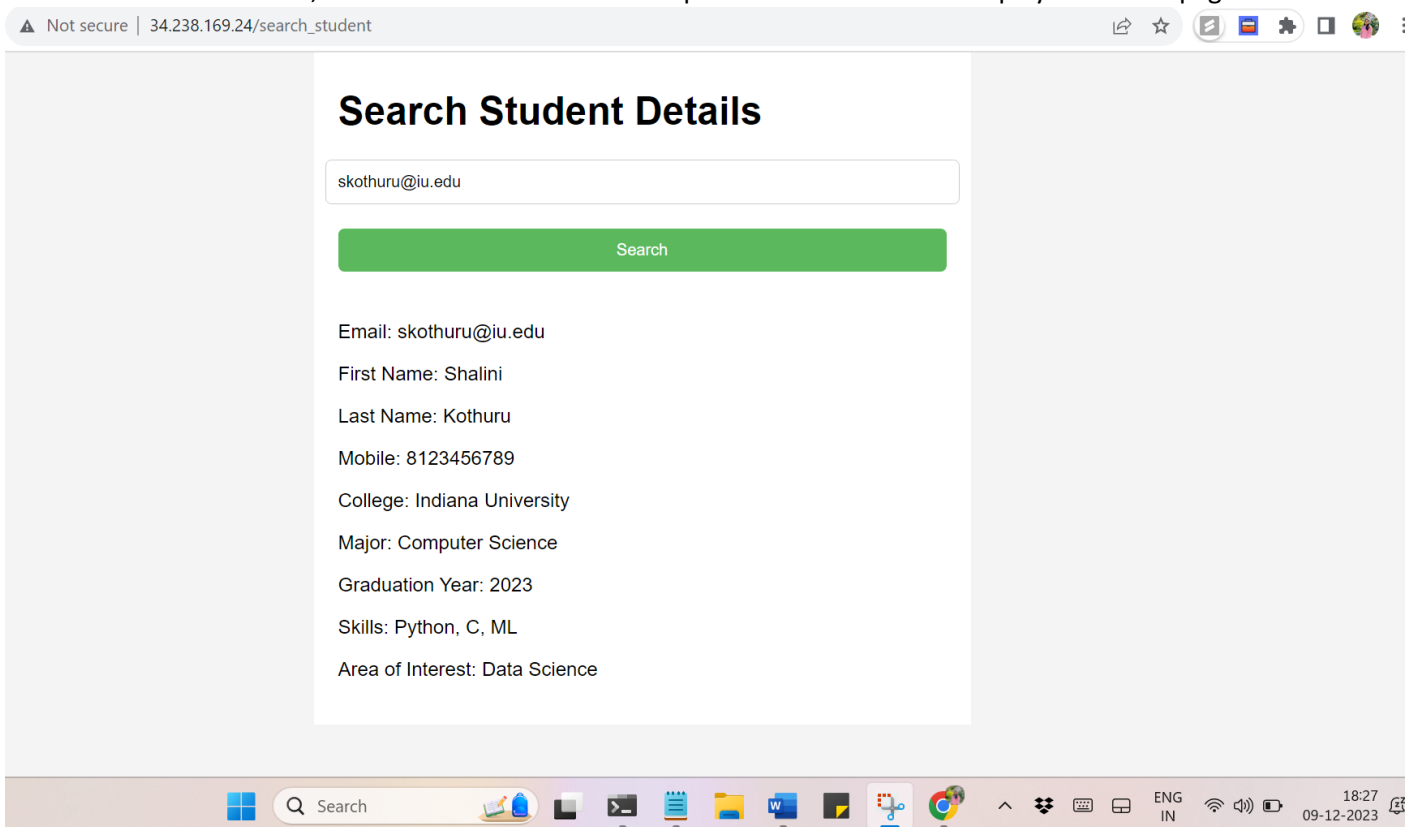
Not secure | 34.238.169.24/add_student

Shalini's details entered in our database successfully!

On clicking view student details in main page, it will redirect to below page



Once user enters email ID, the details associated with respective email ID will be displayed on web page.



