Setting up & Connecting to the Microsoft Azure SQL Cloud Database

Table of Contents

Brief Intro to Cloud DBMS. What and Why?

For the upcoming assignments, we will be using an Azure SQL DBMS hosted in the cloud. Storage of the database schema and data and management of hardware and software needed to run the DBMS will all be managed by the cloud provider (Microsoft). We will configure our database through the web interface and connect to it through the client applications on your laptop/desktop (web browser, Azure Data Studio IDE, your Java application, etc.)

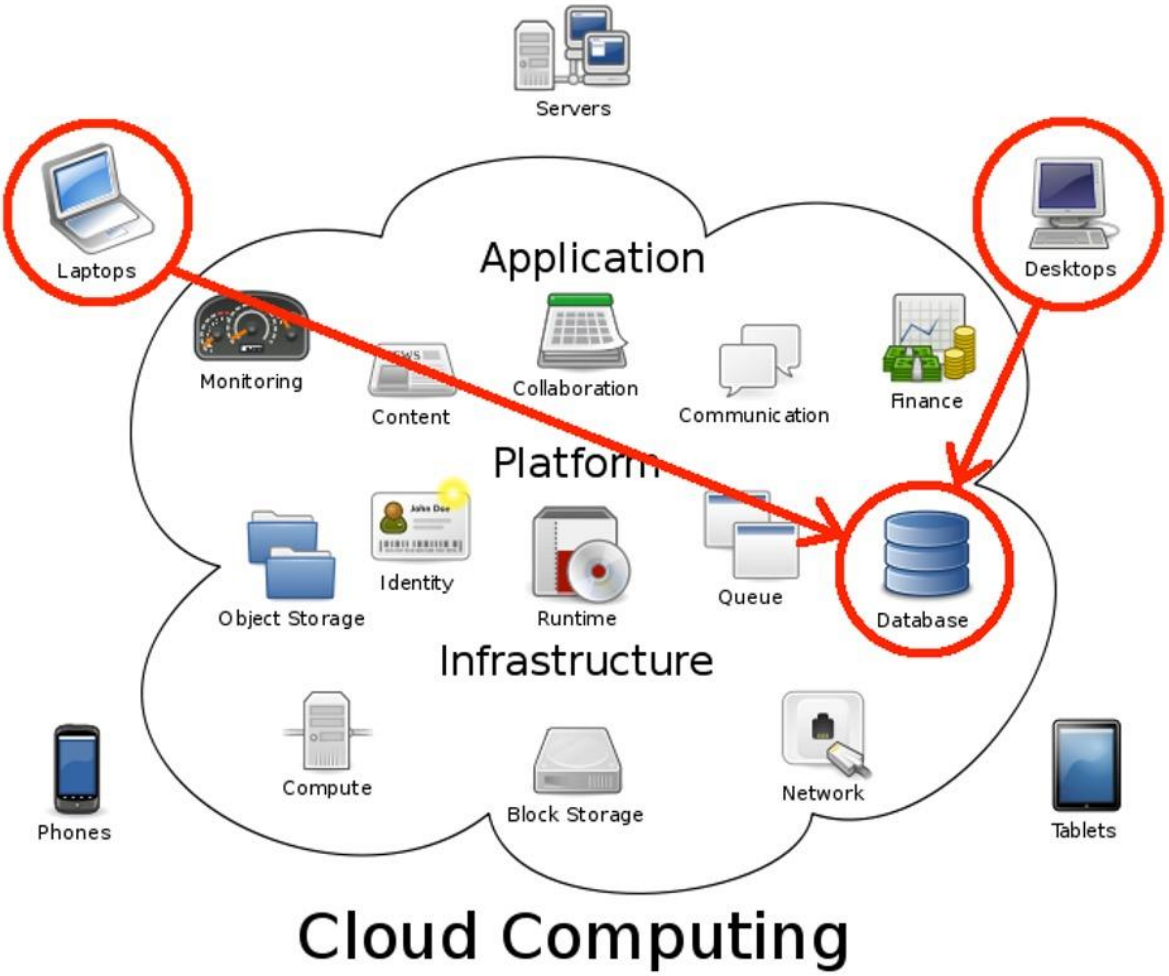


Image source: https://www.fastmetrics.com/blog/tech/what-is-cloud-computing/

Using the cloud for the database hosting is a case of purchasing a Platform-as-a-Service (PaaS) from the cloud provider. Doing so allows us to only care about designing the database, queries, and configuring/developing the applications we use to connect and issue queries to the database (see below image). We no longer need to install the software and provision the hardware to run the DBMS. We also don't need to keep our own desktop/laptop/server running to keep the DBMS alive, as well as we don't require setting up a web server or a domain name so that our database is accessible from the internet. So, as you can tell, this setup offers quite a few advantages.

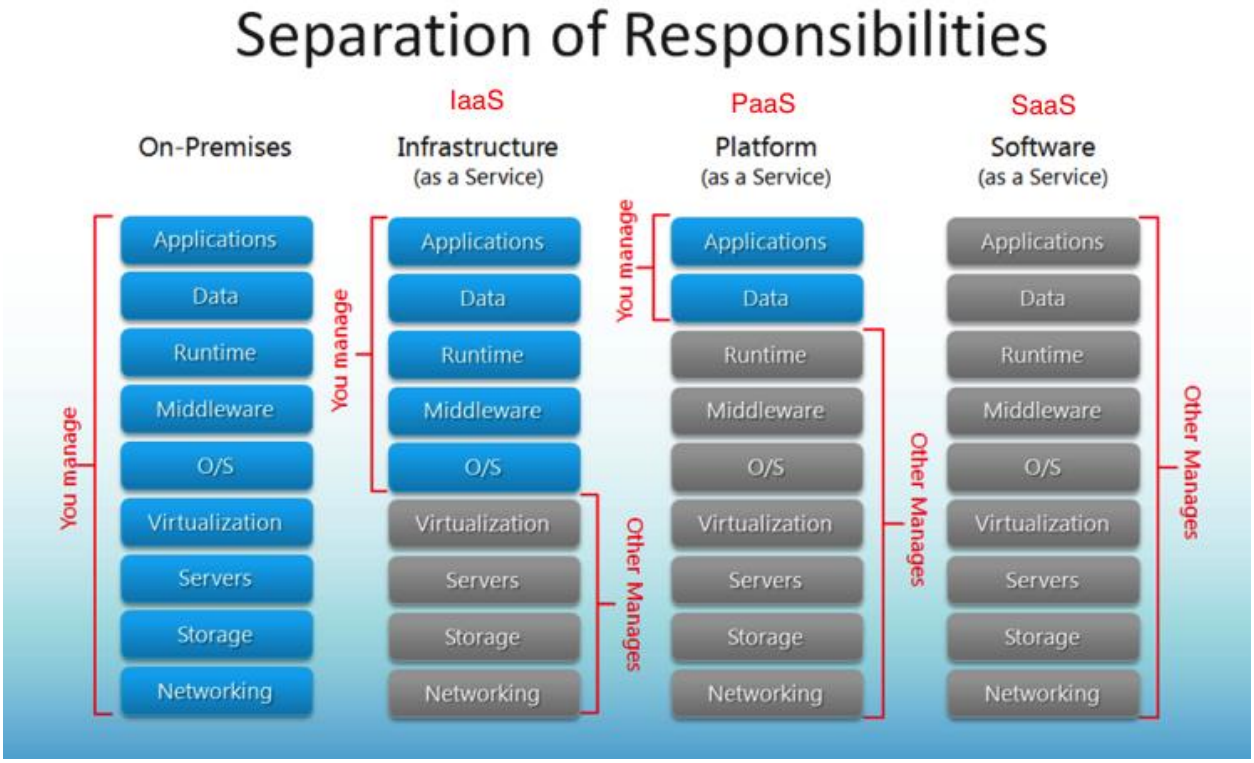
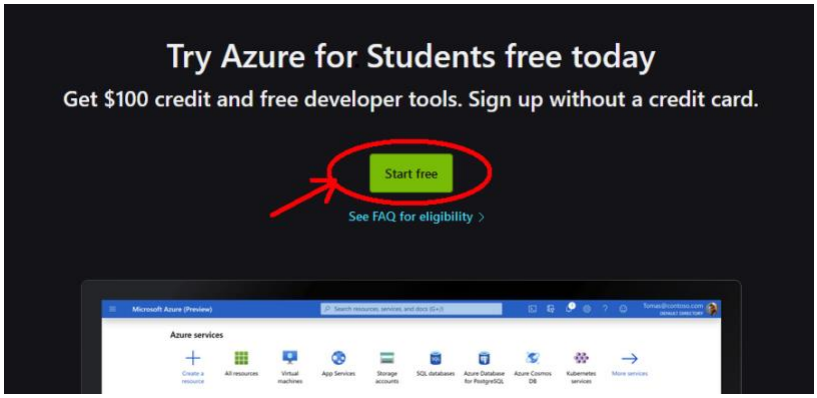


