

Course Project

DeVry University

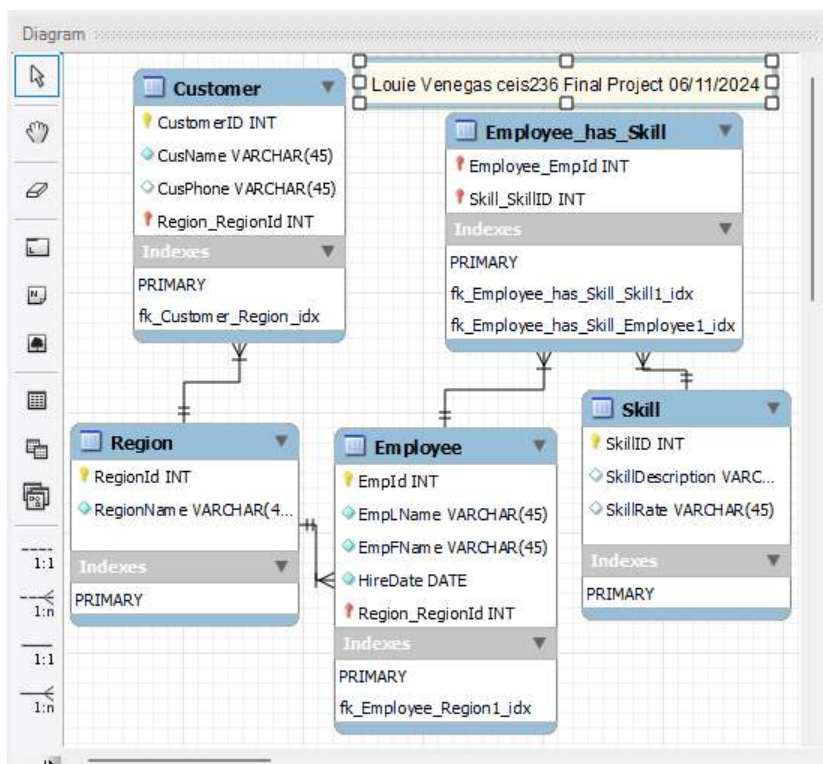
College of Engineering and Information Sciences

Course Number: CEIS236

Course Project Deliverable: Final Course Project

Include screenshots of the code and result

Problem 1



Problem 2

-- MySQL Workbench Forward Engineering

```
SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS,
FOREIGN_KEY_CHECKS=0;
SET @OLD_SQL_MODE=@@SQL_MODE,
SQL_MODE='ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO
_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION';
```

-- Schema ceis236finalproject

-- Schema ceis236finalproject

```
CREATE SCHEMA IF NOT EXISTS `ceis236finalproject` DEFAULT CHARACTER
SET utf8 ;
USE `ceis236finalproject` ;
```

-- Table `ceis236finalproject`.`Region`

```
CREATE TABLE IF NOT EXISTS `ceis236finalproject`.`Region` (
  `RegionId` INT NOT NULL,
  `RegionName` VARCHAR(45) NOT NULL,
  PRIMARY KEY (`RegionId`))
ENGINE = InnoDB;
```

-- Table `ceis236finalproject`.`Customer`

```
CREATE TABLE IF NOT EXISTS `ceis236finalproject`.`Customer` (
  `CustomerID` INT NOT NULL,
  `CusName` VARCHAR(45) NOT NULL,
  `CusPhone` VARCHAR(45) NULL,
  `Region_RegionId` INT NOT NULL,
  PRIMARY KEY (`CustomerID`, `Region_RegionId`),
  INDEX `fk_Customer_Region_idx` (`Region_RegionId` ASC) VISIBLE,
  CONSTRAINT `fk_Customer_Region`
    FOREIGN KEY (`Region_RegionId`)
      REFERENCES `ceis236finalproject`.`Region` (`RegionId`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;
```

```

-----
-- Table `ceis236finalproject`.`Employee`
-----
CREATE TABLE IF NOT EXISTS `ceis236finalproject`.`Employee` (
  `EmpId` INT NOT NULL,
  `EmpLName` VARCHAR(45) NOT NULL,
  `EmpFName` VARCHAR(45) NOT NULL,
  `HireDate` DATE NOT NULL,
  `Region_RegionId` INT NOT NULL,
  PRIMARY KEY (`EmpId`, `Region_RegionId`),
  INDEX `fk_Employee_Region1_idx` (`Region_RegionId` ASC) VISIBLE,
  CONSTRAINT `fk_Employee_Region1`
    FOREIGN KEY (`Region_RegionId`)
      REFERENCES `ceis236finalproject`.`Region` (`RegionId`)
      ON DELETE NO ACTION
      ON UPDATE NO ACTION)
ENGINE = InnoDB;

