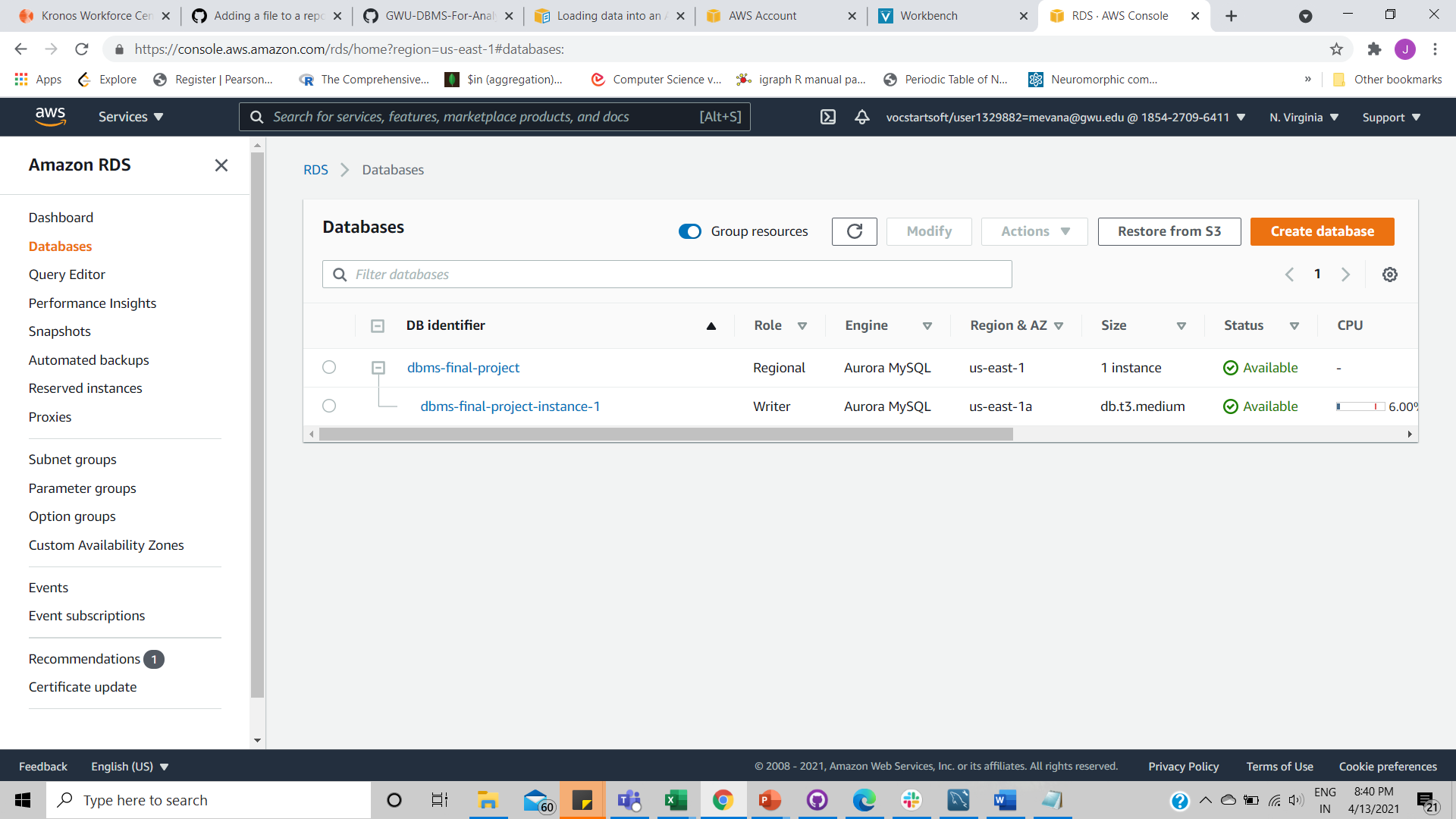
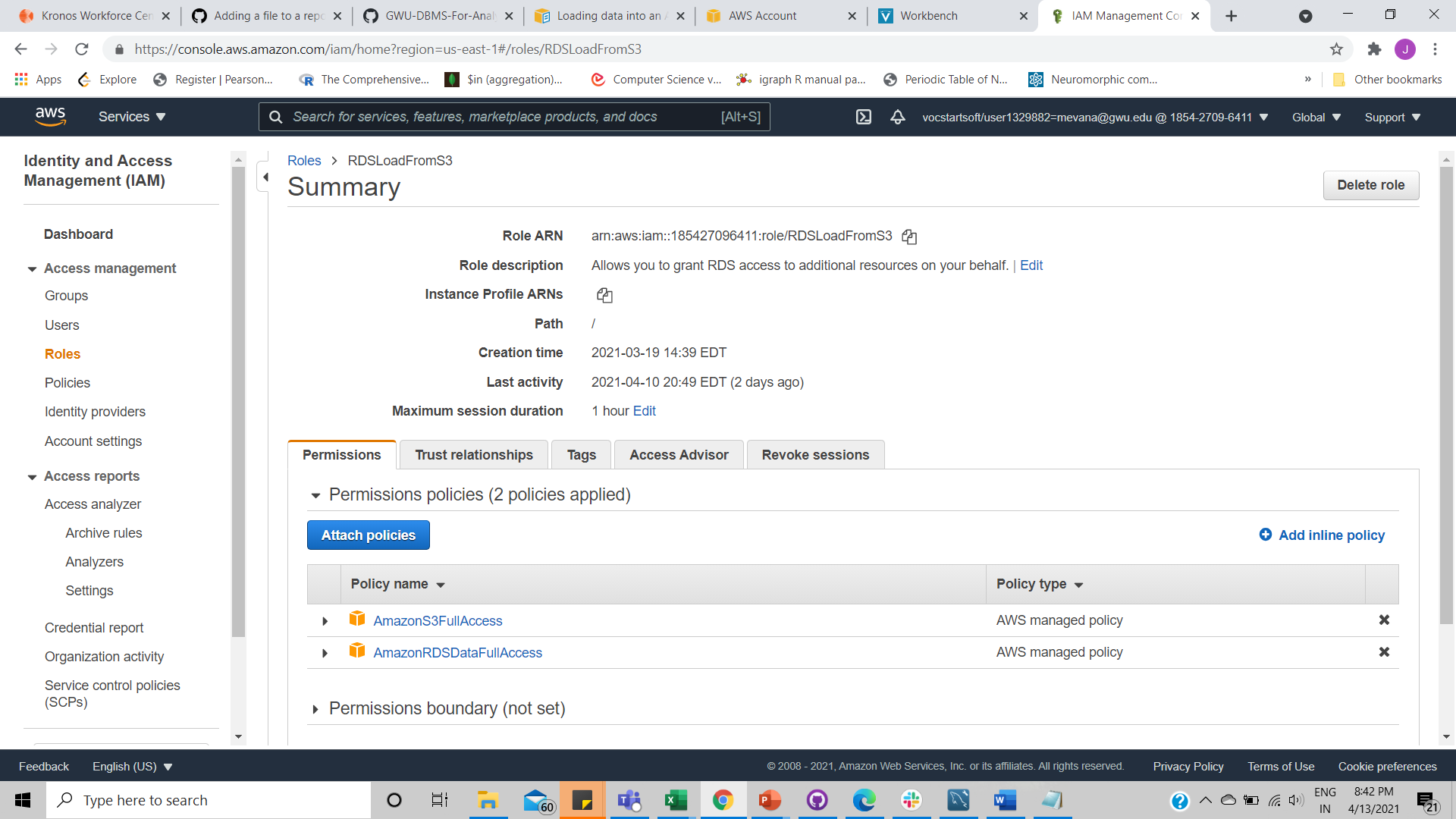
Steps to Create database in AWS and connect to SQL:

1. Open AWS RDS: Create an AWS Identity and Access Management (IAM) policy that provides the bucket and object permissions that allow your Aurora MySQL DB cluster to access Amazon S3.



1. Create an IAM role, and attach the IAM policy you created in [Creating an IAM policy to access Amazon S3 resources](https://docs.aws.amazon.com/AmazonRDS/latest/AuroraUserGuide/AuroraMySQL.Integrating.Authorizing.IAM.S3CreatePolicy.html) to the new IAM role.



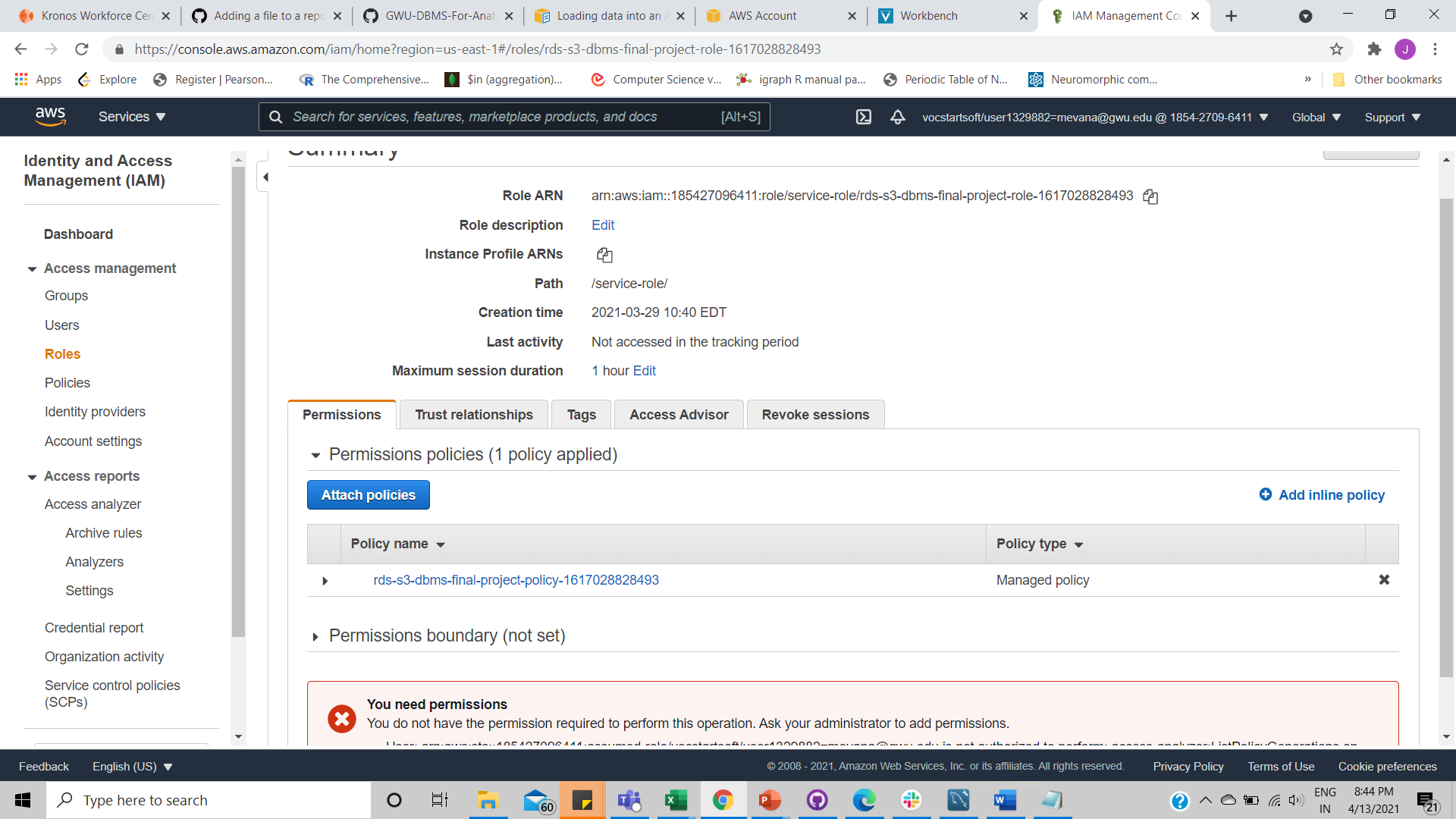
3.To create an IAM policy to grant access to your Amazon S3 resources