Image source: https://robertgreiner.com/windows-azure-iaas-paas-saas-overview/

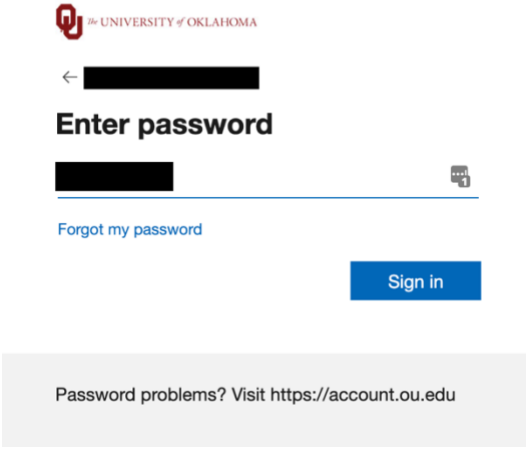
Creating a Free Microsoft Azure Student Subscription

Steps to Follow

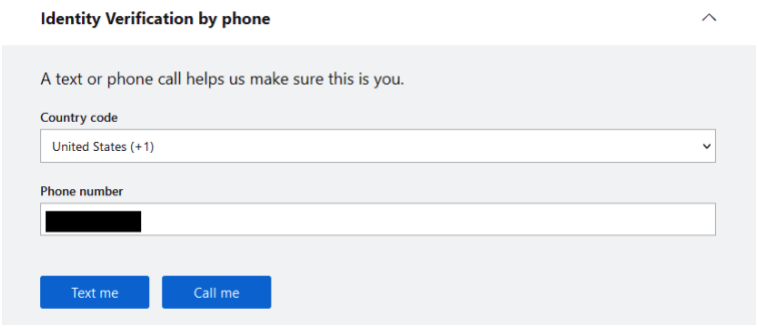
1. Open new **Incognito** (this is important) window in your web browser.
2. Go to - https://azure.microsoft.com/en-us/free/students/
3. Click on "Activate Now" button



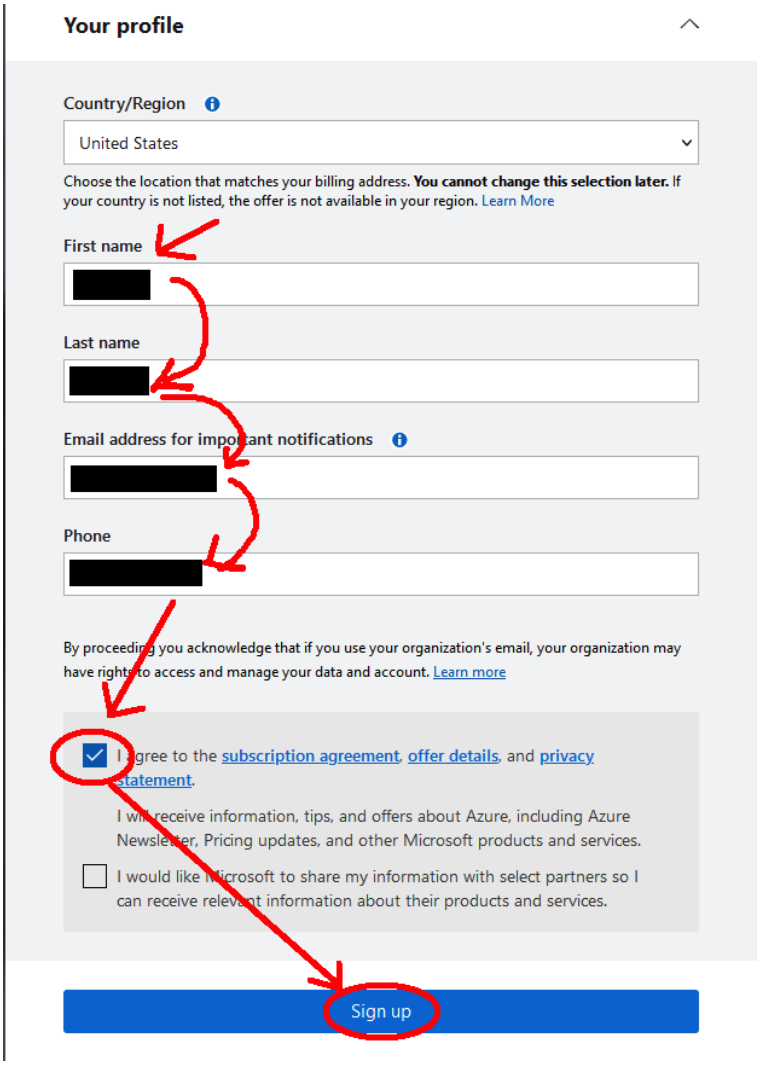
4. Above might lead you to https://login.microsoftonline.com first where you will be asked to log into your OU account



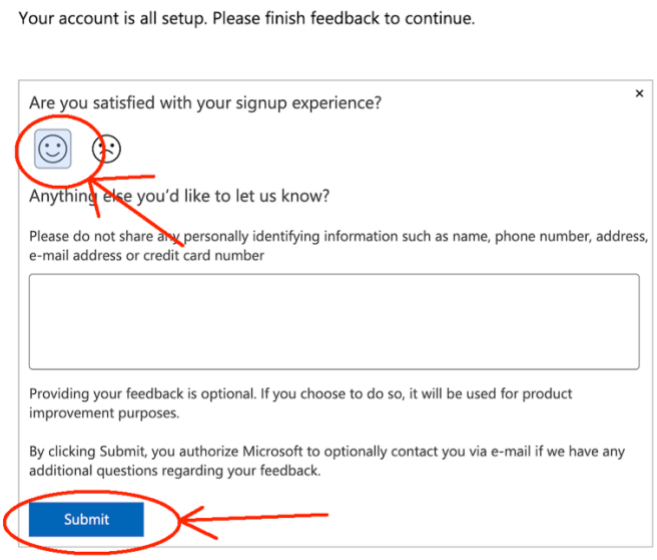
5. You should get redirected to https://signup.azure.com very you will be asked to provide and verify your phone number.



