1. Tạo maven project, sau đó add thêm 2 thư viện logback, và json để parse một dữ liệu json từ url <http://news.admicro.vn:10002/api/realtime?domain=kenh14.vn>. Lấy dữ liệu mỗi 2s một lần, nếu số user trong field user ở json trả ra mà lớn hơn 1% so với số user lần cuối cùng ghi vào file log thì sử dụng logback ghi số user vào file log với level là INFO, nếu không thì sau 12s ghi lại số user đó vào file log với level DEBUG.

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0s | 2s | 4s | 6s | 8s | 10s | 12s | 14s | 16s | 18s | 20s | 22s |
| 1032 | 1022 | 1092 | 1091 | 1082 | 1072 | 1102 | 1112 | 1132 | 1032 | 1232 | 1232 |
| Info | Ko ghi | Info | Ko ghi | Ko ghi | Ko ghi | Ko ghi | Ko ghi | Debug | Ko ghi | Info | Ko ghi |

2. Tạo maven project với project encode là utf-8, sử dụng thêm thư viện jsoup, sau đó build toàn bộ project thành một file jar duy nhất chứa tất cả các thư viện đi kèm (sử dụng plug-in *jar-with-dependencies* của maven). Sử dụng thư viện jsoup lấy nội dung của url ”[http://dantri.com.vn](http://dantri.com.vn/)” và ghi nội dung ra một file với tên file là thời gian hiện tại. Tạo một máy ảo linux sử dụng Docker, đẩy file jar vừa build vào trong máy ảo đó. Sử dụng shell script của linux (file .sh) để chạy file jar mỗi 5 giây một lần.

3. Tạo một project viết một restful webservice (có thể sử dụng sparkjava.com) trả lại dãy số nguyên tố từ một đến n nhập vào từ param trên url của request (Ví dụ: <http://localhost:8080/prime?n=10000> sẽ trả lại danh sách các số nguyên tố từ 1 đến 10000).

Hãy chạy chương trình trong chế độ debug, giả sử nhập vào số n=12571, sử dụng đặt điều kiện vào breakpoint để dừng chương trình tại các số nguyên tố tìm ra mà giá trị của nó nằm trong khoảng 1.000 đến 3.000. Lưu ý: không thêm các câu lệnh rẽ nhánh vào code để thực hiện debug.Cấp phát tối đa 512mb bộ nhớ.

4. Sử dụng guava cache để giảm số lần tính toán khi muốn tìm dãy nguyên tốt cho cùng một số của bài 3, nếu sau 10s không có request hoặc sau khi ghi cache 20s thì cache sẽ tự xóa. Upload project lên github.com, đồng thời cùng lúc làm 2 việc sau để tạo confict:

* Vào project trên github, mở một file trên trình duyệt, sửa nội dung file thêm dòng /\* Chinh sua tren server github \*/, sau đó lưu lại
* Vào project trên máy, cũng vào file đó nhưng thêm dòng /\* Chinh sua tren server may client \*/, sau đó Commit và Push lên git server

Hãy merge và edit conflict 2 đoạn code trên và sửa 2 dòng trên thành dòng comment sau: /\* Chinh sua hop nhat giua server & client \*/,

5. Sử dụng ssh-key để remote vào một máy linux khác (sử dụng máy ảo linux tạo bởi Docker như bài 2) mà không sử dụng password. Deploy project bài 4 thông qua git clone và build file jar bằng lệnh của maven trên máy đó.