1. Open the [IAM Management Console](https://console.aws.amazon.com/iam/home?#home).
2. In the navigation pane, choose Policies.
3. Choose Create policy.
4. On the Visual editor tab, choose a service, and then choose S3.
5. For Actions, choose Expand all, and then choose the bucket permissions and object permissions needed for the IAM policy.

Object permissions are permissions for object operations in Amazon S3, and need to be granted for objects in a bucket, not the bucket itself. Choose Resources, and choose Add ARN for bucket.

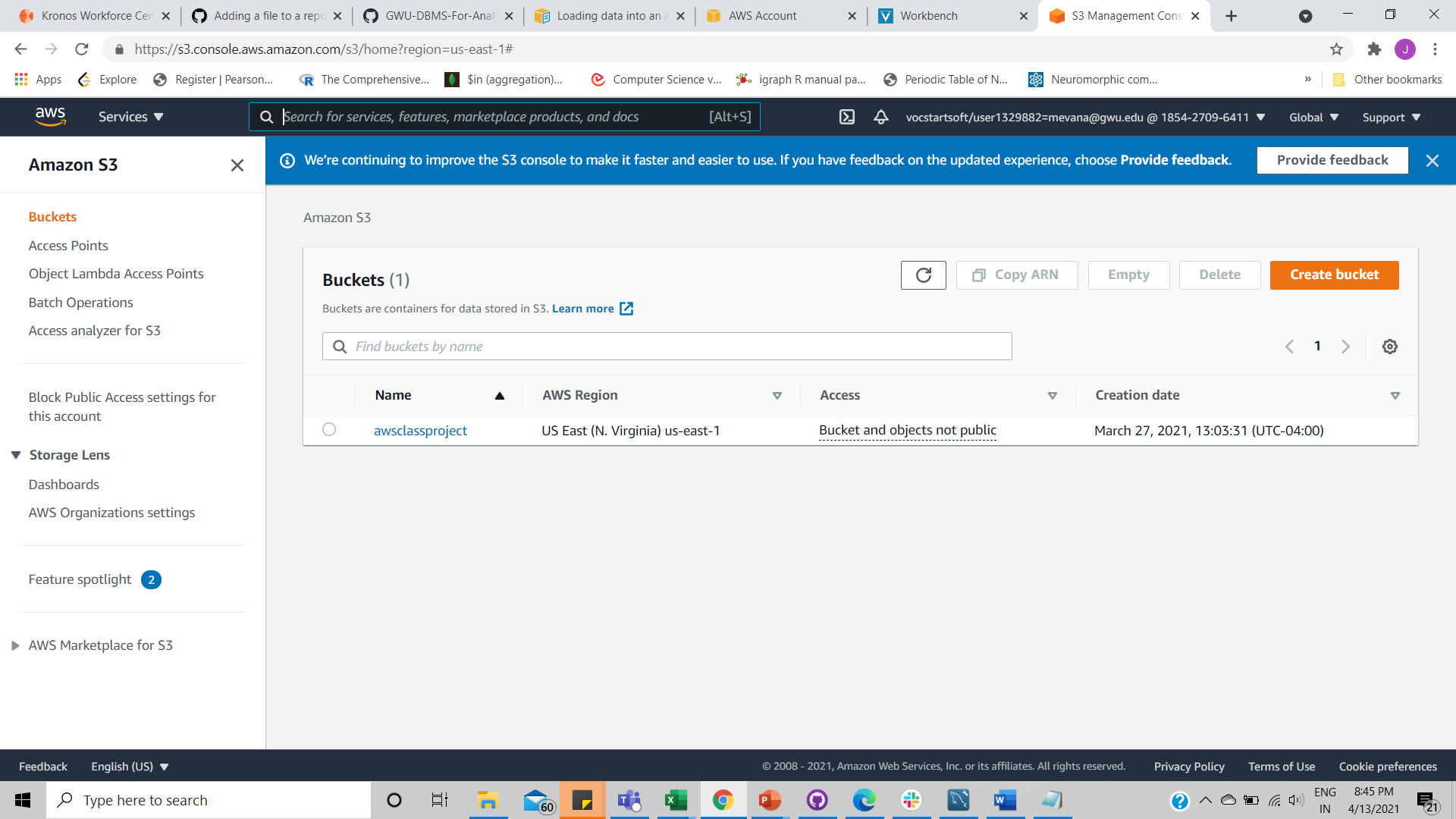
1. In the Add ARN(s) dialog box, provide the details about your resource, and choose Add.

