

Saurabh Sanjiv Jadhav

AWS Mini Project.

Database Backup Automation: EC2 to S3

Key Points :

- Take a backup file of database from ec2-instance to s3 bucket.
- Ensures secure, scalable, and reliable backup storage.
- Simplifies data management with scheduled transfers.

1. Create Ec2-instance.

The screenshot shows the 'Launch an instance' wizard in the AWS EC2 console. The 'Name and tags' section has 's3-Backup-instance' entered. The 'Application and OS Images (Amazon Machine Image)' section shows 'Amazon Linux 2023 AMI 2023.6.2...' selected. The 'Summary' panel on the right shows 1 instance being launched, with 'Launch instance' and 'Preview code' buttons. The bottom navigation bar includes CloudShell, Feedback, and various system icons.

2.Launch instance.

The screenshot shows the 'Launch an instance' wizard in the AWS EC2 console. The 'Configure storage' section shows a 1x 8 GiB gp3 root volume. A tooltip indicates free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. The 'Summary' panel on the right shows 1 instance being launched, with 'Launch instance' and 'Preview code' buttons. The bottom navigation bar includes CloudShell, Feedback, and various system icons.

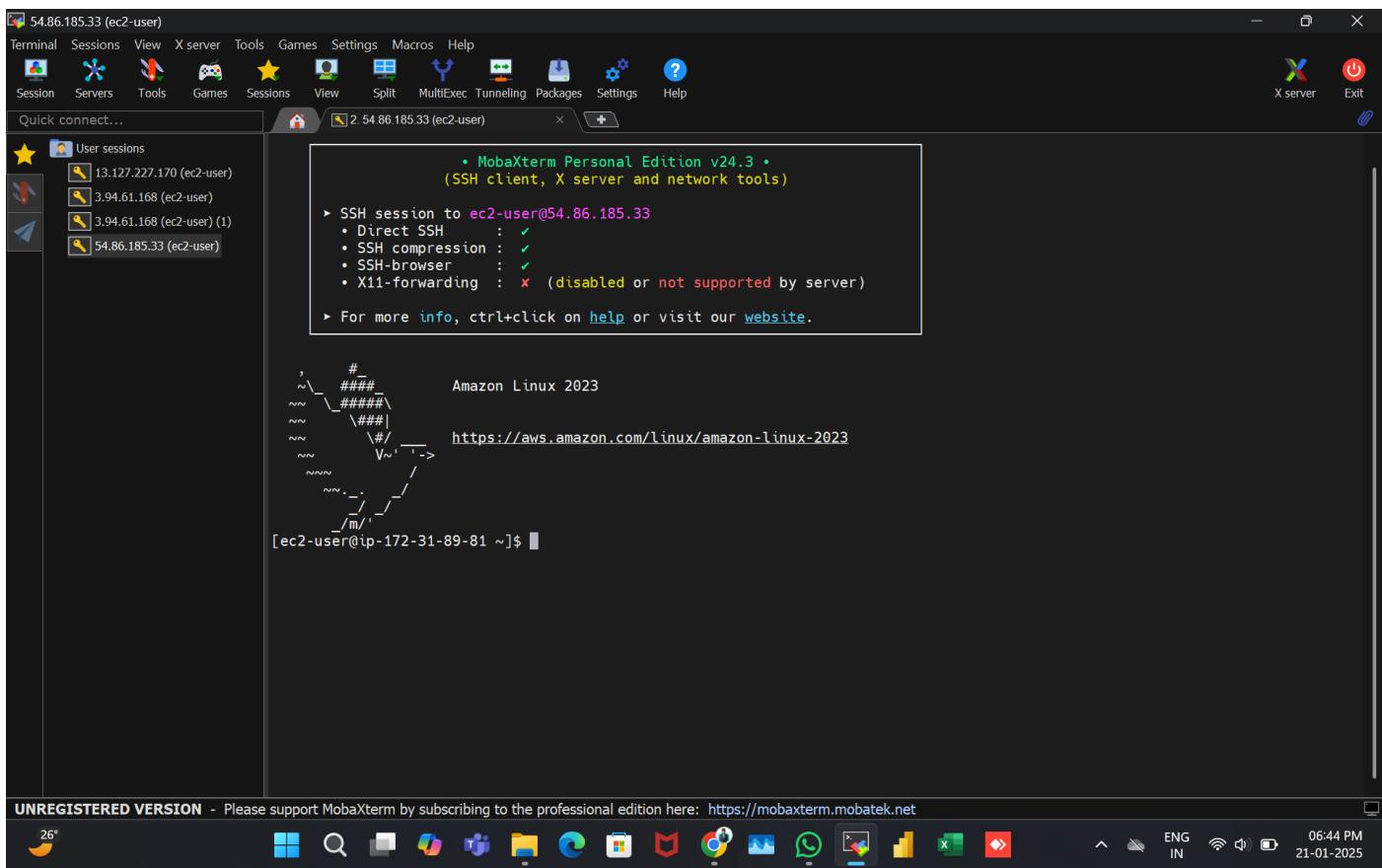
3.Successfully created the instance .

