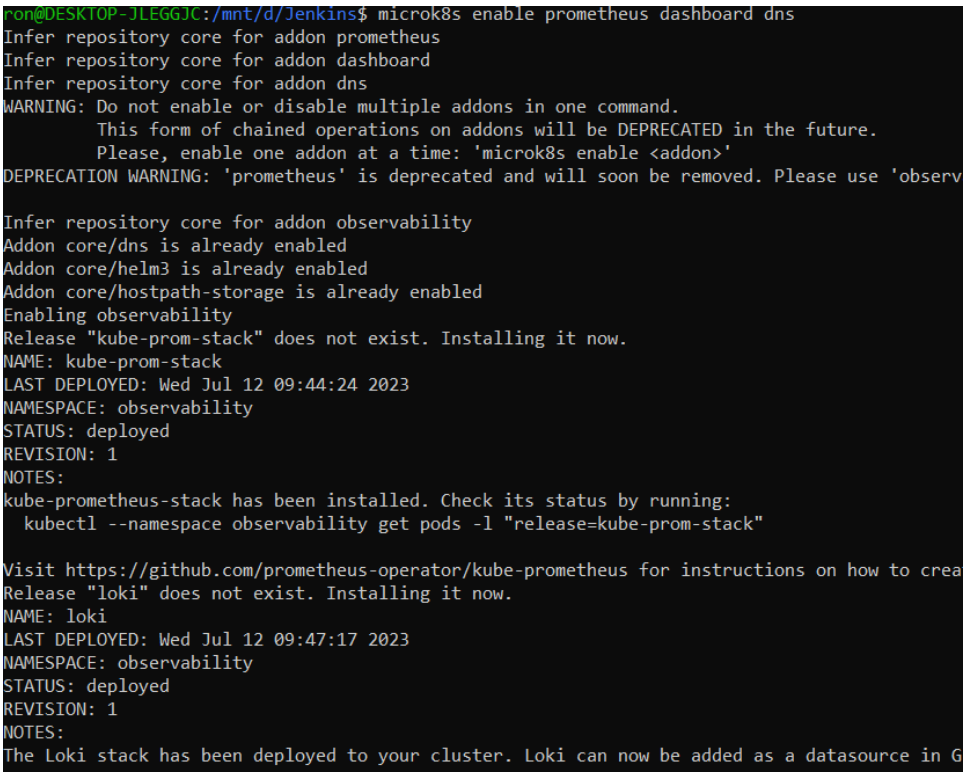
1. Enable Prometheus or Observability, Dashboard, and DNS



Full Output:

ron@DESKTOP-JLEGGJC:/mnt/d/Jenkins$ microk8s enable prometheus dashboard dns
Infer repository core for addon prometheus
Infer repository core for addon dashboard
Infer repository core for addon dns
WARNING: Do not enable or disable multiple addons in one command.
This form of chained operations on addons will be DEPRECATED in the future.
Please, enable one addon at a time: 'microk8s enable <addon>'
DEPRECATION WARNING: 'prometheus' is deprecated and will soon be removed. Please use 'observability' instead.

Infer repository core for addon observability
Addon core/dns is already enabled
Addon core/helm3 is already enabled
Addon core/hostpath-storage is already enabled
Enabling observability
Release "kube-prom-stack" does not exist. Installing it now.
NAME: kube-prom-stack
LAST DEPLOYED: Wed Jul 12 09:44:24 2023
NAMESPACE: observability
STATUS: deployed
REVISION: 1
NOTES:

kube-prometheus-stack has been installed. Check its status by running:
kubectl --namespace observability get pods -l "release=kube-prom-stack"

Visit https://github.com/prometheus-operator/kube-prometheus for instructions on how to create & configure Alertmanager and Prometheus instances using the Operator.
Release "loki" does not exist. Installing it now.
NAME: loki
LAST DEPLOYED: Wed Jul 12 09:47:17 2023
NAMESPACE: observability
STATUS: deployed
REVISION: 1
NOTES:
The Loki stack has been deployed to your cluster. Loki can now be added as a datasource in Grafana.

