Soak Testing & Stress Testing

Soak testing is also called Endurance Testing
Here you test your application which is exposed to normal or moderate load for a given period of time.
You can Detect Issues like
- ◦ Memory Leaks
- ◦ Database Connections Leakage
- ◦ Performance Degradation over time
- ◦ Slow response time to time
- 100 users sends request continuously for 8 hours.

Stress Testing is to send more request to your system beyond its capacity to check how it fails or recovers.

detect issues like:
- Breaking point of the application
- how its recovering from high traffic or how it fails
- error handling happens under a pressure.

Ramp up to 5000 users in 30 seconds to see the server fails or not.

Let's Set up Locust for Stress Testing
install it via pip command: pip install locust
check the version: locust --version

once you can see the version create file: locustfile.py
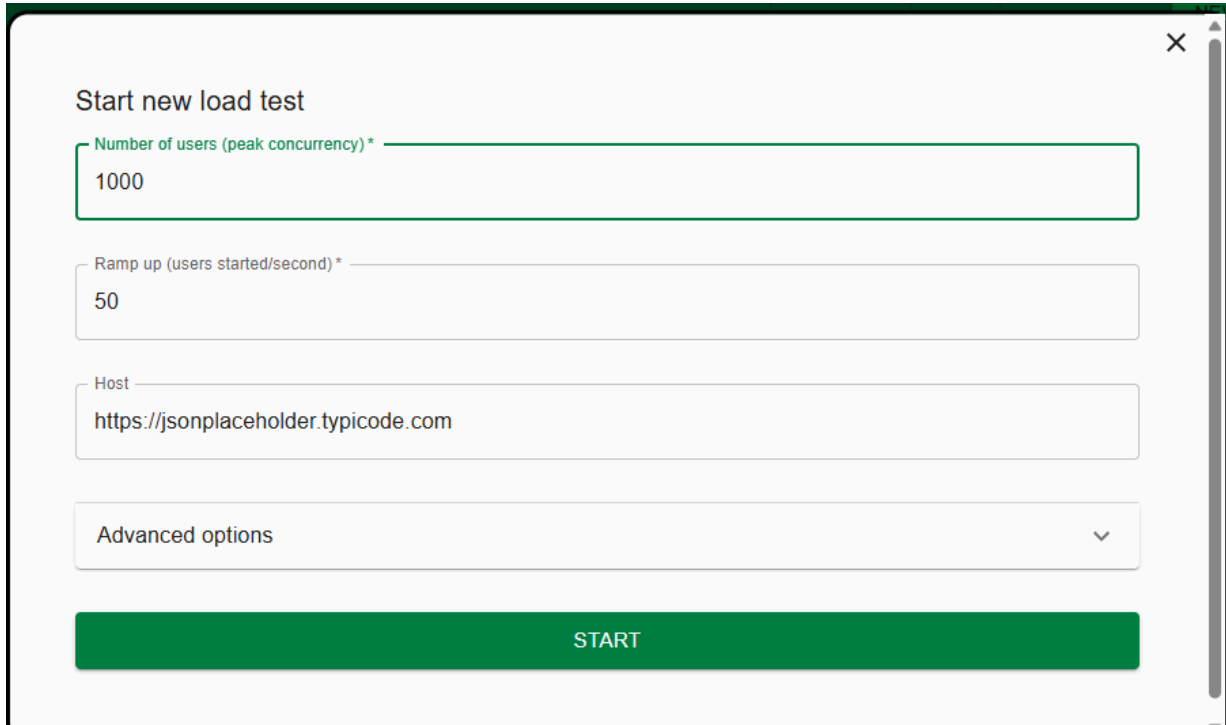
```
from locust import HttpUser,task,between

class MyStressTestUser(HttpUser):
wait_time = between(1,2)
@task
def get_posts(self):
self.client.get('/posts')
```

Save the file and let's run it.
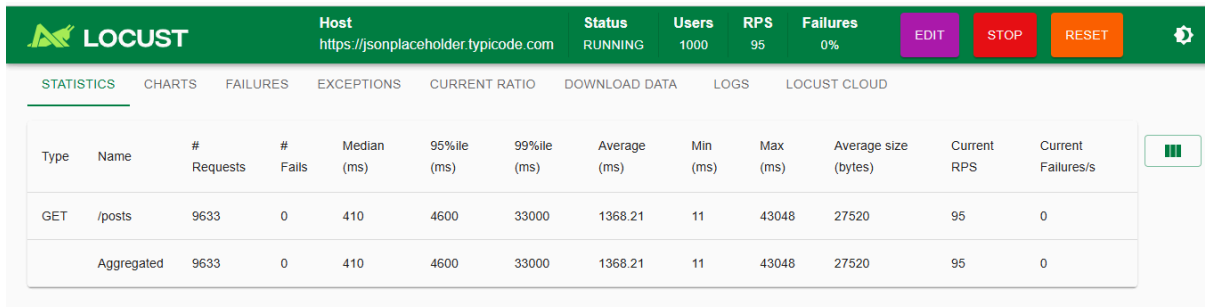
locust -f locustfile.py

```
D:\PhysicsWalla\Devops-March\stress-testing-locust>locust -f locustfile.py
[2025-06-07 11:07:19,327] DESKTOP-4F8ELLU/INFO/locust.main: Starting Locust 2.37.9
[2025-06-07 11:07:19,330] DESKTOP-4F8ELLU/INFO/locust.main: Starting web interface at http://localhost:8089, press ent
er to open your default browser.
[2025-06-07 11:08:20,486] DESKTOP-4F8ELLU/INFO/locust.runners: Ramping to 100 users at a rate of 50.00 per second
[2025-06-07 11:08:21,660] DESKTOP-4F8ELLU/INFO/locust.runners: All users spawned: {"MyStressTestUser": 100} (100 total
 users)
```

open this url in browser: http://localhost:8089/

click on Start

check Charts, Logs, Exceptions, failures etc..