The screenshot shows the AWS EC2 Instances page. On the left sidebar, under the 'Instances' section, there is a list of options: Dashboard, EC2 Global View, Events, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images, AMIs, and AMI Catalog. Under 'Elastic Block Store', there are Volumes and CloudShell. The main content area displays 'Instances (1/1)'. A search bar at the top says 'Find Instance by attribute or tag (case-sensitive)' with the value 'Instance ID = i-0435ebbedd442defa4'. Below the search bar are buttons for 'Connect', 'Actions', and 'Launch instances'. A table lists one instance: 's3-Backup-ins...' with ID 'i-0435ebbedd442defa4', status 'Running', type 't2.micro', and a note 'Initializing'. To the right of the table are buttons for 'View alarms' and a gear icon. Below the table, the instance details for 'i-0435ebbedd442defa4 (s3-Backup-instance)' are shown, including tabs for Details, Status and alarms, Monitoring, Security, Networking, Storage, and Tags. The 'Details' tab is selected. Under 'Instance summary', it shows the Instance ID (i-0435ebbedd442defa4), Public IPv4 address (54.86.185.33), Private IPv4 addresses (172.31.89.81), and Instance state (running). The bottom of the screen shows the AWS navigation bar and a taskbar.

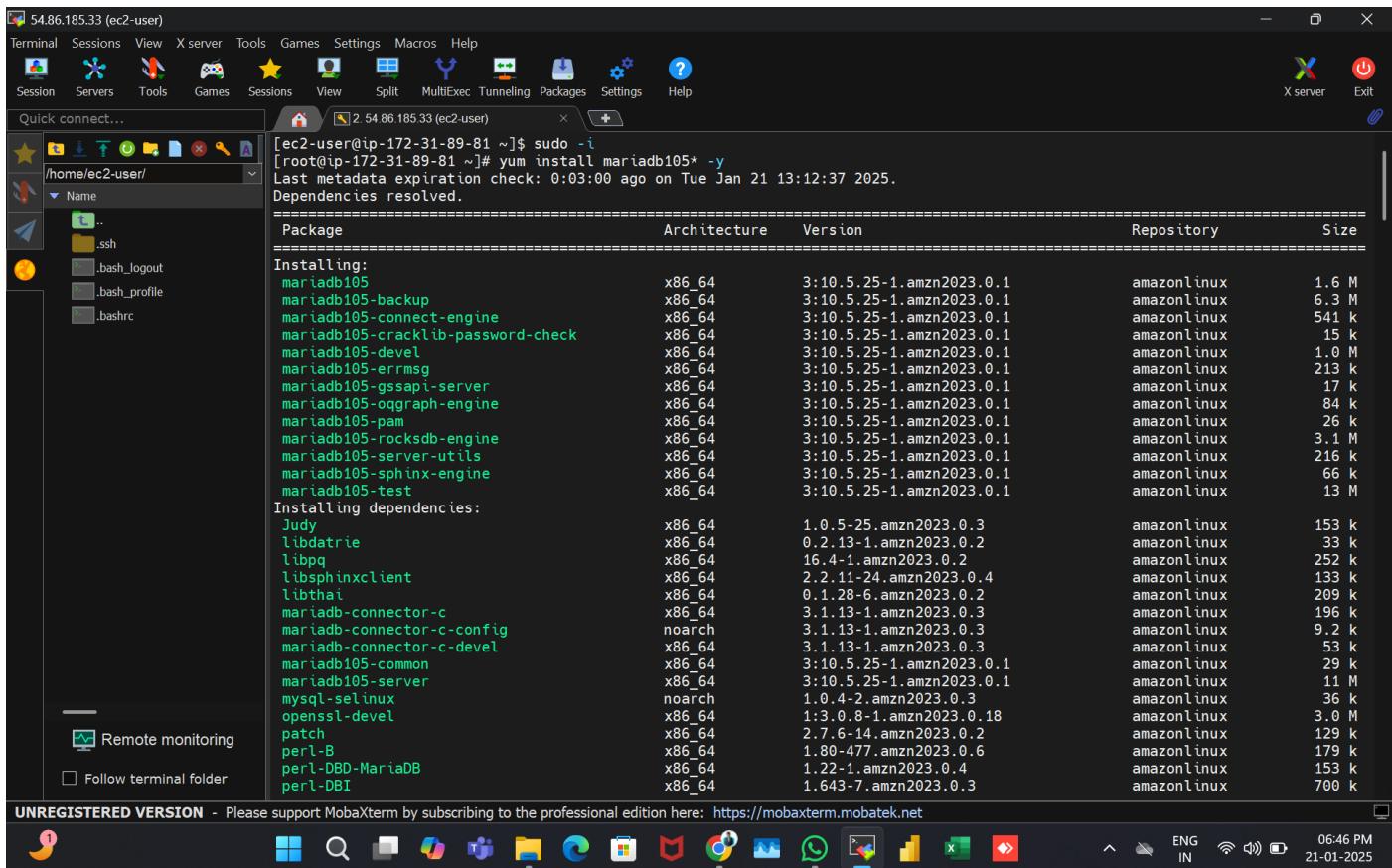