Let's check database records and S3 bucket. Initially there was no record in database but after add student through web application, we can see a record is inserted with same details.

```
mysql> select * from student;
Empty set (0.00 sec)

mysql> select * from students
-> ;
ERROR 1146 (42S02): Table 'student.students' doesn't exist
mysql> select * from student;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| email          | first_name | last_name | mobile | college      | major      | graduation_year | skills          | area_of_interest |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| skothuru@iu.edu | Shalini    | Kothuru   | 8123456789 | Indiana University | Computer Science | 2023 | Python, C, ML | Data Science |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql>
```

CloudShell Feedback © 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 9°C Cloudy 18:28 09-12-2023

In S3 bucket, you can see file was added. In web application, I added logic to rename the file with emailID_resume so the file is stored in S3 as shown in picture. You can see timestamp, it matches with time when query was executed.

Amazon S3 > Buckets > s3bucketshalini

s3bucketshalini Info Publicly accessible

Objects (1) Info

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

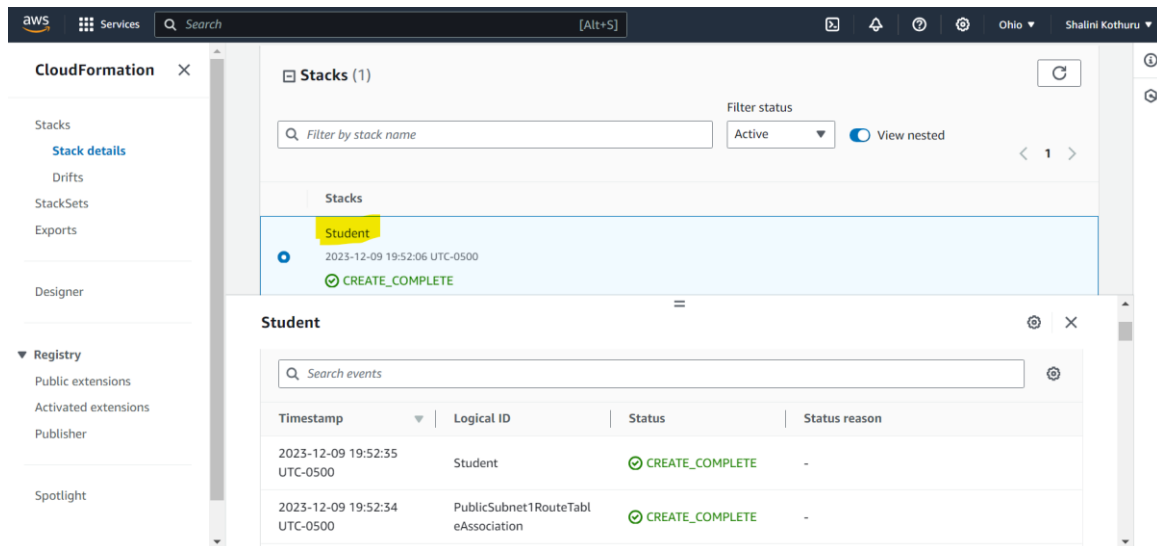
Refresh Copy S3 URI Copy URL Download Open Delete Actions Create folder Upload

<input type="checkbox"/>	Name	Type	Last modified	Size	Storage class
<input type="checkbox"/>	skothuru@iu.edu_resume	edu_resume	December 9, 2023, 18:25:44 (UTC-05:00)	87.3 KB	Standard

© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 18:29 09-12-2023

Step 6: Using Cloud Formation to create VPC.

Go to Cloud Formation in AWS and click on create stack. I have uploaded yaml file and named it student. As you can see, it was created successfully.



Explain the benefits and advantages of using AWS Compute, Storage, Database, and Infrastructure Management Services for your scenario.

- RDS: AWS RDS helped me in simplifying the setup, operation, and scaling of the relational database for the student web application. Unlike traditional databases which requires extensive planning for each phase, AWS RDS is easy to use. Using AWS RDS, it easy to backup and create. I can simply rename the database which changes its name everywhere where it was used.
- EC2: I can choose the operating system in EC2. For hosting different applications, I can create different EC2 instance with different configurations. I tried my application on Linux and Ubuntu when I had issue using one OS. There is option to autoscale applications. Although the application doesn't need autoscaling now, but its very useful.
- S3: I used it for storing the resume files that was uploaded by user. Storing it on local machines may lead to access problems but storing it on S3 buckets, it accessible on any bucket.
- CloudFormation: It allowed me to model and setup up AWS resources. I have the option of automating the deployment and updating the infrastructure which makes it easy to use the infrastructure.

Reflect on the challenges and considerations encountered during the implementation process and discuss how AWS services addressed those challenges.

- I had issue with using one OS, then I just deleted that instance and created another one in just a minute.
- Since I change my thought of doing different application after setting up some part of AWS configuration, it was easy for me to rename the objects to match according to my application.