Exercises on Stored Procedures and Stored functions and Cursors:

1) Create stored procedure **usp_get_employees_salary_above** that **accept a number** as parameter and return **all employees' first and last names** whose salary is **above or equal** to the given number. The result should be sorted by **first_name** then by **last_name** alphabetically.

Example

Supplied number for that example is 48100.

first_name	last_name
Terri	Duffy
Jean	Trenary
Ken	Sanchez
…	…

2) Write a stored procedure **usp_get_towns_starting_with** that **accept string as parameter** and returns **all town names starting with that string.** The result should be sorted by **town name** alphabetically.

Example

Here is the list of all towns **starting with "b".**

town
Bellevue
Bothell
Bordeaux
Berlin

3) Write a function **ufn_get_salary_level** that receives **salary of an employee** and returns the **level of the salary**.

- If salary is < **30000** return **"Low"**
- If salary is **between 30000 and 50000 (inclusive)** return **"Average"**
- If salary is > **50000** return **"High"**

4) Create a function **ufn_calculate_future_value** that accepts as **parameters – sum, yearly interest rate and number of years**. It should calculate and return the future value of the initial sum. Using the following formula:

$FV = I \times ((1 + R)^T)$

- **I – Initial sum**
- **R – Yearly interest rate**
- **T – Number of years**

  Example:

Input	Output
Initial sum: 1000 **Yearly Interest rate:** 10% **years:** 5 ufn_calculate_future_value(1000, 0.1, 5)	1610.51

5) Create a cursor to find the list of all employees in a department passed as an argument from the employee table.

6) Create a cursor to increment the salary based on designation:
if designation = manager
20000
if designation = trainer
30000