Specify the Amazon S3 bucket to allow access to. For instance, if you want to allow Aurora to access the Amazon S3 bucket named example-bucket, then set the Amazon Resource Name (ARN) value to arn:aws:s3:::example-bucket.



1. If the object resource is listed, choose Add ARN for object.
2. In the Add ARN(s) dialog box, provide the details about your resource.
3. Choose Review policy.
4. For Name, enter a name for your IAM policy, for example AllowAuroraToExampleBucket. You use this name when you create an IAM role to associate with your Aurora DB cluster. You can also add an optional Description value.
5. Choose Create policy.

For the Amazon S3 bucket, specify the Amazon S3 bucket to allow access to. For the object, you can choose Any to grant permissions to any object in the bucket.



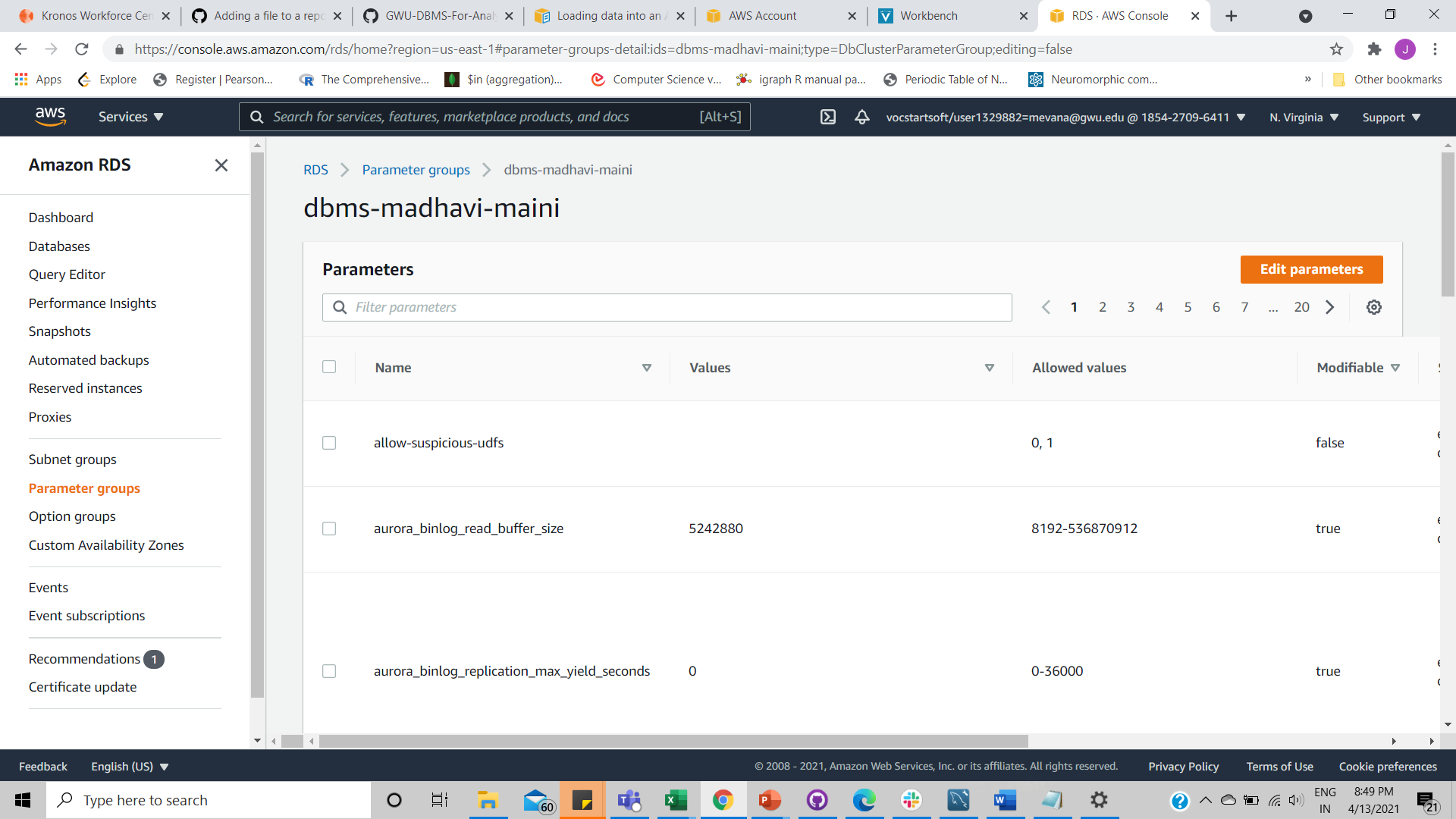
1. Make sure the DB cluster is using a custom DB cluster parameter group.

For more information about creating a custom DB cluster parameter group.