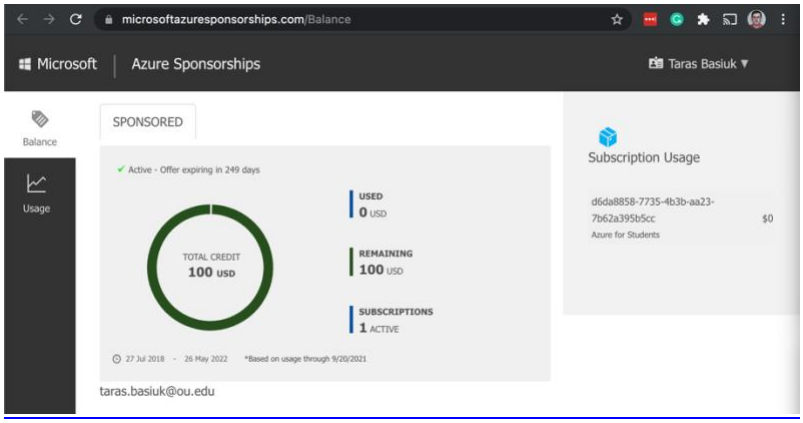
6. Next, fill the forms, read, and accept subscription agreement, offer details, privacy statement, and communications policy. Click "Sign Up" button.



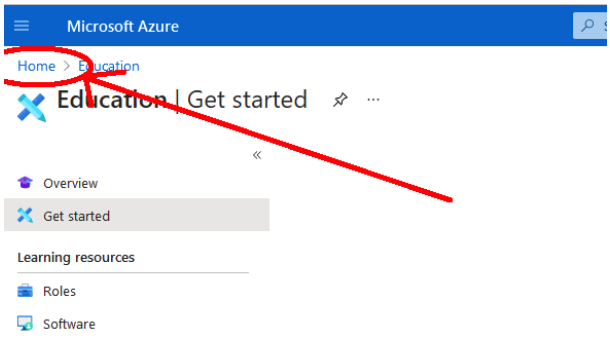
7. If prompted for feedback, please submit it (see screenshot below).



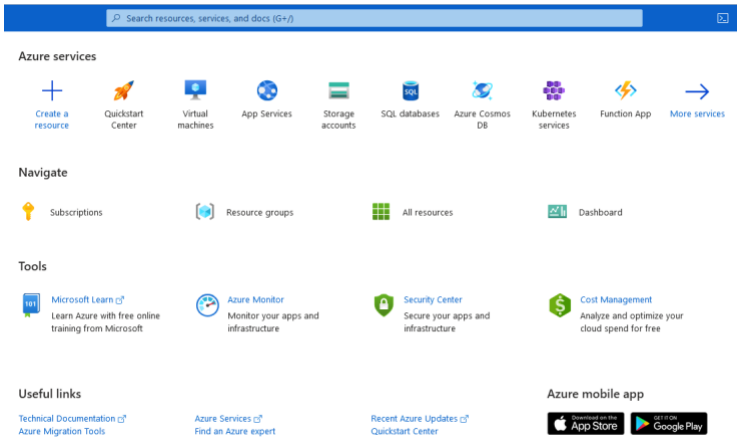
8. Doing all above will add $100 of free credit to your student account to be used within a year. You can check your remaining balance (see the screenshot below) at https://www.microsoftazuresponsorships.com/Balance



9. When logging in https://portal.azure.com for the first time you might see "Education" tab opened for you. We won't be using it, so go to "Home"



10. Once you get redirected to https://portal.azure.com/#home and see the below perspective on your screen, you're ready to proceed to create your own SQL server



Possible Complications

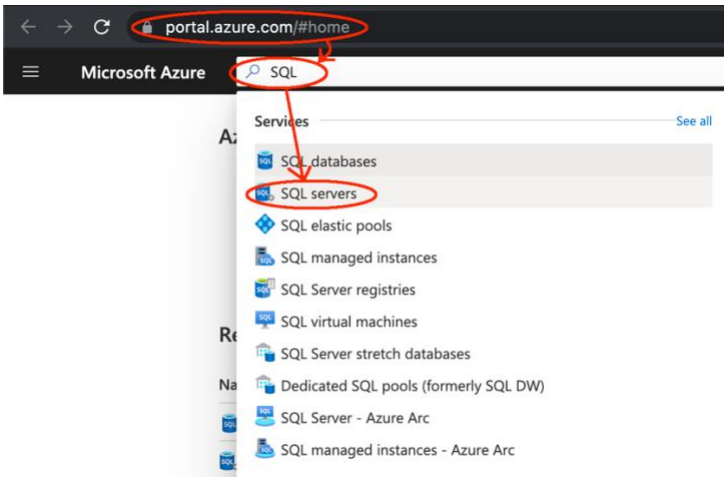
We estimate that the free credit provided by the Microsoft for your account would be sufficient for your work in this course if you carefully follow the instructions below. However, If you find yourself ineligible for the offer, short of free credit or your student account was (will be) deactivated due to the offer expiring after a year, you need to upgrade it to a pay-as-you-go account (see the link below), in which case your expenditures will sum up to $5/month if you carefully follow our instructions.

https://docs.microsoft.com/en-us/azure/billing/billing-upgrade-azure-subscription

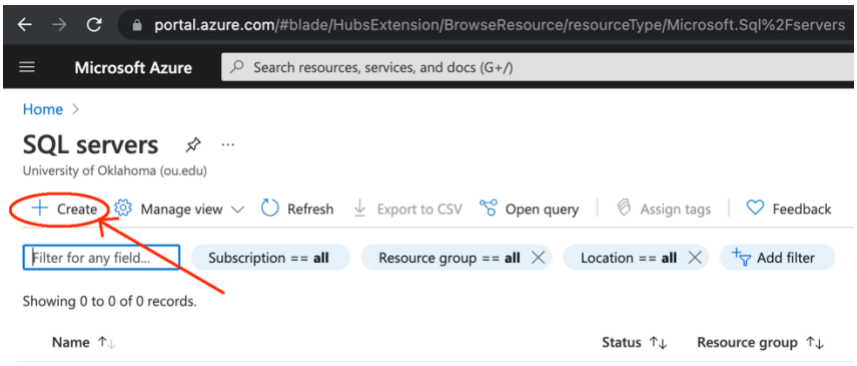
Creating your Personal SQL Server

Steps to Follow

1. Navigate to https://portal.azure.com/#home Enter SQL into the search bar, select "SQL Server" from the presented options