4.Add Security group and add (MYSQL/Aurora).

The screenshot shows the AWS EC2 Security Groups page. In the top navigation bar, under 'EC2 > Security Groups > sg-0c5cca548a4a35b69 - launch-wizard-6 > Edit inbound rules', the 'Edit inbound rules' section is highlighted. It shows two existing rules: one for SSH (TCP port 22) and another for MySQL/Aurora (TCP port 3306). Both rules have a source of '0.0.0.0/0'. A warning message at the bottom states: '⚠️ Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.' The bottom of the screen shows the AWS navigation bar and a taskbar.

5.Get SSH of your ec2-instance



6. Switch to root user and install the mariadb.



7. Start the mariadb.

The screenshot shows the MobaXterm interface. The title bar indicates the session is 54.86.185.33 (ec2-user). The terminal window displays the command: [root@ip-172-31-89-81 ~]# systemctl start mariadb.service followed by a prompt [root@ip-172-31-89-81 ~]#. The left sidebar shows the file structure of the user's home directory, including .ssh, .bash_logout, .bash_profile, and .bashrc. The bottom status bar shows the date and time as 21-01-2025, 06:47 PM, and the system language as ENG IN.

```
[root@ip-172-31-89-81 ~]# systemctl start mariadb.service
[root@ip-172-31-89-81 ~]#
```