```

```

-----
-- Table `ceis236finalproject`.`Skill`
-----
CREATE TABLE IF NOT EXISTS `ceis236finalproject`.`Skill` (
  `SkillID` INT NOT NULL,
  `SkillDescription` VARCHAR(45) NULL,
  `SkillRate` VARCHAR(45) NULL,
  PRIMARY KEY (`SkillID`))
ENGINE = InnoDB;

```

```

-----
-- Table `ceis236finalproject`.`Employee_has_Skill`
-----
CREATE TABLE IF NOT EXISTS
`ceis236finalproject`.`Employee_has_Skill` (
  `Employee_EmpId` INT NOT NULL,
  `Skill_SkillID` INT NOT NULL,
  PRIMARY KEY (`Employee_EmpId`, `Skill_SkillID`),
  INDEX `fk_Employee_has_Skill_Skill1_idx` (`Skill_SkillID` ASC)
  VISIBLE,
  INDEX `fk_Employee_has_Skill_Employee1_idx` (`Employee_EmpId`
  ASC) VISIBLE,
  CONSTRAINT `fk_Employee_has_Skill_Employee1`
    FOREIGN KEY (`Employee_EmpId`)
      REFERENCES `ceis236finalproject`.`Employee` (`EmpId`)
      ON DELETE NO ACTION

```

```

    ON UPDATE NO ACTION,
    CONSTRAINT `fk_Employee_has_Skill_Skill1`
    FOREIGN KEY (`Skill_SkillID`)
    REFERENCES `ceis236finalproject`.`Skill` (`SkillID`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;

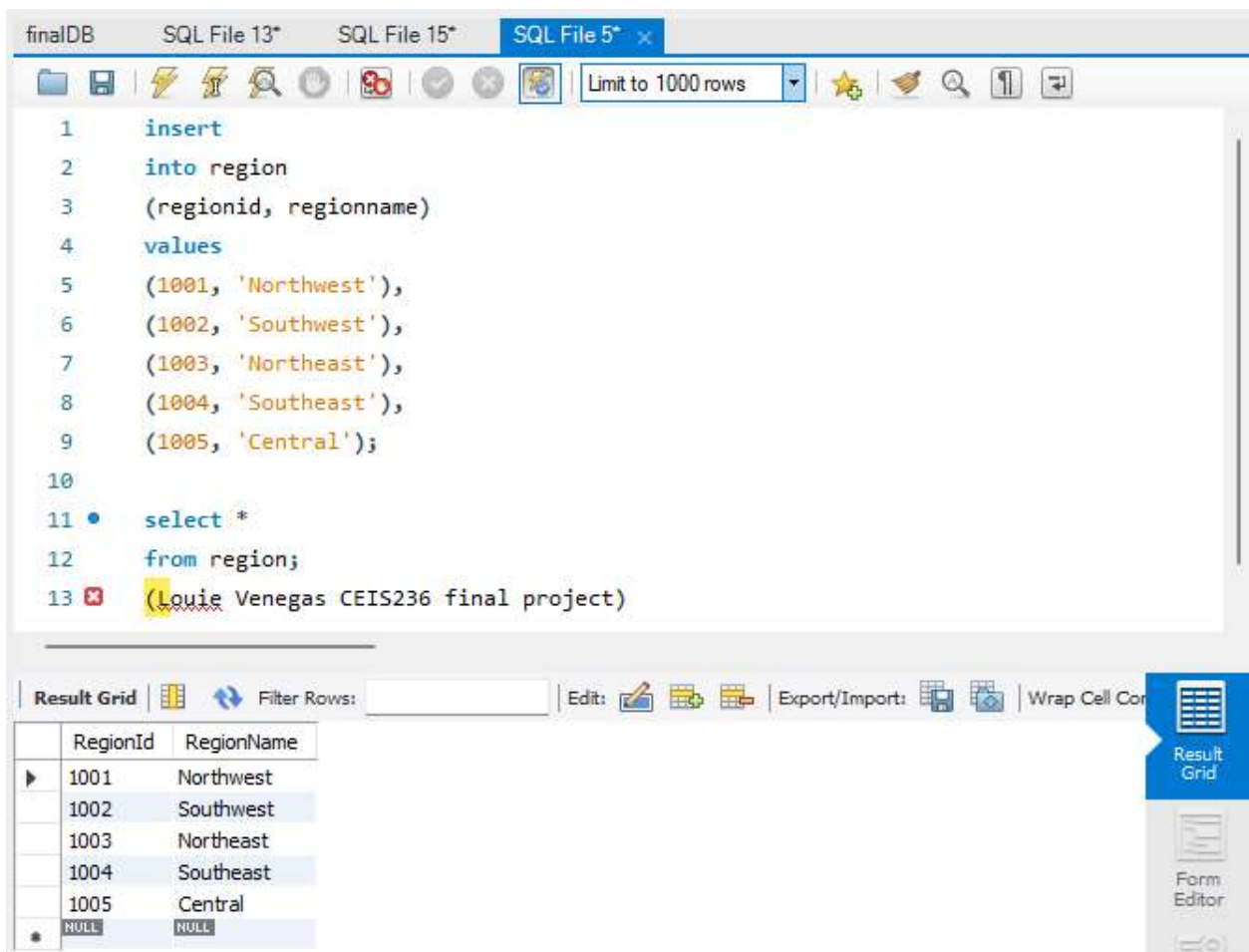
```

```

SET SQL_MODE=@OLD_SQL_MODE;
SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;