2. You should be redirected to the webpage seen on the screenshot below. Click on "+ Add" button.

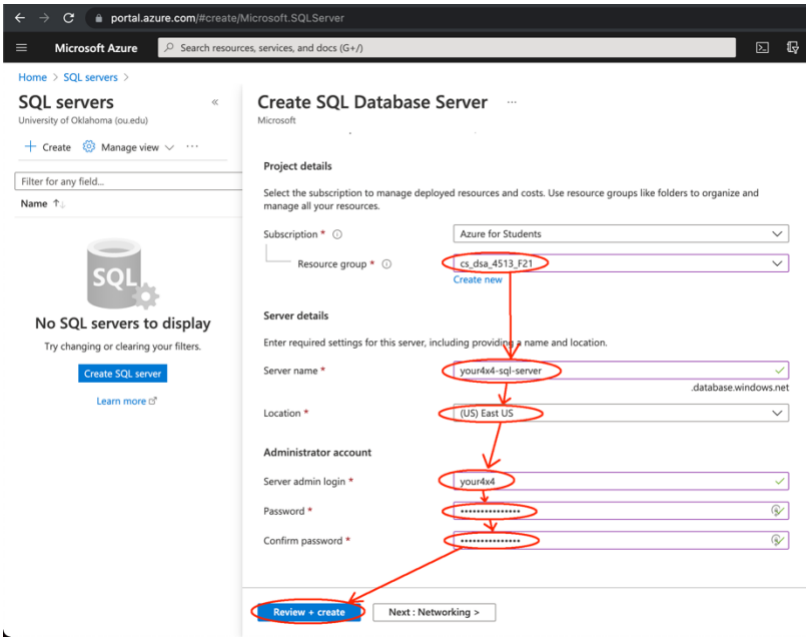


3. You should be redirected to the web-page seen on the screenshot below with the fields being empty (or having default values).
4. Type in the values into the empty fields of the right panel, make appropriate option selections (as seen on the below screenshot), create and select a new resource group

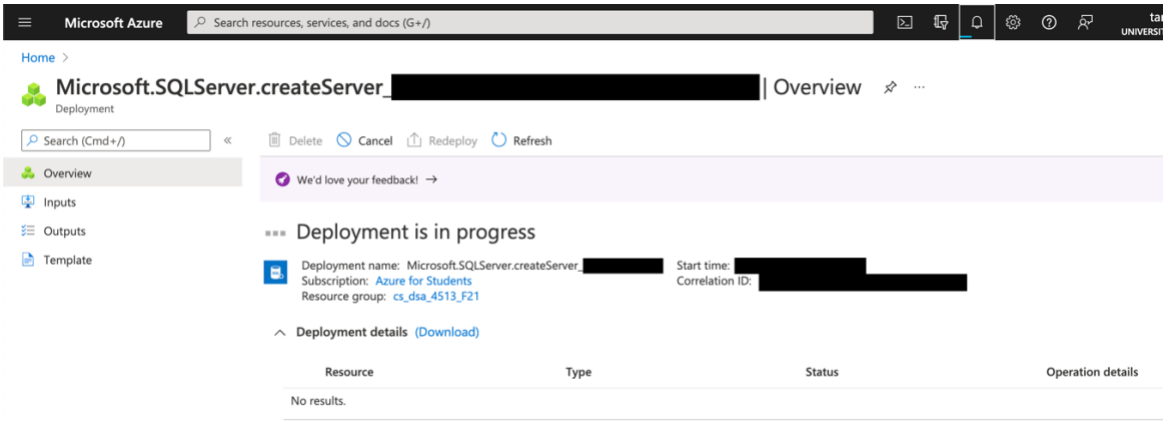
cs_dsa_4513_F21 by clicking on "create new" link, pick your own password. Instead of our <*your4x4*> placeholder, use your actual 4x4 (a.k.a OUNetID), which you can look-up at https://accounts.ou.edu/

   a. !!!WARNING!!! Some students reported problems creating database servers in "(US) East US" location. If you're experiencing it, try picking a different location geographically adjacent to you. If nothing works, try again later. If nothing is available after a day, please contact us. !!!WARNING!!!

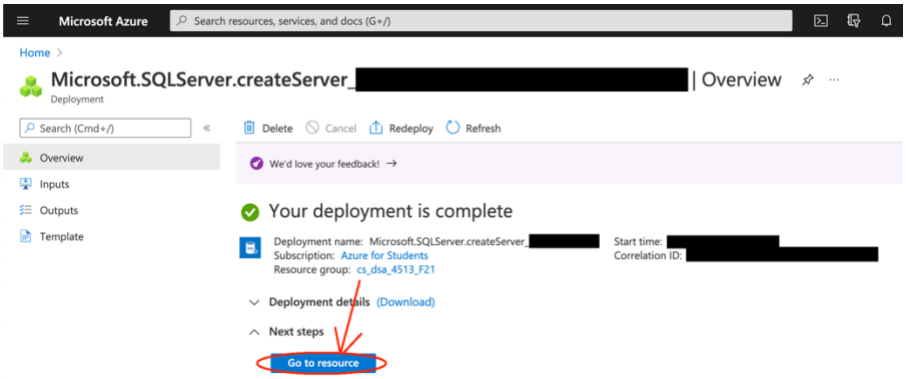
5. Finally, click on the "Review + create" button.



6. Clicking the "Create" button on the next screen will start the deployment of your SQL server. You should be redirected to the deployment progress reporting window.



7. Wait for your deployment to finish. If everything is well, then click on "Go to resource" and proceed to the next section of this instruction document.



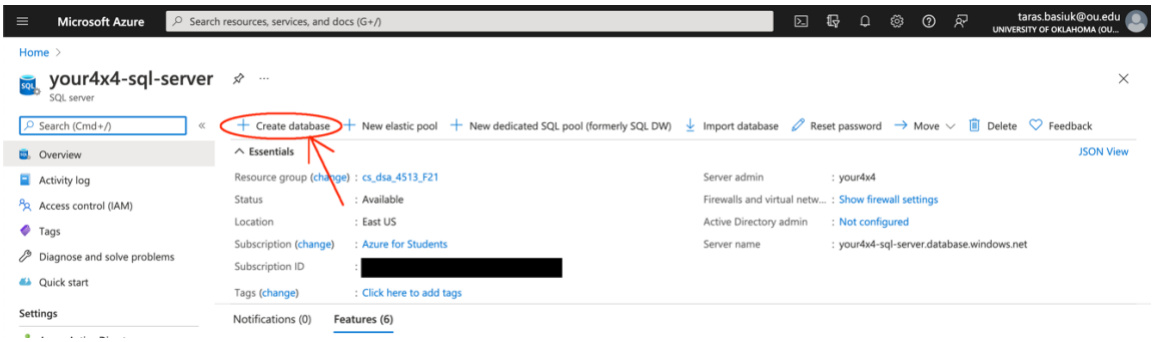
Possible Complications

If you made a mistake while creating your SQL server (wrong input for example), short of resetting the password, adjustments cannot be made. Instead, you should delete the server and create a new one. Follow the below instruction regarding the navigation to your newly created SQL server. But instead of following other instructions, click on "Delete" button, confirm the deletion by typing in the server name ("*<your4x4>-sql-server*") into the form and click another "Delete" button at the bottom. Now repeat the instructions at the beginning of Part 2, but now hopefully correcting for your mistake.