8. Assign password for database

The screenshot shows the MobaXterm interface. The title bar indicates the session is 54.86.185.33 (ec2-user). The terminal window displays the mysql_secure_installation command: [root@ip-172-31-89-81 ~]# mysql_secure_installation. The output of the script is shown, including prompts for the root password and the confirmation of successful execution. The left sidebar shows the file structure of the user's home directory. The bottom status bar shows the date and time as 21-01-2025, 06:49 PM, and the system language as ENG IN.

```
[root@ip-172-31-89-81 ~]# mysql_secure_installation
NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB
      SERVERS IN PRODUCTION USE!  PLEASE READ EACH STEP CAREFULLY!

In order to log into MariaDB to secure it, we'll need the current
password for the root user. If you've just installed MariaDB, and
haven't set the root password yet, you should just press enter here.

Enter current password for root (enter for none):
OK, successfully used password, moving on...

Setting the root password or using the unix_socket ensures that nobody
can log into the MariaDB root user without the proper authorisation.

You already have your root account protected, so you can safely answer 'n'.

Switch to unix_socket authentication [Y/n]
Enabled successfully!
Reloading privilege tables..
... Success!

You already have your root account protected, so you can safely answer 'n'.

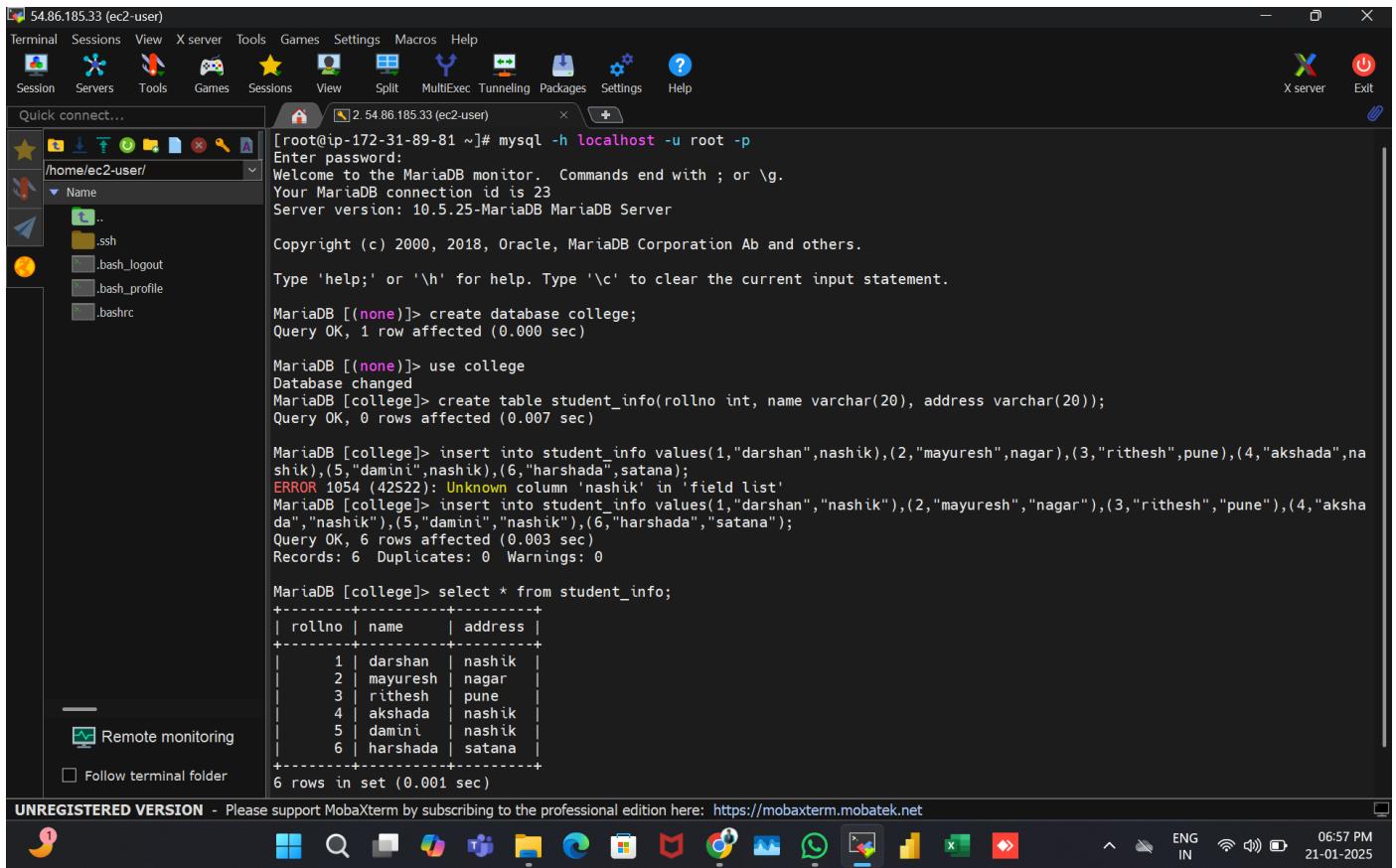
Change the root password? [Y/n]
New password:
Re-enter new password:
Password updated successfully!
Reloading privilege tables..
... Success!

By default, a MariaDB installation has an anonymous user, allowing anyone
to log into MariaDB without having to have a user account created for
them. This is intended only for testing, and to make the installation
go a bit smoother. You should remove them before moving into a
production environment.
```