```

Problem 3



The screenshot shows a SQL IDE window with multiple tabs. The active tab is 'SQL File 5*'. The SQL script in the editor is as follows:

```

1  insert
2  into region
3  (regionid, regionname)
4  values
5  (1001, 'Northwest'),
6  (1002, 'Southwest'),
7  (1003, 'Northeast'),
8  (1004, 'Southeast'),
9  (1005, 'Central');
10
11 • select *
12 from region;
13 (Louie Venegas CEIS236 final project)

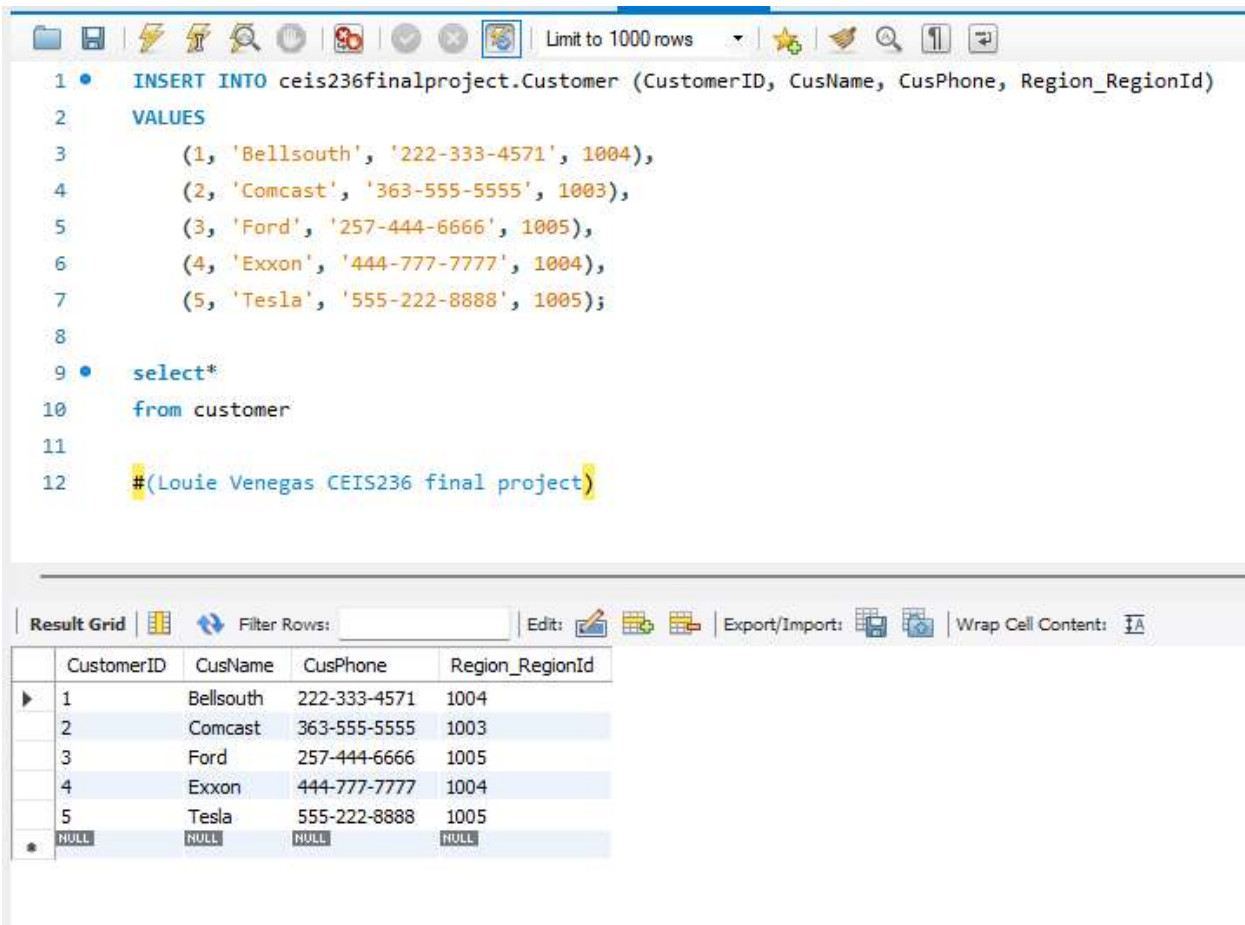
```