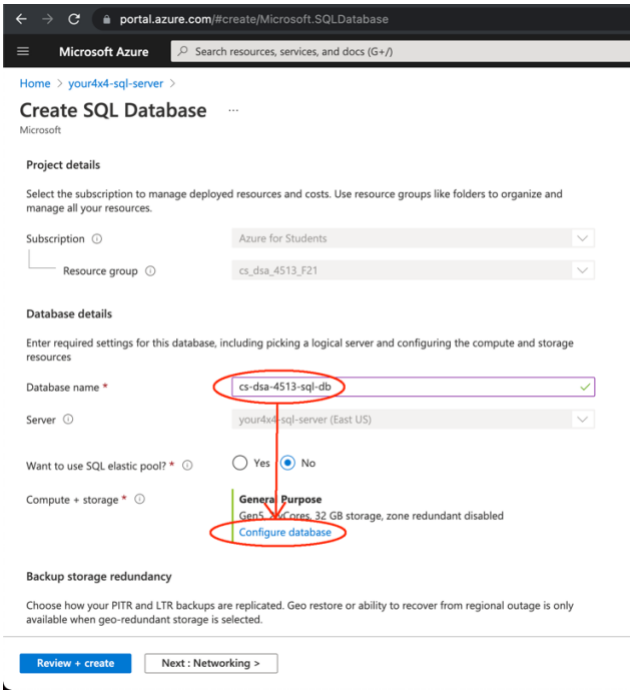
Create Your Own SQL Database

Steps to Follow

1. If you clicked on the "Go to resource" button mentioned in the previous section, you should see the webpage from the screenshot below. Alternatively, you can search for *<your4x4>-sql-server* in the search field. Alternatively, your new server should now show up on "SQL Servers" webpage from page 8.



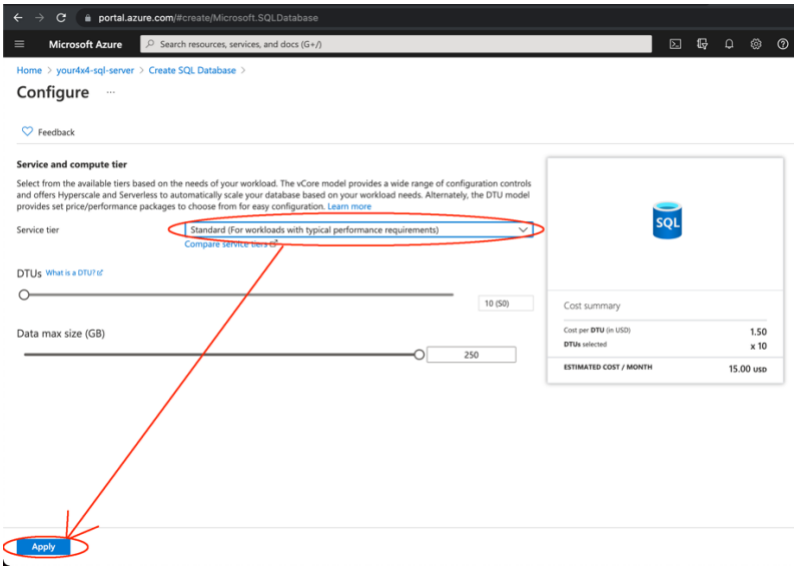
2. Click on "+ Create database" button. You should now see the panel below. Please pay some extra attention next.



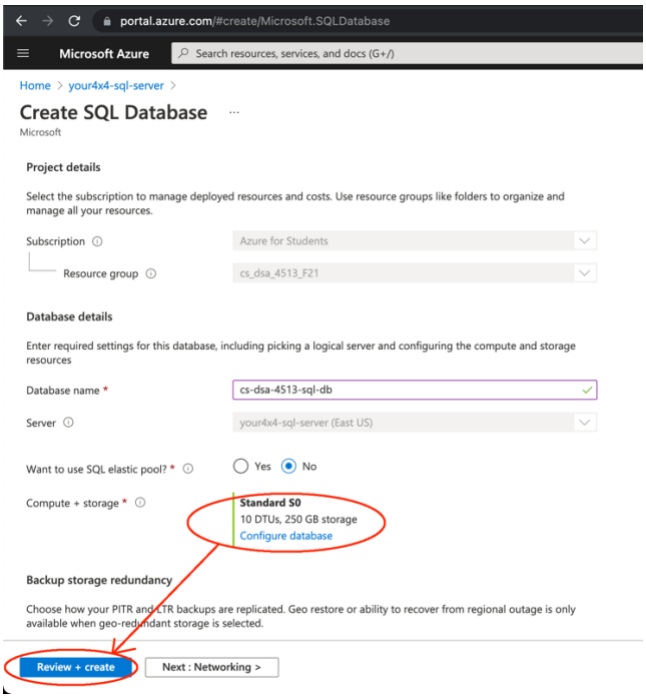
3. Type "*cs-dsa-4513-sql-db*" into "Database name" field.

4. **Click on "Configure database" link.**

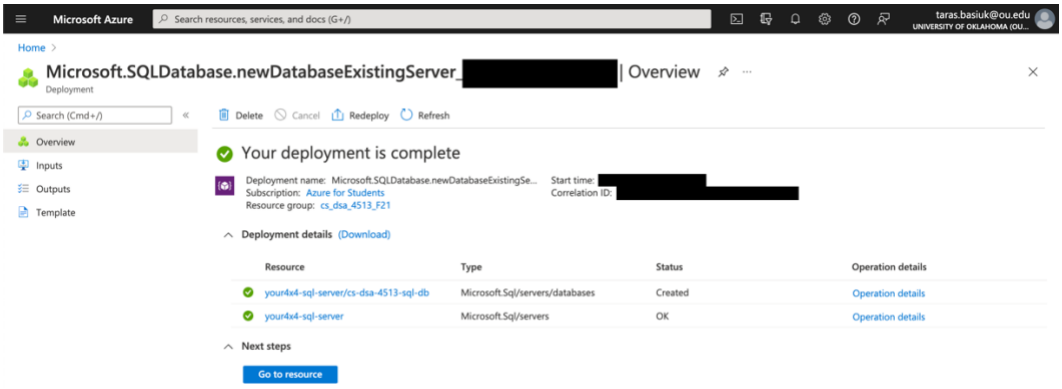
5. Once on the "Configure" page, select "Standard" under the Service Tier, then click "Apply".



6. Now your "SQL database" panel should look like below. Verify that "Compute + storage" no longer lists "Gen5, 2 vCores, 32 GB" configuration. Only then click on "Review + create" button. If you don't do the verification, your free credits will evaporate very quickly.



7. Clicking “Create” button on the next screen will start the deployment of your SQL database. Wait until it finishes. We will try to connect to your database in the next section.