9. Connect Database .

Create Database - >

Create Table -> insert values on that table .



```
[root@ip-172-31-89-81 ~]# mysql -h localhost -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 23
Server version: 10.5.25-MariaDB MariaDB Server

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

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

MariaDB [(none)]> create database college;
Query OK, 1 row affected (0.000 sec)

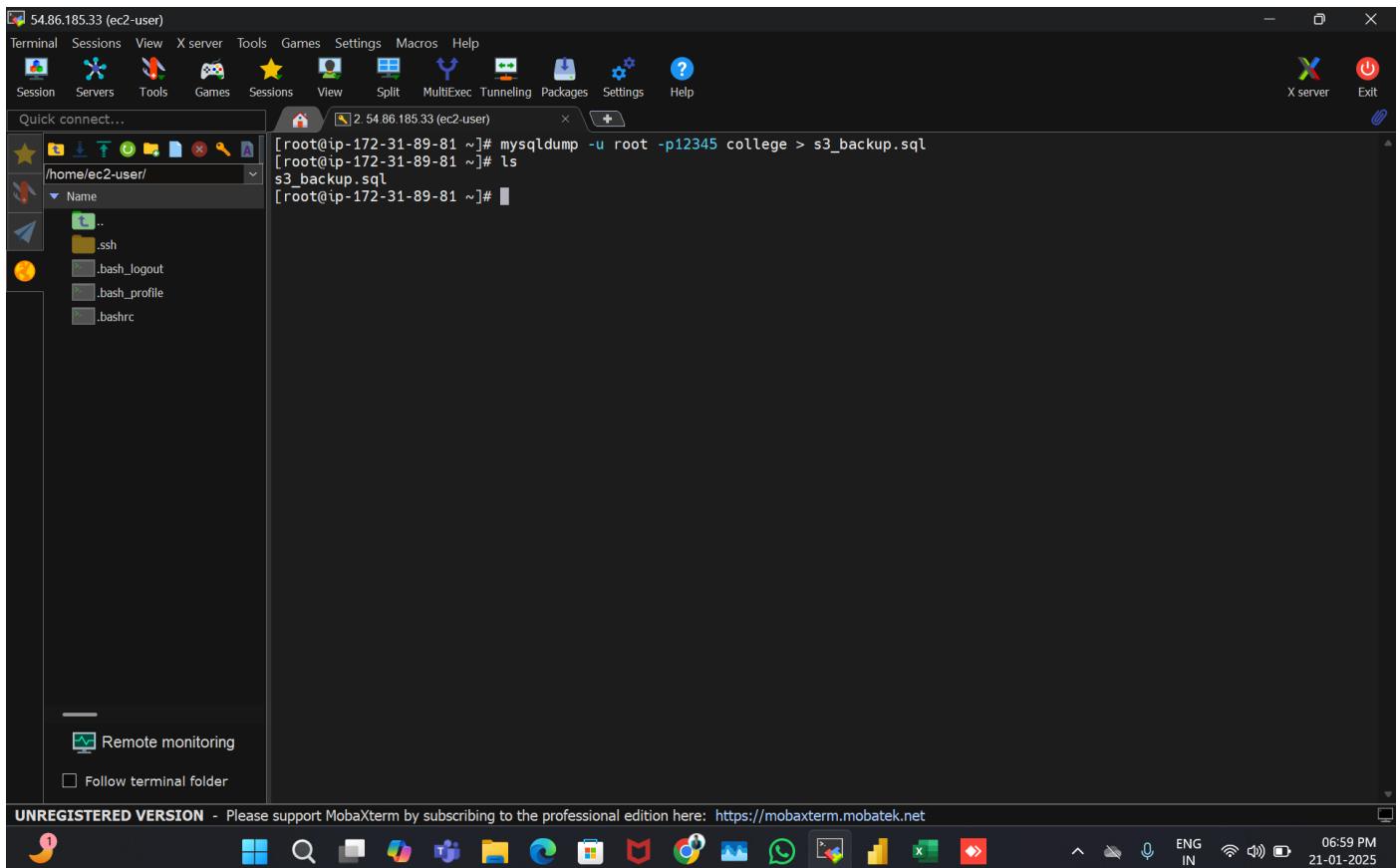
MariaDB [(none)]> use college
Database changed
MariaDB [college]> create table student_info(rollno int, name varchar(20), address varchar(20));
Query OK, 0 rows affected (0.007 sec)

MariaDB [college]> insert into student_info values(1,"darshan","nashik"),(2,"mayuresh","nagar"),(3,"rithesh","pune"),(4,"akshada","nashik"),(5,"damini","nashik"),(6,"harshada","satana");
ERROR 1054 (42S22): Unknown column 'nashik' in 'field list'
MariaDB [college]> insert into student_info values(1,"darshan","nashik"),(2,"mayuresh","nagar"),(3,"rithesh","pune"),(4,"akshada","nashik"),(5,"damini","nashik"),(6,"harshada","satana");
Query OK, 6 rows affected (0.003 sec)
Records: 6  Duplicates: 0  Warnings: 0

MariaDB [college]> select * from student_info;
+-----+-----+-----+
| rollno | name   | address |
+-----+-----+-----+
|     1  | darshan | nashik |
|     2  | mayuresh | nagar  |
|     3  | rithesh | pune   |
|     4  | akshada | nashik |
|     5  | damini  | nashik |
|     6  | harshada | satana |
+-----+-----+-----+
6 rows in set (0.001 sec)
```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

