Task 2: Aisle has millions of users from over 150 countries. Each user gets 10 free daily Likes. Unused Likes are not carried forward to the next day. We must refresh the number of daily Likes at 12:00 pm according to the user's local time. How would you go about doing this in a scalable way? No need to write code, simply explain to us in theory the backend logic in your own words.

Answer:

We can run a Scheduler(@Scheduled) which will take Cron expression, we can define Cron expression to run it at every day 12am midnight for all the time zones, which will be like this(0 0 * * *, zone). Also, we can use ShedLock(@SchedulerLock) which makes sure our scheduled tasks run only once at the same time.

I can explain it with the help of *Java Spring Boot* example:

```
Arrays.stream(new String[] {"IST", "MST", "CST", "EST"}).
      .forEach(zone -> {
        scheduler.schedule(() -> {
        }, new CronTrigger("0 0 * * * ", TimeZone.getTimeZone(zone)));
      });
```

Here in the parameter of Arrays.stream, we can pass the array of all the time zones for which we want to run this scheduler.

We can implement scheduler.schedule() function which will run a SQL query on the database, which will return the “Aisle Likes” of all the users for the given time zone, then we can update all of them once to 0 by using our given TaskRepository in the spring boot.