Below the script, the 'Result Grid' is displayed, showing the output of the SQL query. The grid has two columns: 'RegionId' and 'RegionName'. The data is as follows:

RegionId	RegionName
1001	Northwest
1002	Southwest
1003	Northeast
1004	Southeast
1005	Central
NULL	NULL

The IDE interface includes a toolbar with various icons for file operations, a 'Limit to 1000 rows' dropdown, and a sidebar on the right with buttons for 'Result Grid', 'Form Editor', and other tools.

Problem 4



The screenshot shows a SQL IDE interface. The top toolbar includes icons for file operations, execution, and search, along with a "Limit to 1000 rows" dropdown. The SQL editor contains the following script:

```
1 • INSERT INTO ceis236finalproject.Customer (CustomerID, CusName, CusPhone, Region_RegionId)
2   VALUES
3     (1, 'Bellsouth', '222-333-4571', 1004),
4     (2, 'Comcast', '363-555-5555', 1003),
5     (3, 'Ford', '257-444-6666', 1005),
6     (4, 'Exxon', '444-777-7777', 1004),
7     (5, 'Tesla', '555-222-8888', 1005);
8
9 • select*
10  from customer
11
12  #(Louie Venegas CEIS236 final project)
```

Below the editor is the "Result Grid" section, which includes a "Filter Rows:" input field and buttons for "Edit:", "Export/Import:", and "Wrap Cell Content:". The results are displayed in a table with 5 columns: CustomerID, CusName, CusPhone, and Region_RegionId.

	CustomerID	CusName	CusPhone	Region_RegionId
▶	1	Bellsouth	222-333-4571	1004
	2	Comcast	363-555-5555	1003
	3	Ford	257-444-6666	1005
	4	Exxon	444-777-7777	1004
	5	Tesla	555-222-8888	1005
*	NULL	NULL	NULL	NULL

Limit to 1000 rows

```
1 • INSERT INTO ceis236finalproject.Customer (CustomerID, CusName, CusPhone, Region_RegionId)
2   VALUES
3     (1, 'Bellsouth', '222-333-4571', 1004),
4     (2, 'Comcast', '363-555-5555', 1003),
5     (3, 'Ford', '257-444-6666', 1005),
6     (4, 'Exxon', '444-777-7777', 1004),
7     (5, 'Tesla', '555-222-8888', 1005);
8
9 • select*
10  from customer
11
12  #(Louie Venegas CEIS236 final project)
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: [IA](#)

	CustomerID	CusName	CusPhone	Region_RegionId
▶	1	Bellsouth	222-333-4571	1004
	2	Comcast	363-555-5555	1003
	3	Ford	257-444-6666	1005
	4	Exxon	444-777-7777	1004
	5	Tesla	555-222-8888	1005
*	NULL	NULL	NULL	NULL

Problem 5

finalDB
SQL File 13*
SQL File 15*
SQL File 5*
SQL File 6*
SQL File 7* x

Limit to 1000 rows