10. Create a Backup File of Database and give a name of that file .



```
[root@ip-172-31-89-81 ~]# mysqldump -u root -p12345 college > s3_backup.sql
[root@ip-172-31-89-81 ~]# ls
s3_backup.sql
[root@ip-172-31-89-81 ~]#
```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

11. Create a Bucket .

The screenshot shows the 'Create bucket' configuration page in the AWS Management Console. The 'General configuration' section is visible, showing the AWS Region as 'US East (N. Virginia) us-east-1'. Under 'Bucket type', 'General purpose' is selected, with a note about its use for most use cases. A 'Bucket name' field contains 'my-db-backup1'. Below it, a note states that the bucket name must be unique and follow naming rules, with a link to 'See rules for bucket naming'. A 'Copy settings from existing bucket - optional' section is present, with a 'Choose bucket' button and a placeholder 'Format: s3://bucket/prefix'. The browser's address bar shows the URL: `us-east-1.console.aws.amazon.com/s3/bucket/create?region=us-east-1&bucketType=general`. The bottom of the screen shows the Windows taskbar with various pinned icons.

12. There is no data in that bucket.

The screenshot shows the 'my-db-backup1' objects page in the AWS Management Console. The 'Objects' tab is selected, showing '0' objects. A prominent orange 'Upload' button is at the top right. Below it, a message says 'No objects' and 'You don't have any objects in this bucket.' The browser's address bar shows the URL: `us-east-1.console.aws.amazon.com/s3/buckets/my-db-backup1?region=us-east-1&bucketType=general&tab=objects`. The bottom of the screen shows the Windows taskbar with various pinned icons.

13. Create a Role and Select the Service .

Create a custom trust policy to enable others to perform actions in this account.

Use case
Allow an AWS service like EC2, Lambda, or others to perform actions in this account.

Service or use case
EC2

Choose a use case for the specified service.