See http://docs.grafana.org/features/datasources/loki/ for more detail.
Release "tempo" does not exist. Installing it now.
NAME: tempo
LAST DEPLOYED: Wed Jul 12 09:47:53 2023
NAMESPACE: observability
STATUS: deployed
REVISION: 1
TEST SUITE: None
[sudo] password for ron:

Note: the observability stack is setup to monitor only the current nodes of the MicroK8s cluster. For any nodes joining the cluster at a later stage this addon will need to be set up again.

Observability has been enabled (user/pass: admin/prom-operator)
Addon core/dashboard is already enabled
Addon core/dns is already enabled
ron@DESKTOP-JLEGGJC:/mnt/d/Jenkins$

2. Check or monitor if Prometheus jobs are enabled

   microk8s kubectl get services -n observability

```
ron@DESKTOP-JLEGGJC:/mnt/d/Jenkins$ microk8s kubectl get services -n observability
NAME                                     TYPE        CLUSTER-IP       EXTERNAL-IP   PORT(S)
 AGE
kube-prom-stack-prometheus-node-exporter   ClusterIP   10.152.183.76    <none>        9100/TCP
 49m
kube-prom-stack-kube-prome-alertmanager    ClusterIP   10.152.183.161   <none>        9093/TCP
 49m
kube-prom-stack-grafana                    ClusterIP   10.152.183.96    <none>        80/TCP
 49m
kube-prom-stack-kube-prome-operator        ClusterIP   10.152.183.210   <none>        443/TCP
 49m
kube-prom-stack-kube-prome-prometheus      ClusterIP   10.152.183.212   <none>        9090/TCP
 49m
kube-prom-stack-kube-state-metrics         ClusterIP   10.152.183.43    <none>        8080/TCP
 49m
alertmanager-operated                      ClusterIP   None             <none>        9093/TCP,9094/TCP,9094/UDP
 48m
prometheus-operated                        ClusterIP   None             <none>        9090/TCP
 48m
loki-headless                              ClusterIP   None             <none>        3100/TCP
 48m
loki-memberlist                            ClusterIP   None             <none>        7946/TCP
 48m
loki                                       ClusterIP   10.152.183.203   <none>        3100/TCP
 48m
tempo                                      ClusterIP   10.152.183.189   <none>        3100/TCP,16687/TCP,16686/TCP
 47m
```

microk8s kubectl get pods -n observability

```
ron@DESKTOP-JLEGGJC:/mnt/d/Jenkins$ microk8s kubectl get pods -n observability
NAME                                                READY   STATUS                 RESTARTS      AGE
kube-prom-stack-kube-prome-operator-64ffd55b77-mg285   1/1     Running                0             50m
tempo-0                                             2/2     Running                0             48m
kube-prom-stack-kube-state-metrics-6c586bf4c8-864tn   1/1     Running                0             50m
alertmanager-kube-prom-stack-kube-prome-alertmanager-0   2/2     Running                1 (47m ago)   49m
loki-0                                              1/1     Running                0             49m
loki-promtail-d82wl                                 1/1     Running                0             49m
prometheus-kube-prom-stack-kube-prome-prometheus-0   2/2     Running                0             49m
kube-prom-stack-grafana-6c47f548d6-hb4wb            3/3     Running                3 (37m ago)   50m
kube-prom-stack-prometheus-node-exporter-wg46h      0/1     CreateContainerError   0             50m
ron@DESKTOP-JLEGGJC:/mnt/d/Jenkins$
```

Error was failed to generate spec: path "/" is mounted on "/" but it is not a shared or slave mount container error node exporter

Then execute the command below to correct the error

```
mount --make-rshared /
```