```

1 • INSERT INTO ceis236finalproject.Employee (EmpID, EmplName, EmpFName, HireDate, Region_RegionId)
2   VALUES
3     ('E1', 'Louie', 'Venegas', '2024-6-11', 1004),
4     ('E2', 'Craig', 'Brett', '2019-2-7', 1004),
5     ('E3', 'Williams', 'Josh', '2019-3-30', 1003),
6     ('E4', 'Cope', 'Leslie', '2017-4-21', 1002),
7     ('E5', 'Mudd', 'Roger', '2007-10-18', 1001);
8
9 • select*
10  from employee
11
12  #(Louie Venegas CEIS236 final project)

```

Result Grid

Filter Rows:
Edit:

Export/Import:

Wrap Cell Content:

	EmpId	EmplName	EmpFName	HireDate	Region_RegionId
▶	E1	Louie	Venegas	2024-06-11	1004
	E2	Craig	Brett	2019-02-07	1004
	E3	Williams	Josh	2019-03-30	1003
	E4	Cope	Leslie	2017-04-21	1002
	E5	Mudd	Roger	2007-10-18	1001
*	NULL	NULL	NULL	NULL	NULL

Problem 6

```
finalDB SQL File 13* SQL File 15* SQL File 5* SQL File 6* SQL File 7* SQL File 8* x
1 • INSERT INTO ceis236finalproject.Skill (SkillID, SkillDescription, SkillRate)
2 VALUES
3     ('S1', 'Data Entry I', 20),
4     ('S2', 'Database Admin I', 35),
5     ('S3', 'Database Admin II', 50),
6     ('S4', 'Database Architect', 65),
7     ('S5', 'Java I', 40),
8     ('S6', 'Java II', 65),
9     ('S7', 'Python I', 35),
10    ('S8', 'Python II', 55);
11
12 • select*
13 from skill;
14