Use case

- EC2
Allows EC2 instances to call AWS services on your behalf.
- EC2 Role for AWS Systems Manager
Allows EC2 instances to call AWS services like CloudWatch and Systems Manager on your behalf.
- EC2 Spot Fleet Role
Allows EC2 Spot Fleet to request and terminate Spot Instances on your behalf.
- EC2 - Spot Fleet Auto Scaling
Allows Auto Scaling to access and update EC2 spot fleets on your behalf.
- EC2 - Spot Fleet Tagging
Allows EC2 to launch spot instances and attach tags to the launched instances on your behalf.
- EC2 - Spot Instances

14 . Select the policies of your role.

Step 1
Select trusted entity

Step 2
Add permissions

Step 3
Name, review, and create

Add permissions Info

Permissions policies (1/1022) Info

Choose one or more policies to attach to your new role.

Filter by Type

s3 All types 13 matches

Policy name	Type	Description
AmazonDMSRedshiftS3R...	AWS managed	Provides access to manage S3 settings...
<input checked="" type="checkbox"/> AmazonS3FullAccess	AWS managed	Provides full access to all buckets via t...
AmazonS3ObjectLambda...	AWS managed	Provides AWS Lambda functions permi...
AmazonS3OutpostsFullA...	AWS managed	Provides full access to Amazon S3 on ...
AmazonS3OutpostsRead...	AWS managed	Provides read only access to Amazon S...
AmazonS3ReadOnlyAccess	AWS managed	Provides read only access to all bucket...
AmazonS3TablesFullAcces...	AWS managed	Provides full access to all S3 table bu...

15 . Give a Role name .

The screenshot shows the 'Name, review, and create' step of the IAM role creation wizard. On the left, a sidebar lists 'Step 1 Select trusted entity', 'Step 2 Add permissions', 'Step 3', and 'Name, review, and create' (which is highlighted). The main area is titled 'Role details'. It contains fields for 'Role name' (set to 's3-ec2-role'), 'Description' (set to 'Allows EC2 instances to call AWS services on your behalf.'), and a note about character limits. Below this is the 'Step 1: Select trusted entities' section, which includes a 'Trust policy' editor showing the JSON policy:

```
1 "Version": "2012-10-17",  
2 "Statement": [
```

. The browser's address bar shows 'us-east-1.console.aws.amazon.com/iam/home?region=us-east-1#/roles'.

16. Your Role is Successfully Created .

The screenshot shows the 'Roles (1/7)' page in the IAM console. The left sidebar shows 'Identity and Access Management (IAM)' and 'Access management' (with 'Roles' selected). The main area displays a table of roles, with the 's3-ec2-role' row selected (indicated by a blue border). The table columns are 'Role name', 'Trusted entities', and 'Last activity'. The 's3-ec2-role' row shows it was created for the 'ec2' service. The browser's address bar shows 'https://us-east-1.console.aws.amazon.com/iam/home?region=us-east-1#/roles'.

Role name	Trusted entities	Last activity
AWSServiceRoleForAutoScaling	AWS Service: autoscaling (Service-Link...	39 days ago
AWSServiceRoleForElasticLoadBalancing	AWS Service: elasticloadbalancing (S...	39 days ago
AWSServiceRoleForRDS	AWS Service: rds (Service-Linked Rol...	30 minutes
AWSServiceRoleForSupport	AWS Service: support (Service-Linker...	-
AWSServiceRoleForTrustedAdvisor	AWS Service: trustedadvisor (Service...	-
rds-monitoring-role	AWS Service: monitoring.rds	-
s3-ec2-role	AWS Service: ec2	-

17. Assign Role to Ec2-instance .

Goto security -> modify IAM role.

The screenshot shows the AWS EC2 Instances page. A context menu is open over an instance named 's3-Backup-ins...'. The 'Modify IAM role' option is highlighted in the menu. The instance details show it is 'Running' with an 'i-0435ebedd442defa4' ID and 't2.micro' type.

18. Select the you created role and assign that.

The screenshot shows the 'Modify IAM role' dialog box. The 'Instance ID' field is set to 'i-0435ebedd442defa4 (s3-Backup-instance)'. The 'IAM role' section shows a dropdown menu with 's3-ec2-role' selected. A confirmation message at the bottom right says 'want to remove from the selected instance?'. The 'Update IAM role' button is visible at the bottom right.