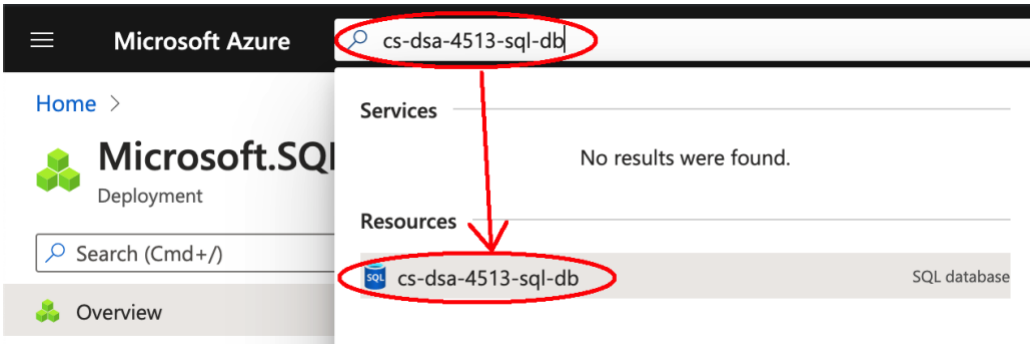
Possible Complications

If you’ve made a mistake while creating your SQL database, see the below instructions on how to navigate to it, delete it using the corresponding icon, and recreate it correcting for your mistake.

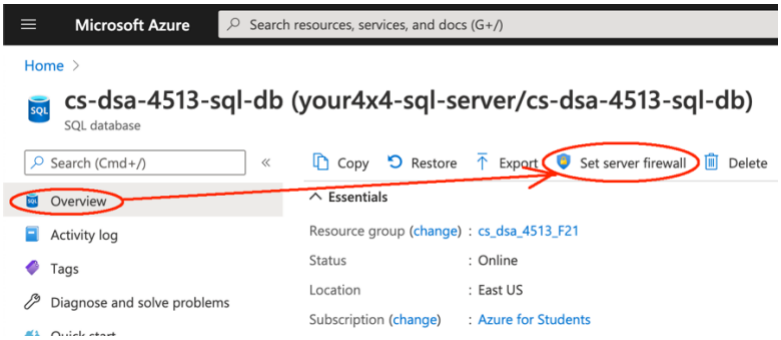
Connecting to Your SQL Database from Azure Portal

Firewall Configuration Steps

1. Enter "cs-dba-4513-sql-db" into the search field and select your SQL database (see the screenshot below). Alternatively, you should be able to find your database under the "Recent Resources" at https://portal.azure.com/#home

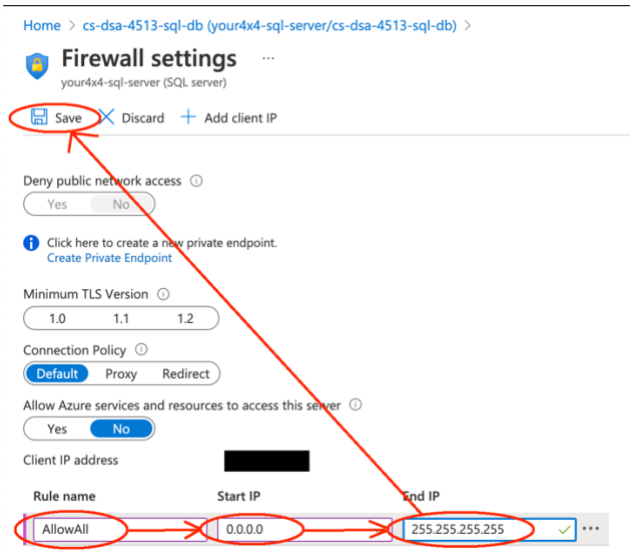


2. This should bring you to your new database overview seen below.
3. Now we need to create Firewall rules to allow outside connection to your DB. Click on "Overview" and then "Set server firewall"

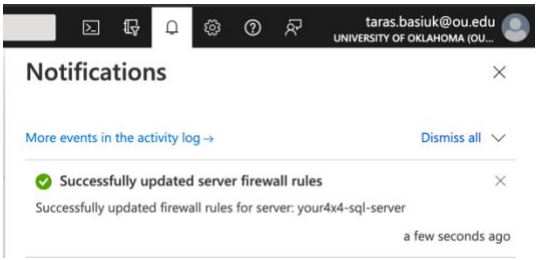


4. You should now see a panel corresponding to the screenshot below (with the input fields being empty).
    a. Type in "*AllowAll*" for "Rule Name" field.
    b. Type in "*0.0.0.0*" for "Start IP" field.
    c. Type in "*255.255.255.255*" for "End IP" field.

d. Click the “Save” icon.



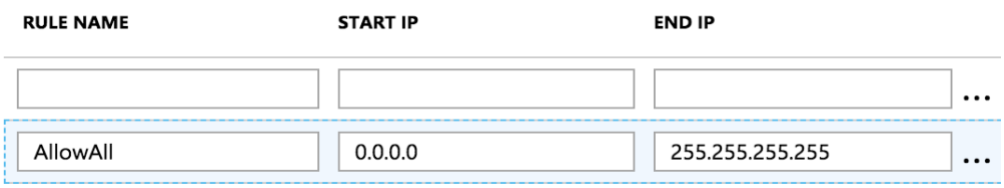
5. Wait for the updated firewall rules to be deployed.



6. !!!WARNING!!! Doing above allows connections to your SQL server from the entire internet. This should be fine for a class project. In general case, however, you would only allow connections from the IP ranges you trust. !!!WARNING!!!

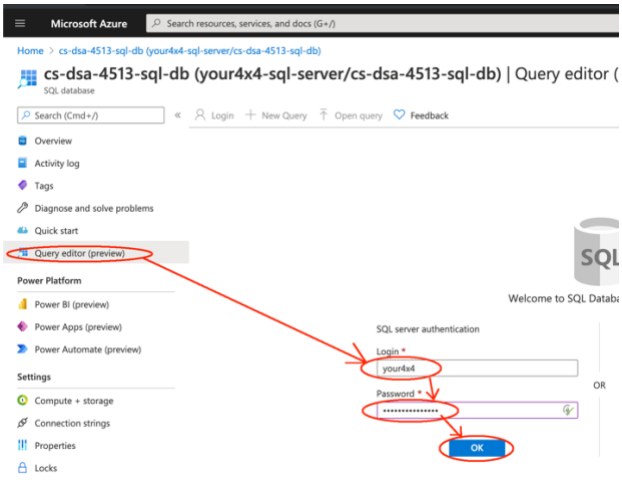
Possible Firewall Configuration Complications

If you’ve made a mistake while creating a new firewall rule, select it from the list, it will become editable, make appropriate changes, don't forget to click on the “Save” icon (see below).



