6 Jun 2023

Question 1 is mandatory.
Out of other questions you have to answer 2, so choose wisely.
All problems other than [1] will require you to code, so code well.
We expect you to provide gists in github or a github project - which compiles and runs.

1. Mandatory : Elaborate how you tested your internship or academic projects.
    a. What did the system used to do?
    b. What other systems have you seen in the wild like that?
    c. How do you approach the testing problem?
    d. What were interesting bugs?
    e. How did you fix them?

2. There is one database. Let's say it is hosted locally and one of the team members migrates it to AWS or GCP. How can one confirm that the copied data is the same as the original data ? What would be the check points ?
   Imagine that data from table is of the form : List<Map<String,String>>

3. Write a program to add two integers to be taken as string and which returns a string and come up with the test cases to verify the output
   function string addNumbers (string val_a, string val_b)

4. There is a function that takes a list of integers and is supposed to return the sorted version of the input list. Write test cases for the function.
    a. Note : It is NOT trivial. Think very carefully.
    b. Bonus points: automate the test cases by writing generic unit testing, how?

5. There is an API that one must call to get data. The trouble is it will not let you cross the limit of call - say 15 calls per minute. If you cross the limit, the system penalises you by one additional minute of penalty where you can not make any call. Here is how the API looks like: function string call_me(string input).

   Propose a solution by which:
    1. You would be able to use the API within the safe limit.
    2. What happens if you are supposed to call the API 20 times per minute? Is there any way to accomplish this?
    3. If you were the API designer, what would you do to implement this behaviour?

6. Banking works by transferring money from account A to account B. Most of the time account A is in one bank while account B is another bank.

   Suppose someone writes an implementation for such a money transfer.
    1. What are the test cases?
    2. What are the issues in such a system?
    3. What can we do to mitigate some of the issues ?

4. Write the code yourself to demonstrate the mitigations.