15 #(Louie Venegas CEIS236 final project)
```

Result Grid		 Filter Rows:	Edit: 			Export/Import: 		Wrap Cell Content: 
	SkillID	SkillDescription	SkillRate					
▶	S1	Data Entry I	20					
	S2	Database Admin I	35					
	S3	Database Admin II	50					
	S4	Database Architect	65					
	S5	Java I	40					
	S6	Java II	65					
	S7	Python I	35					
	S8	Python II	55					
✱	NULL	NULL	NULL					

Problem 7

finalDB SQL File 13* SQL File 15* SQL File 5* SQL File 6* SQL File 7* SQL File 8* SQL File 9*

Limit to 1000 rows

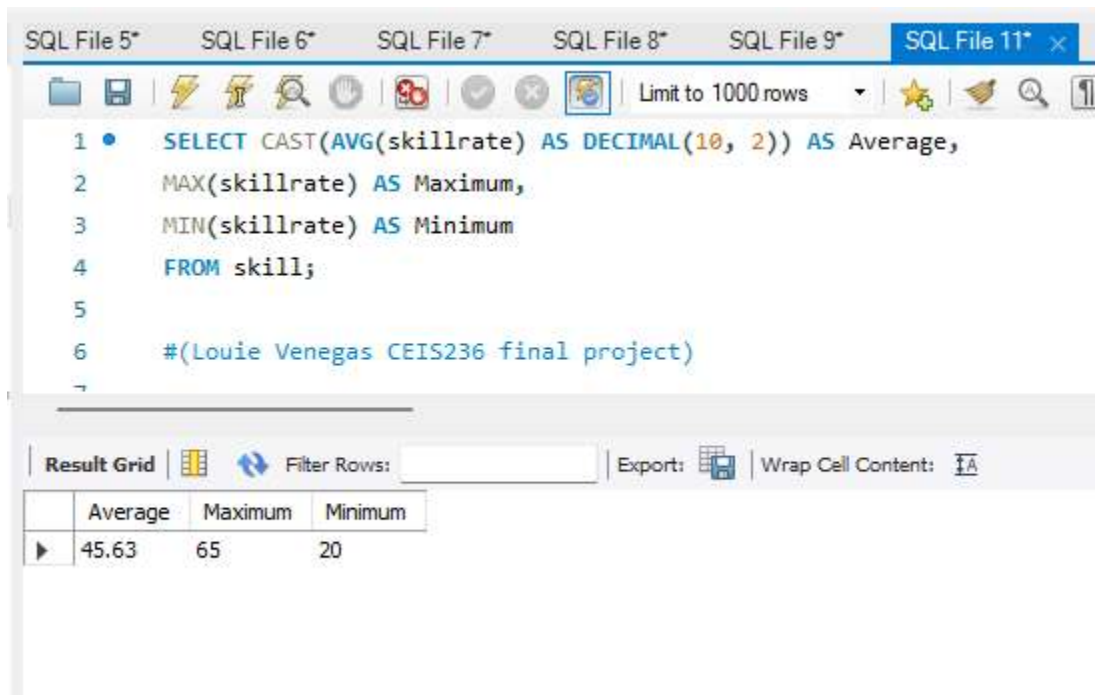
```
1 • INSERT INTO ceis236finalproject.EmpSkill (Employee_EmpID, Skill_SkillID)
2   VALUES
3     ('E1', 1004),
4     ('E2', 1003),
5     ('E3', 1005),
6     ('E4', 1004),
7     ('E5', 1005);
8
9 • select*
10  from empSkill;
11
12  #(Louie Venegas CEIS236 final project)
```

Result Grid

	SkillID	SkillDescription	SkillRate
▶	S1	Data Entry I	20
	S2	Database Admin I	35
	S3	Database Admin II	50
	S4	Database Architect	65
	S5	Java I	40
	S6	Java II	65
	S7	Python I	35
	S8	Python II	55
*	NULL	NULL	NULL

Edit: Export/Import: Wrap Cell Content:

Problem 8



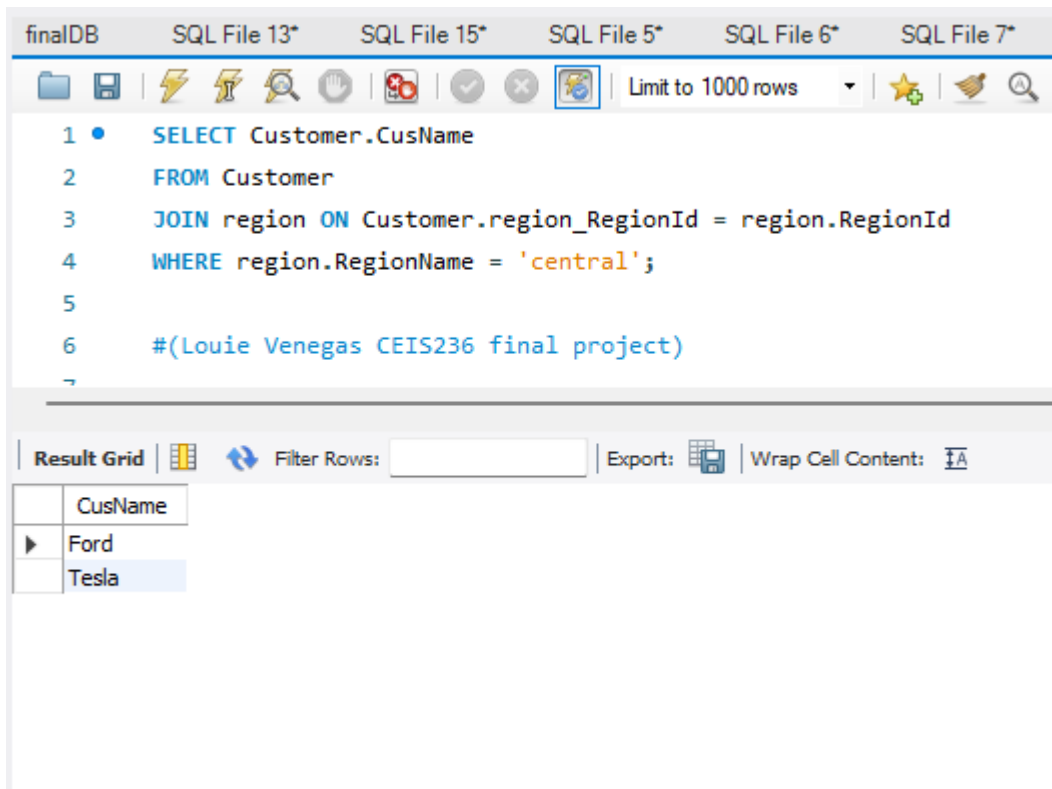
The screenshot shows a SQL IDE with multiple tabs. The active tab is 'SQL File 11*'. The query editor contains the following SQL code:

```
1 • SELECT CAST(AVG(skillrate) AS DECIMAL(10, 2)) AS Average,  
2     MAX(skillrate) AS Maximum,  
3     MIN(skillrate) AS Minimum  
4     FROM skill;  
5  
6     #(Louie Venegas CEIS236 final project)  
7
```

Below the query editor is the 'Result Grid' section. It includes a 'Filter Rows' input field, an 'Export' button, and a 'Wrap Cell Content' checkbox. The results are displayed in a table with the following data:

	Average	Maximum	Minimum
▶	45.63	65	20

Problem 9

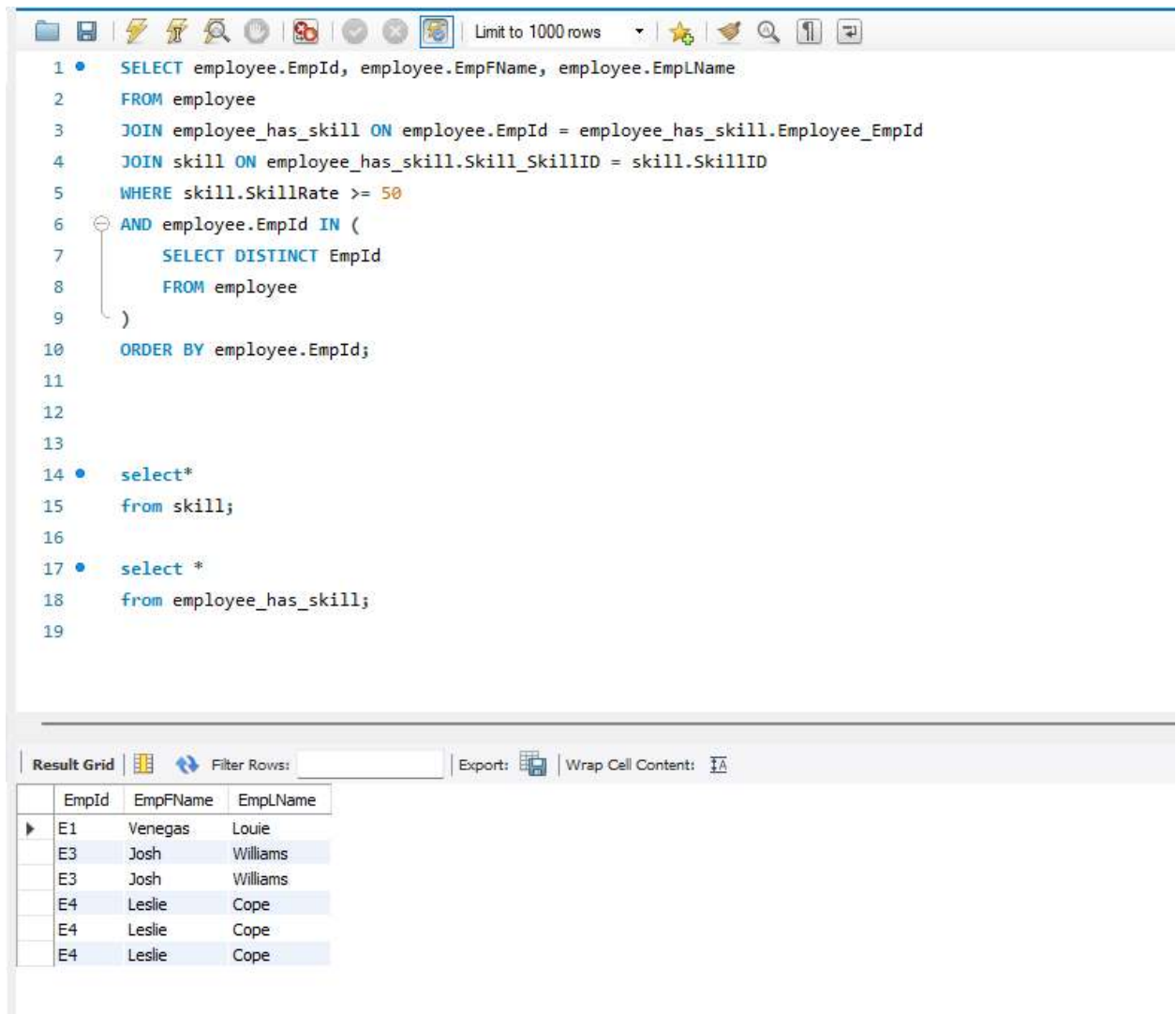


The screenshot shows a SQL IDE interface with a toolbar at the top containing icons for file operations, execution, and search. Below the toolbar, a query is displayed in a text editor. The query is a SQL SELECT statement that retrieves the names of customers from the 'Customer' table, joined with the 'region' table, where the region name is 'central'. The query is numbered 1 through 7. Below the query editor, there is a 'Result Grid' section. It includes a 'Filter Rows' input field, an 'Export' button, and a 'Wrap Cell Content' checkbox. The result grid itself is a table with two columns: 'CusName' and a list of customer names: 'Ford' and 'Tesla'.