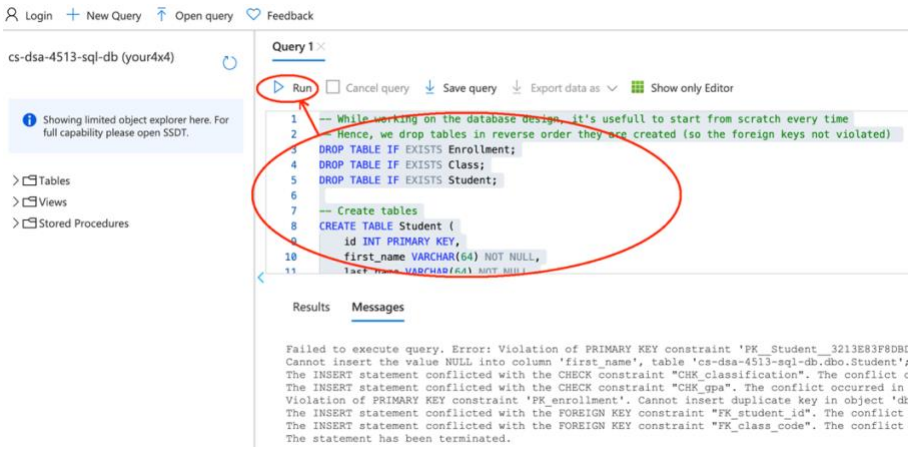
Connecting to the Database from Azure Portal

1. Navigate back to the database overview (from the search bar or azure portal home).
2. Click on "Query editor (preview)"
3. Use your SQL server admin login and password and click on "OK"

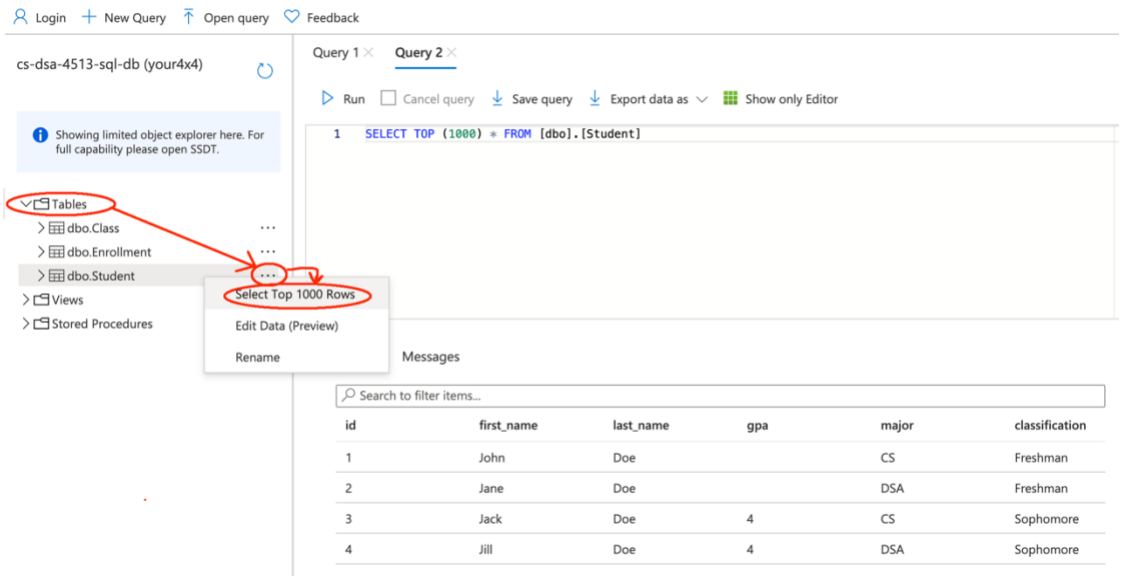


4. Run sample queries (see Appendix A) to make sure the database works (see the screenshot below, ignore the error messages, SQL in Appendix A is deliberately provoking those).
5. **!!!WARNING!!!** You will get logged out from the query editor automatically after some time passes (~30 minutes) and your queries will get discarded. Make sure to work on your queries in some outside environment (see the section below, for example) which persists them and only copy-paste them into the online editor immediately prior to execution.

!!!WARNING!!!



6. Whenever you have a table created with some data in it, you can select it in Tables folder and browse the data in it by using the “Select Top 1000 Rows” tab (you will be asked to acknowledge that you’re using a “preview” feature).

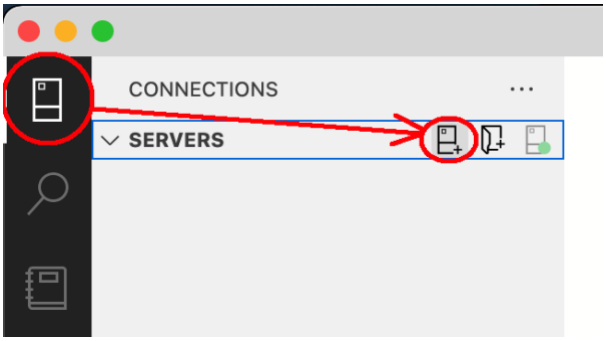


Connecting to Your SQL Database from Azure Data Studio IDE

You will need a development environment for working on your SQL queries with a bit more support than what Azure Portal Query Editor provides, we recommend you download & install Azure Data Studio.

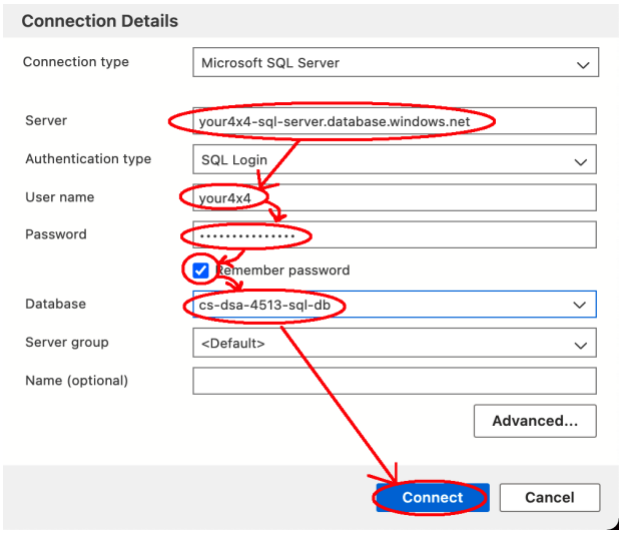
Steps to Follow

1. Download & Install Azure Data Studio according to the instructions here - https://docs.microsoft.com/en-us/sql/azure-data-studio/download
2. Launch Azure Data Studio, close "Welcome" tab.
3. Click on "Servers" icon to the top left, click on "Add Connection" button

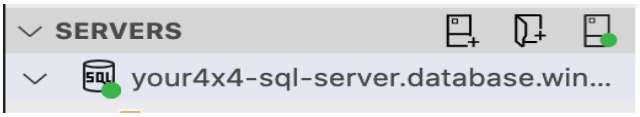


4. You should now see the "Connection" panel to the right, which has some mostly empty input fields at the bottom.
5. Fill out the input fields and make the selections as described below (see the screenshot)
    a. Server => "*<your4x4>-sql-server.database.windows.net*"
    b. Username => "*<your4x4>*"
    c. Password => "*<password you've chosen>*"

d. Database => Pick "*cs-dsa-4513-sql-db*" (option takes few seconds to appear, given other fields are typed-in correctly) .

