Input *(Data)*: ASDE Assignment

Output *(Report)*

Fetch top 3 departments along with their name and average monthly salary. Below is the format of the report.

DEPT_NAME	AVG_MONTHLY_SALARY (USD)

Task-1 SQL

In the attachment above, use each worksheet as a table in a relational database and write an SQL query that generates the output report

Task-2 Scripting

With the same attachment, use each worksheet as a CSV file and write a script (Bash or Python) that generates the same report. Data is to be read from the CSV files not from a database.

Task-3 Debugging

Given below is a Bash / Python script that performs following computation on an integer input (n):

1. If n is less than 10: Calculate its Square
    a. Example: 4 => 16
2. If n is between 10 and 20: Calculate the factorial of (n-10)
    a. Example: 15 => 120
3. If n is greater than 20: Calculate the sum of all integers between 1 and (n-20)
    a. Example: 25 => 15

The task is to identify the bugs in the script, fix them and share the new script. Only one of the two scripts required Bash or Python. **Hint**: You can correct the script by only changing less than 5 characters.

Script (Bash)

```bash
#!/bin/bash
N=$1
if [ $N -lt 10 ]
then
        OUT=$((N*N))
elif [ $N -lt 20 ]
then
        OUT=1
        LIM=$((N - 10))
        for (( i=1; i<$LIM; i++ ))
        do
                OUT=$((OUT * i))
        done
else
        LIM=$((N - 20))
        OUT=$((LIM * LIM))
        OUT=$((OUT - LIM))
        OUT=$((OUT / 2))
fi
echo $OUT
```

Script (Python)

```python
def compute(n):
    if n < 10:
        out = n ** 2
    elif n < 20:
        out = 1
        for i in range(1, n-10):
            out *= i
    else:
        lim = n - 20
        out = lim * lim
        out = out - lim
        out = out / 2
    print(out)
```


```
n = int(input("Enter an integer: "))
compute(n)
```

Coding Guidelines

1. Code (Script / SQL) has to be executable and free of any errors.
2. Identify and mention the test cases that need to be covered.
3. Mention any assumptions clearly.
4. Add brief comments to describe the logic/code.
5. Follow best coding practices and structure your code to be modular and readable.
6. Add instructions on how to run the code.
7. For the debugging problem, mention all the edge cases you have tested for and the corresponding fixes.
8. All the 3 tasks are mandatory.
9. The assignment will have to be submitted within 24 hours.
10. This document is read only, create your own files /documents for submissions.