```
1 • SELECT Customer.CusName
2 FROM Customer
3 JOIN region ON Customer.region_RegionId = region.RegionId
4 WHERE region.RegionName = 'central';
5
6 #(Louie Venegas CEIS236 final project)
7
```

	CusName
▶	Ford
	Tesla

Problem 10



The screenshot shows a SQL IDE interface. The top toolbar includes icons for file operations, execution, and a 'Limit to 1000 rows' dropdown. The main area contains a SQL query with line numbers 1 through 19. The query is as follows:

```
1 • SELECT employee.EmpId, employee.EmpFName, employee.EmpLName
2 FROM employee
3 JOIN employee_has_skill ON employee.EmpId = employee_has_skill.Employee_EmpId
4 JOIN skill ON employee_has_skill.Skill_SkillID = skill.SkillID
5 WHERE skill.SkillRate >= 50
6 AND employee.EmpId IN (
7     SELECT DISTINCT EmpId
8     FROM employee
9 )
10 ORDER BY employee.EmpId;
11
12
13
14 • select*
15 from skill;
16
17 • select *
18 from employee_has_skill;
19
```

Below the query editor is the 'Result Grid' section. It includes a 'Filter Rows' input field, an 'Export' button, and a 'Wrap Cell Content' checkbox. The result grid displays the following data:

	EmpId	EmpFName	EmpLName
▶	E1	Venegas	Louie
	E3	Josh	Williams
	E3	Josh	Williams
	E4	Leslie	Cope
	E4	Leslie	Cope
	E4	Leslie	Cope

Problem 11

finalDB SQL File 13* SQL File 15* SQL File 5* SQL File 6* SQL File 7* SQL File 8* SQL File 9* SQL File 11* SQL

Limit to 1000 rows

```
1 • SELECT employee.EmpId, employee.EmpFName, employee.EmpLName, skill.SkillDescription, skill.SkillRate
2 FROM employee
3 JOIN employee_has_skill ON employee.EmpId = employee_has_skill.Employee_EmpId
4 JOIN skill ON employee_has_skill.Skill_SkillID = skill.SkillID
5 AND employee.EmpId IN (
6     SELECT DISTINCT EmpId
7     FROM employee
8 )
9 ORDER BY employee.EmpId;
```

Result Grid Filter Rows: Export: Wrap Cell Content:

	EmpId	EmpFName	EmpLName	SkillDescription	SkillRate
▶	E1	Venegas	Louie	Database Admin I	35
	E1	Venegas	Louie	Java I	40
	E1	Venegas	Louie	Python II	55
	E2	Brett	Craig	Data Entry I	20
	E3	Josh	Williams	Database Architect	65
	E3	Josh	Williams	Java II	65
	E3	Josh	Williams	Python I	35
	E4	Leslie	Cope	Database Architect	65
	E4	Leslie	Cope	Java II	65
	E4	Leslie	Cope	Python II	55