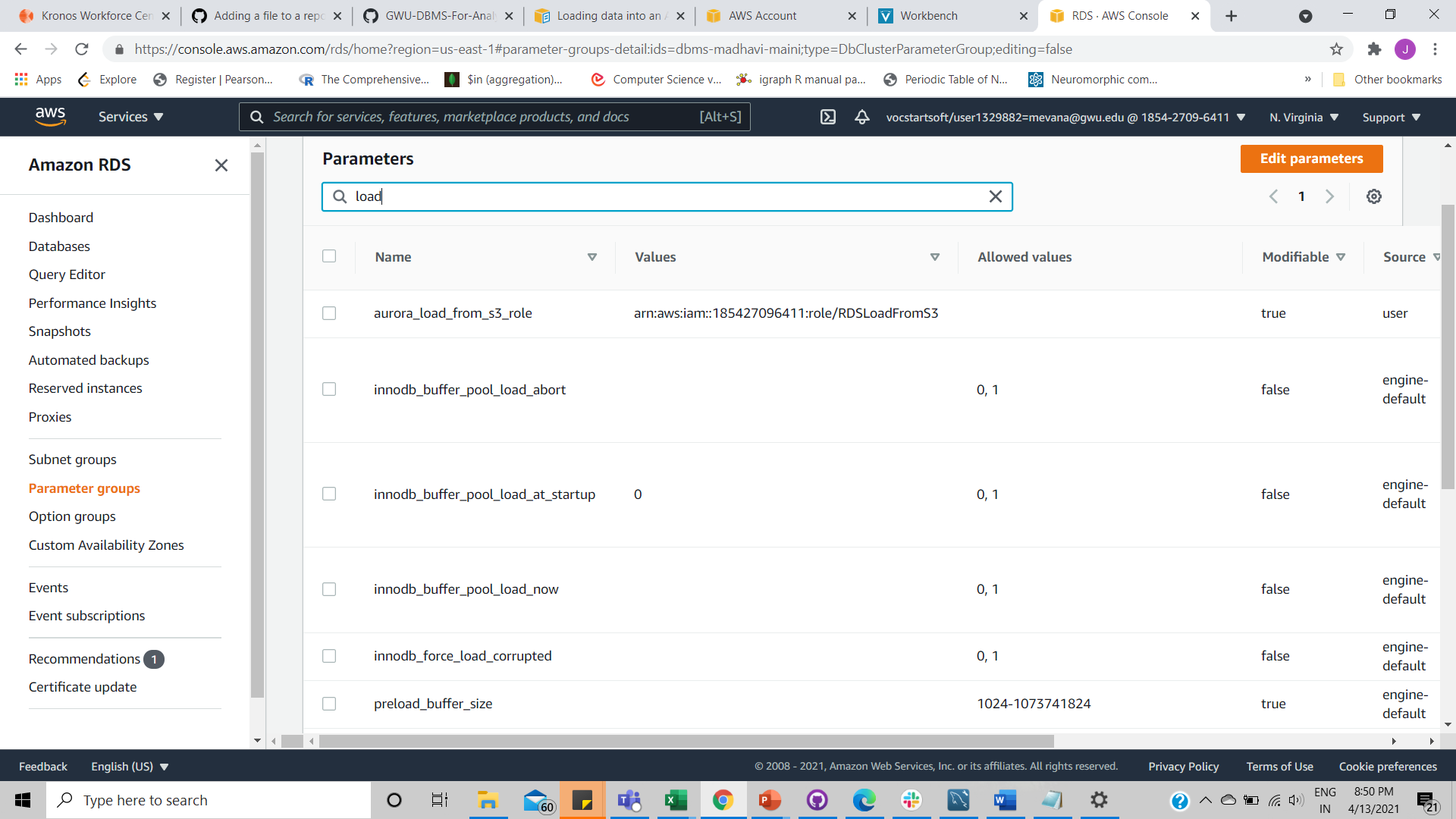
1. Sign in to the AWS Management Console and open the Amazon RDS console at <https://console.aws.amazon.com/rds/>.
2. In the navigation pane, choose Parameter groups.
3. Choose Create parameter group.

The Create parameter group window appears.

1. In the Parameter group family list, select a DB parameter group family
2. In the Type list, select DB Cluster Parameter Group.
3. In the Group name box, enter the name of the new DB cluster parameter group.
4. In the Description box, enter a description for the new DB cluster parameter group.
5. Choose Create.



1. Set either the aurora\_load\_from\_s3\_role or aws\_default\_s3\_role DB cluster parameter to the Amazon Resource Name (ARN) of the new IAM role. If an IAM role isn't specified for aurora\_load\_from\_s3\_role, Aurora uses the IAM role specified in aws\_default\_s3\_role.



1. To permit database users in an Aurora MySQL DB cluster to access Amazon S3, associate the role that you created in [Creating an IAM role to allow Amazon Aurora to access AWS services](https://docs.aws.amazon.com/AmazonRDS/latest/AuroraUserGuide/AuroraMySQL.Integrating.Authorizing.IAM.CreateRole.html) with the DB cluster. For an Aurora global database, associate the role with each Aurora cluster in the global database. For information about associating an IAM role with a DB cluster, see [Associating an IAM role with an Amazon Aurora MySQL DB cluster](https://docs.aws.amazon.com/AmazonRDS/latest/AuroraUserGuide/AuroraMySQL.Integrating.Authorizing.IAM.AddRoleToDBCluster.html).
2. Configure the Aurora MySQL DB cluster to allow outbound connections to Amazon S3
3. Specifying a path to an Amazon S3 bucket:

s3-region://bucket-name/file-name-or-prefix

1. LOAD DATA FROM S3