19. Role Successfully attached to ec2-instance.

The screenshot shows the AWS Management Console with the 'EC2' service selected. A green success message at the top right states: 'Successfully attached s3-ec2-role to instance i-0435ebbedd442defa4'. Below this, the 'Instances' section displays one instance named 's3-Backup-ins...' with the ID 'i-0435ebbedd442defa4', which is currently 'Running'. The instance type is 't2.micro'. The details page for this instance is open, showing its summary information including Public IPv4 address (54.86.185.33), Private IPv4 address (172.31.89.81), and Public IPv4 DNS. The AWS navigation bar at the bottom includes CloudShell, Feedback, and various system icons.

20. Enter the cmd and then get backup of your database file.

The screenshot shows a terminal session in MobaXterm connected to the IP 54.86.185.33 (ec2-user). The user runs 'mysqldump -u root -p12345 college > s3_backup.sql' to create a backup of the 'college' database. They then use 'aws s3 cp s3_backup.sql s3://my-db-backup1/database-backup.sql' to upload the backup file to an S3 bucket. The terminal output indicates that the user-provided path 's3_backup_sql' does not exist, so the command is modified to use the full file name. The session also shows the user navigating through their home directory (~) and viewing files like .ssh and .bashrc.

21. Goto S3 bucket .See your database backup file is .

The screenshot shows the AWS S3 console with the URL <https://us-east-1.console.aws.amazon.com/s3/buckets/my-db-backup1?region=us-east-1&bucketType=general&tab=objects>. The top navigation bar includes links for IAM, S3, EC2, VPC, and RDS. The main content area displays the 'my-db-backup1' bucket with one object listed:

Name	Type	Last modified	Size	Storage class
database-backup.sql	sql	January 21, 2025, 19:12:35 (UTC+05:30)	2.1 KB	Standard



22. For Cross Check (Open it).

(Your data)

```
File Edit View
/*
-- enable the sandbox mode /
-- MariaDB dump 10.19 Distrib 10.5.25-MariaDB, for Linux (x86_64)
--
-- Host: localhost      Database: college
-- -----
-- Server version      10.5.25-MariaDB

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8mb4 */;
/*!40103 SET @OLD_TIME_ZONE=@TIME_ZONE */;
/*!40103 SET TIME_ZONE='+00:00' */;
/*!40014 SET @OLD_UNIQUE_CHECKS=@UNIQUE_CHECKS, UNIQUE_CHECKS=0 */;
/*!40014 SET @OLD_FOREIGN_KEY_CHECKS=@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0 */;
/*!40101 SET @OLD_SQL_MODE=@SQL_MODE, SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;
/*!40111 SET @OLD_SQL_NOTES=@SQL_NOTES, SQL_NOTES=0 */;

-- Table structure for table `student_info`


DROP TABLE IF EXISTS `student_info`;
/*!40101 SET @saved_cs_client_ = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `student_info` (
  `rollno` int(11) DEFAULT NULL,
  `name` varchar(20) DEFAULT NULL,
  `address` varchar(20) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COLLATE=latin1_swedish_ci;
/*!40101 SET character_set_client = @saved_cs_client */;

-- Dumping data for table `student_info`


LOCK TABLES `student_info` WRITE;
/*!40000 ALTER TABLE `student_info` DISABLE KEYS */;
INSERT INTO `student_info`(`rollno`, `name`, `address`)
VALUES (1, 'John Doe', '123 Main St');
INSERT INTO `student_info`(`rollno`, `name`, `address`)
VALUES (2, 'Jane Smith', '456 Elm St');
INSERT INTO `student_info`(`rollno`, `name`, `address`)
VALUES (3, 'Bob Johnson', '789 Oak St');

UNLOCK TABLES;
Ln 1, Col 1  2,140 characters
100% Unix (LF) UTF-8
CloudShell Feedback
© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences
ENG IN 07:12 PM 21-01-2025
```