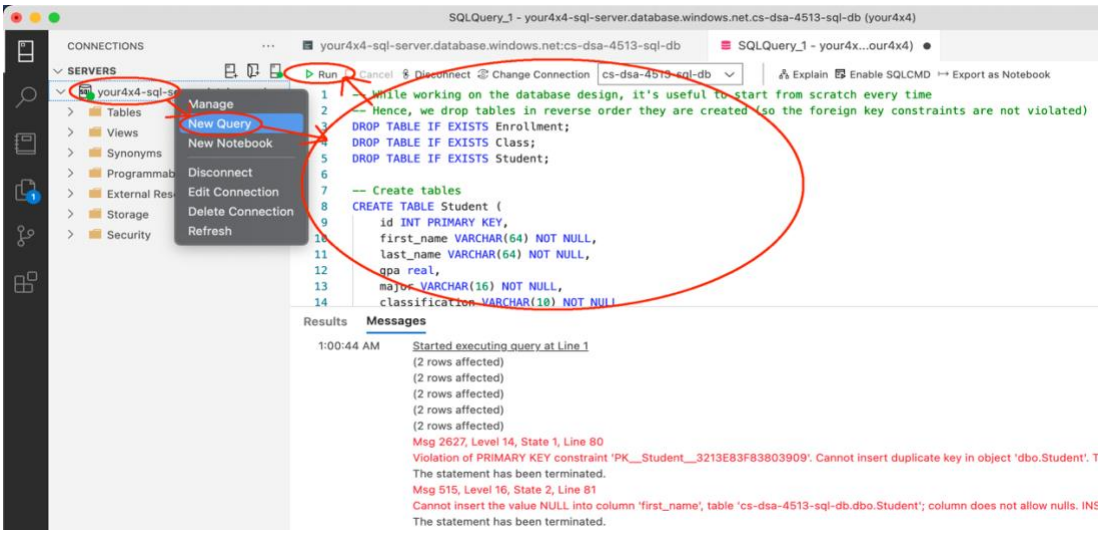


6. Click on "Connect" button. Once connected, your SQL server should be added to the servers list (see below).



7. You can now "right click" on your server and select "New Query". This will open a query editor for you to work with (see screenshot below).

8. You now should have everything you need to complete the Graded Homework #2.



Helpful Links

- Your Microsoft Azure Landing Page
    - https://portal.azure.com
- Azure Student Account Free Credit Balance
    - https://www.microsoftazuresponsorships.com/Balance
- Azure SQL Documentation (and especially helpful sections for this class, see the left tab)
    - https://docs.microsoft.com/en-us/sql/t-sql/language-reference
        - Statements
        - Queries
        - Data Types
        - Functions

Appendix A. Sample SQL Queries

*Below code is also available in a separate file on Canvas to help with the copy-paste issues

```
-- While working on the database design, it's useful to start from scratch every time
-- Hence, we drop tables in reverse order they are created (so the foreign key constraints are not violated)
DROP TABLE IF EXISTS Enrollment;
DROP TABLE IF EXISTS Class;
DROP TABLE IF EXISTS Student;

-- Create tables
CREATE TABLE Student (
    id INT PRIMARY KEY,
    first_name VARCHAR(64) NOT NULL,
    last_name VARCHAR(64) NOT NULL,
    gpa real,
    major VARCHAR(16) NOT NULL,
    classification VARCHAR(10) NOT NULL,

    -- Some examples of attribute value constraints
    CONSTRAINT CHK_gpa CHECK (gpa BETWEEN 0.0 AND 4.0),
    CONSTRAINT CHK_classification CHECK (classification IN ('Freshman', 'Sophomore', 'Junior', 'Senior'))
);

CREATE TABLE Class (
    code VARCHAR(10) PRIMARY KEY,
    name VARCHAR(64),
    description VARCHAR(1024)
);

CREATE TABLE Enrollment (
    student_id INT,
    class_code VARCHAR(10),
    semester VARCHAR(6) NOT NULL,
    year INT NOT NULL,
    grade CHAR(1),
```

```
    -- Multi-attribute primary key
    CONSTRAINT PK_enrollment PRIMARY KEY (student_id, class_code, semester, year),

    -- Some foreign key constraint examples
    CONSTRAINT FK_student_id FOREIGN KEY (student_id) REFERENCES Student,
    CONSTRAINT FK_class_code FOREIGN KEY (class_code) REFERENCES Class,

    CONSTRAINT CHK_semester CHECK (semester IN ('Spring', 'Fall')),
    CONSTRAINT CHK_grade CHECK (grade IN ('A', 'B', 'C', 'D', 'F'))
);

-- Insert some valid records

-- Students without GPA
INSERT INTO Student
    (id, first_name, last_name, major, classification)
VALUES
    (1, 'John', 'Doe', 'CS', 'Freshman'),
    (2, 'Jane', 'Doe', 'DSA', 'Freshman');

-- Students with GPA
INSERT INTO Student
VALUES
    (3, 'Jack', 'Doe', 4.0, 'CS', 'Sophomore'),
    (4, 'Jill', 'Doe', 4.0, 'DSA', 'Sophomore');

-- Classes
INSERT INTO Class
VALUES
    ('CS4513', 'Database Management Systems', 'Just read the sylabus.'),
    ('CS5513', 'Advanced Database Management Systems', NULL);

-- Enrollments without grades
INSERT INTO Enrollment
    (student_id, class_code, semester, year)
VALUES
```

```
    (1, 'CS4513', 'Fall', 2020),
    (2, 'CS4513', 'Fall', 2020);

-- Enrollments with grades
INSERT INTO Enrollment
VALUES
    (3, 'CS5513', 'Spring', 2020, 'A'),
    (4, 'CS5513', 'Spring', 2020, 'A');

-- Try inserting some invalid records (none should succeed)
INSERT INTO Student VALUES (1, 'Valid', 'Val', 4.0, 'CS', 'Sophomore'); -- PK Violation
INSERT INTO Student VALUES (5, NULL, 'Val', 4.0, 'CS', 'Sophomore'); -- NOT NULL Violation
INSERT INTO Student VALUES (5, 'Valid', 'Val', 4.0, 'CS', 'Master'); -- CHK_classification Violation
INSERT INTO Student VALUES (5, 'Valid', 'Val', 5.0, 'CS', 'Sophomore'); -- CHK_gpa Violation
INSERT INTO Enrollment VALUES (3, 'CS5513', 'Spring', 2020, 'A'); -- PK_enrollment violation
INSERT INTO Enrollment VALUES (10, 'CS5513', 'Spring', 2020, 'A'); -- FK_student_id violation
INSERT INTO Enrollment VALUES (3, 'CS1234', 'Spring', 2020, 'A'); -- FK_class_code violation

-- Run some sample SELECT queries
SELECT * FROM Student; -- Get everything stored in Student table
SELECT description FROM Class WHERE code = 'CS4513'; -- Get the description of the DBMS class

-- Get the average GPA of students for each classification
SELECT classification, avg(gpa)
FROM Student
GROUP BY classification;

-- Get the names of students who ever got an A in CS5513 class
SELECT first_name, last_name
FROM Student, Enrollment
WHERE
    Student.id = Enrollment.student_id AND
    Enrollment.class_code = 'CS5513' AND
    Enrollment.grade = 'A';

-- Run some UPDATE queries AND DELETE queries
```

```
UPDATE Enrollment SET grade = 'B' WHERE student_id = 1 AND class_code = 'CS4513' AND semester = 'Fall' AND year = 2020;
DELETE FROM Enrollment WHERE student_id = 2 AND class_code = 'CS4513' AND semester = 'Fall' AND year = 2020;
SELECT * FROM Enrollment; -- Verify results
```