Name:_____

- Due date: by 11:59pm on **April 27, 2019**
- **Late penalty: 10 points/week after due date. The maximum score will be 60 after the answers are given.**
- **No resubmission after receiving the score. You MUST completely test your assignments before the deadline.**
- Please write stored procedures and stored functions based on the **CPS3740** database.
- You must create and test your programs on database **CPS3740_2019S** at server **imc.kean.edu**.
- Your program names must EXACTLY follow each question format. The program name is case sensitive.
- XXXX means your email ID in the test case examples.
- The test cases in each question provide as the testing SQL queries with the corresponding results. You should test your programs with different input values and make sure the results are correct.
- The title of your output should also match the test cases.

1. (**20 points**) Write a **stored function** named **fHW2_1_XXXX (…)** based on the **Staff** table. Your program should take one argument **amount** and find the total number of staff who has salary >= amount. Your program should meet the following test cases.

1A. _____ (**5 points**) If the given **amount** is **NULL**, please return an error message.

mysql> select fHW2_1_XXXX(null) as output;

```
+------------------------------+
| output                       |
+------------------------------+
| Error! Amount cannot be NULL |
+------------------------------+
```

1B. _____ (**5 points**) If the given **amount** is not NULL, your program will the following message if none of the staff has salary >= amount.

mysql> select fHW2_1_XXXX(50000);

```
+------------------+
| output           |
+------------------+
| No record found! |
+------------------+
```

1C. _____ (**10 points**) If the given **amount** is not NULL, your program will the following message if there is one or more staff who has salary >= amount.

mysql> select fHW2_1_XXXX(10000) as output;

```
+----------------------------------------+
| output                                 |
+----------------------------------------+
| There are 5 staff who has salary >=10000 |
+----------------------------------------+
```

2. (20 points) Write a **stored procedure** named **pHW2_2_XXXX (…)** based on the **Staff** and **Branch** tables. Your program should take one argument **city**, and the program should meet the following test cases. The output header must also match each test case.

2**A**. _____ (**5 points**) If the given **city is empty or null**, please display the following error message.

mysql> call pHW2_2_XXXX(NULL);
mysql> call pHW2_2_XXXX('');

```
+---------------------------------+
| message                         |
+---------------------------------+
| Please input a valid city name. |
+---------------------------------+
```

2B. _____ (**5 points**) If the given **city** is NOT in the **Branch** table, please display the following message.

mysql> call pHW2_2_XXXX('Edison');

```
+--------------------------------------+
| message                              |
+--------------------------------------+
| No branchno found in the city: Edison |
+--------------------------------------+
```

2C. _____ (**10 points**) If the given **city** is in the **Branch** table, please display the number of staff for each gender in each office at the given city based on the Staff table.

mysql> call pHW2_2_XXXX('London');

```
+----------+------+------+
| branchno | sex  | myct |
+----------+------+------+
| B002     | M    |    1 |
| B005     | F    |    1 |
| B005     | M    |    1 |
+----------+------+------+
```

3. (20 points) Write a **stored procedure** named **pHW2_3_XXXX (…)** based on the **Guest** table. It should take one IN argument **city** and have one OUT argument **result**. Your program should meet the following test cases.

3A. _____ (**6 points**) If the given **city** is empty or NULL, please display "Please input a valid city." **mysql> call pHW2_3_XXXX(NULL,@result); select @result;** **mysql> call pHW2_3_XXXX('',@result); select @result;**	+----------------------------+ | @result | +----------------------------+ | Please input a valid city. | +----------------------------+
3B. _____ (**9 points**) If the given **city** is in the **Guest** table, please display guest **names** for all guests whose addresses has pattern matching the **given city**. **mysql> call pHW2_3_XXXX('London', @result); select @result;**	+-----------------------+ | @result | +-----------------------+ | John Kay,Mike Ritchie | +-----------------------+
3C. _____ (**5 points**) If the given **city** xyz is not in the **Guest** table, please display "NO people found for city: xyz". DO NOT hardcode the city name in your output. **mysql> call pHW2_3_xxxx('xyz',@result); select @result;**	Your output should clearly indicate the input city name. +-----------------------------+ | @result | +-----------------------------+ | NO people found for city:xyz | +-----------------------------+

4. (20 points) Write a **stored function** named **fHW2_4_XXXX (…)** based on the **Guest** table. Your program should take one argument **city** and it should meet the following test cases.

4A. _____ (**9 points**) If the given city is in the **Guest** table, please return text with **names** for all guests whose address includes the **given city**. **mysql> select fHW2_4_xxxx('London') as output;**	+-----------------------+ | output | +-----------------------+ | John Kay,Mike Ritchie | +-----------------------+
4B. _____ (**6 points**) If the given **city** is empty or NULL, please display "Please input a valid city." **mysql> select fHW2_4_xxxx('') as output;** **mysql> select fHW2_4_xxxx(NULL) as output;**	+----------------------------+ | output | +----------------------------+ | Please input a valid city. | +----------------------------+
4C. _____ (**5 points**) If the given **city** is not in the **Guest** table, please display "No result found". **mysql> select fHW2_4_xxxx('Union') as output;**	+-----------------+ | output | +-----------------+ | No result found | +-----------------+

5. (20 points) Write a **stored function** named **fHW2_5_XXXX (…)** that should take one argument integer **N** and it should meet the following test cases and the output examples.

5A. _____ (**5 points**) If the given N <=0, return an error message. **mysql> select fHW2_5_xxxx(-1) as output;**	+---------------------------------+ | output | +---------------------------------+ | Please input a positive number. | +---------------------------------+
5B. _____ (**7 points**) If the given N > 0 and N<=5, return the string 1+2+3+…+N and the sum. **mysql> select fHW2_5_xxxx(4) as output;**	+--------------+ | output | +--------------+ | 1+2+3+4=10 | +--------------+
5C. _____ (**8 points**) If the given N>5, return the string 1+2+…+(N-1)+N and the sum. The string should contain the first 2 and last 2 numbers, and "+"and "…" between the numbers. **mysql> select fHW2_5_xxxx(8) as output;**	+----------------+ | output | +----------------+ | 1+2+..+7+8=36 | +----------------+