# HU Extension      Assignment 05      E-90 Cloud Computing

Handed out: 10/03/2015                Due by 11:59 PM EST on Friday, 10/09/2015

Place all of your narratives and illustrations in a single Word or PDF document named E90_LastNameFirstNameHW05.docx [.pdf]. Use this assignment as the initial template. Add your steps and your code below problem statements used for that problem. Upload your homework file and your working code (e.g., filename.java) into your Assignment 5 folder.  Do not include executables.

**Problem 1. MySQL Community Edition.**
Download and install MySQL Community Edition server to your local machine.
Create a new database (your local MySQL database) within that server
Create a new user that has all the privileges on that database.
Create table FLOWERS
         name of every flower,
         typical height in inches
         and a brief description of the flower.
Populate your table with information on 3 flowers.
Query the data.
Delete a record.
Close the database connection.
**[15  points]**

**Problem 2. Use any programming language (e.g., Java, .NET, Python, …).**
Starting with the attached classes `PersonDAO.java` and `MySQLAccess.java` we used in class, write new Java classes or classes in any other language of your choice that will populate and query existing table FLOWERS in your local MySQL database.  Close the database connection.
 **[20 points]**


Write new Java classes (Starting with `PersonDAO.java` and `MySQLAccess.java`) to:
Populate existing table FLOWERS in your local MySQL database
Query existing table FLOWERS in your local MySQL database
Close the database connection.

**Problem 3**. **Use MySQL Client and any programming language (e.g., Java, .NET, Python, …).**
Manually delete table FLOWERS **(MySQL Client)** and then extend the program from Problem 2 (in the language of your choice) so that when connected to the database it can recreate table FLOWERS and perform: insert a few records, query the data, delete a record. Close the database connection.
**[15 points]**

Manually delete table FLOWERS **(MySQL Client)**
Extend the program from Problem 2 to:
  Connect to the database
  Recreate table FLOWERS
  Insert a few records
  Query the data
  Delete a record
  Close the database connection

**Problem 4. Use AWS Console and MySQL Client.**
1. Create new VPC with two subnets (one public and one private). Create a new security group associated with your new VPC, and adjust permissions (CIDR) of that security group so that you can access future RDS from your local machine.
2. Create a new micro instance of MySQL database in Amazon RDS service – using AWS Console.
3. Demonstrate that you can connect to the remote RDS database using your local MySQL client.
**[15 points]**

1. Create new VPC with two subnets (one public and one private)
2. Create a new security group associated with your new VPC
3. Adjust permissions (CIDR) of that security group so that you can access future RDS from your local machine.
4. Create a new micro instance of MySQL database in Amazon RDS service – using AWS Console.
5. Demonstrate that you can connect to the remote RDS database using your local MySQL client.

**Problem 5. Use any programming language (e.g., Java, .NET, Python, …).**
1. Modify the code developed in Problem 3 so that it can now connect to your Amazon's RDS database instance that you created in problem 4.
2. Demonstrate that your client code could perform on remote RDS database instance the same operations it performed on the local database (create table, insert a few records, query the data, delete a record (on your remote RDS database instance). Close the database connection.

**[15 points]**


1. Modify the code developed in Problem 3 so that it can now connect to your Amazon's RDS database instance that you created in problem 4

2. On remote RDS database instance:
   a. create table
   b. insert a few records
   c. query the data
   d. delete a record

3. Close the database connection


**Problem 6. Use AWS SDK for any programming language (e.g., Java, .NET, Python, …).**
Write a program that when given a bucket name will go out to S3 and wipe that bucket out. That program has to locate all objects in all the folders, first delete those objects, then delete those folders and finally delete the bucket. Try first with a single S3 Bucket folder with several objects (files). Finally write a program that could handle an arbitrarily nested folder structure with objects of any S3 bucket.

**[20 points]**

Write a program that when given a bucket name will go out to S3 and wipe a bucket out that contains a bucket with a single folder with several objects.

1. Locate all objects in all the folders
2. Delete those objects
3. Delete those folders
4. Delete the bucket

Write a program that when given a bucket name with an arbitrarily nested folder structure can wipe that bucket out.

1. Locate all objects in all the folders
2. Delete those objects
3. Delete those folders
4. Delete the bucket