Then again check all, notice that all objects are successfully ran

```
root@DESKTOP-JLEGGJC:~# kubectl get all -n observability
NAME                                                     READY   STATUS    RESTARTS      AGE
pod/kube-prom-stack-kube-state-metrics-77d7bc94d5-p9xjg  1/1     Running   0             24m
pod/kube-prom-stack-kube-prome-operator-57f9bcd99b-8w741 1/1     Running   0             24m
pod/alertmanager-kube-prom-stack-kube-prome-alertmanager-0   2/2     Running   0             23m
pod/prometheus-kube-prom-stack-kube-prome-prometheus-0   2/2     Running   0             23m
pod/loki-promtail-6xbcl                                  1/1     Running   0             21m
pod/loki-0                                               1/1     Running   0             21m
pod/kube-prom-stack-grafana-58cb7d8f66-xwbrm             3/3     Running   2 (18m ago)   24m
pod/kube-prom-stack-prometheus-node-exporter-mzqvn       1/1     Running   0             24m

NAME                                              TYPE        CLUSTER-IP       EXTERNAL-IP   PORT(S)                      AGE
service/kube-prom-stack-prometheus-node-exporter  ClusterIP   10.152.183.198   <none>        9100/TCP                     25m
service/kube-prom-stack-kube-state-metrics        ClusterIP   10.152.183.120   <none>        8080/TCP                     25m
service/kube-prom-stack-grafana                   ClusterIP   10.152.183.200   <none>        80/TCP                       25m
service/kube-prom-stack-kube-prome-operator       ClusterIP   10.152.183.246   <none>        443/TCP                      25m
service/kube-prom-stack-kube-prome-prometheus     ClusterIP   10.152.183.52    <none>        9090/TCP,8080/TCP            25m
service/kube-prom-stack-kube-prome-alertmanager   ClusterIP   10.152.183.117   <none>        9093/TCP,8080/TCP            25m
service/alertmanager-operated                     ClusterIP   None             <none>        9093/TCP,9094/TCP,9094/UDP   24m
service/prometheus-operated                       ClusterIP   None             <none>        9090/TCP                     24m
service/loki-headless                             ClusterIP   None             <none>        3100/TCP                     21m
service/loki-memberlist                           ClusterIP   None             <none>        7946/TCP                     21m
service/loki                                      ClusterIP   10.152.183.109   <none>        3100/TCP                     21m

NAME                                                 DESIRED   CURRENT   READY   UP-TO-DATE   AVAILABLE   NODE SELECTOR
daemonset.apps/loki-promtail                         1         1         1       1            1           <none>
daemonset.apps/kube-prom-stack-prometheus-node-exporter   1         1         1       1            1           kubernetes.io/os=l

NAME                                               READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/kube-prom-stack-kube-state-metrics 1/1     1            1           24m
deployment.apps/kube-prom-stack-kube-prome-operator   1/1     1            1           24m
deployment.apps/kube-prom-stack-grafana            1/1     1            1           24m

NAME                                                     DESIRED   CURRENT   READY   AGE
replicaset.apps/kube-prom-stack-kube-state-metrics-77d7bc94d5    1         1         1       24m
replicaset.apps/kube-prom-stack-kube-prome-operator-57f9bcd99b   1         1         1       24m
replicaset.apps/kube-prom-stack-grafana-58cb7d8f66               1         1         1       24m

NAME                                                         READY   AGE
statefulset.apps/alertmanager-kube-prom-stack-kube-prome-alertmanager   1/1     24m
statefulset.apps/prometheus-kube-prom-stack-kube-prome-prometheus       1/1     23m
statefulset.apps/loki                                        1/1     21m
root@DESKTOP-JLEGGJC:~#
```

3. Execute below command for port forwarding prometheus
   microk8s kubectl port-forward -n observability service/kube-prom-stack-kube-prome-prometheus --address 0.0.0.0 9090:9090

   sample:

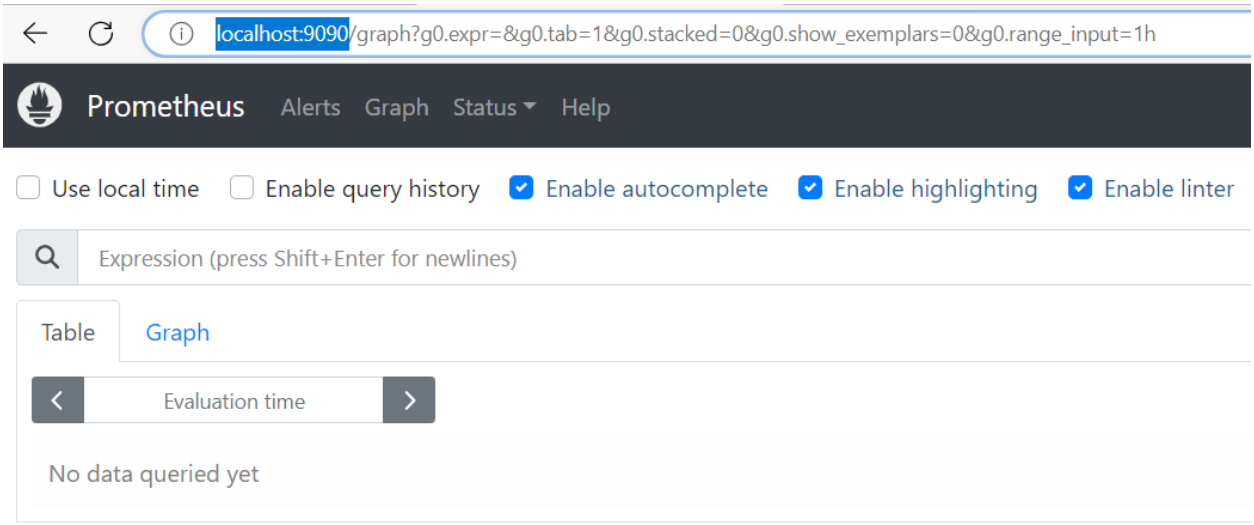
```
root@DESKTOP-JLEGGJC:~# microk8s kubectl port-forward -n observability service
Forwarding from 0.0.0.0:9090 -> 9090
```

4. After executing port forward for Prometheus, it will continue executing until Ctrl + C but we don't have intention of exiting and therefore open a new command prompt window and execute below command for port forwarding Grafana
   microk8s kubectl port-forward -n observability service/kube-prom-stack-grafana --address 0.0.0.0 3000:80

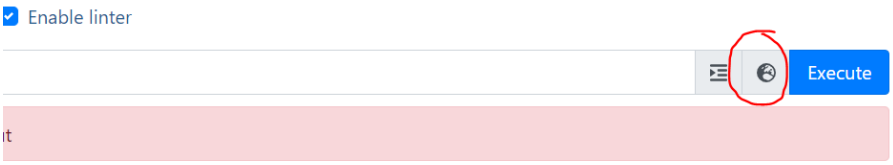
   Sample:

```
root@DESKTOP-JLEGGJC:~# microk8s kubectl port-forward -n observability se
Forwarding from 0.0.0.0:3000 -> 3000
```

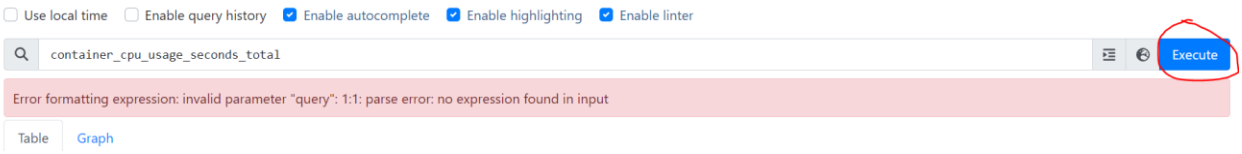
5. Open link http://localhost:9090/ in browser. It will go to Prometheus page
   Sample:



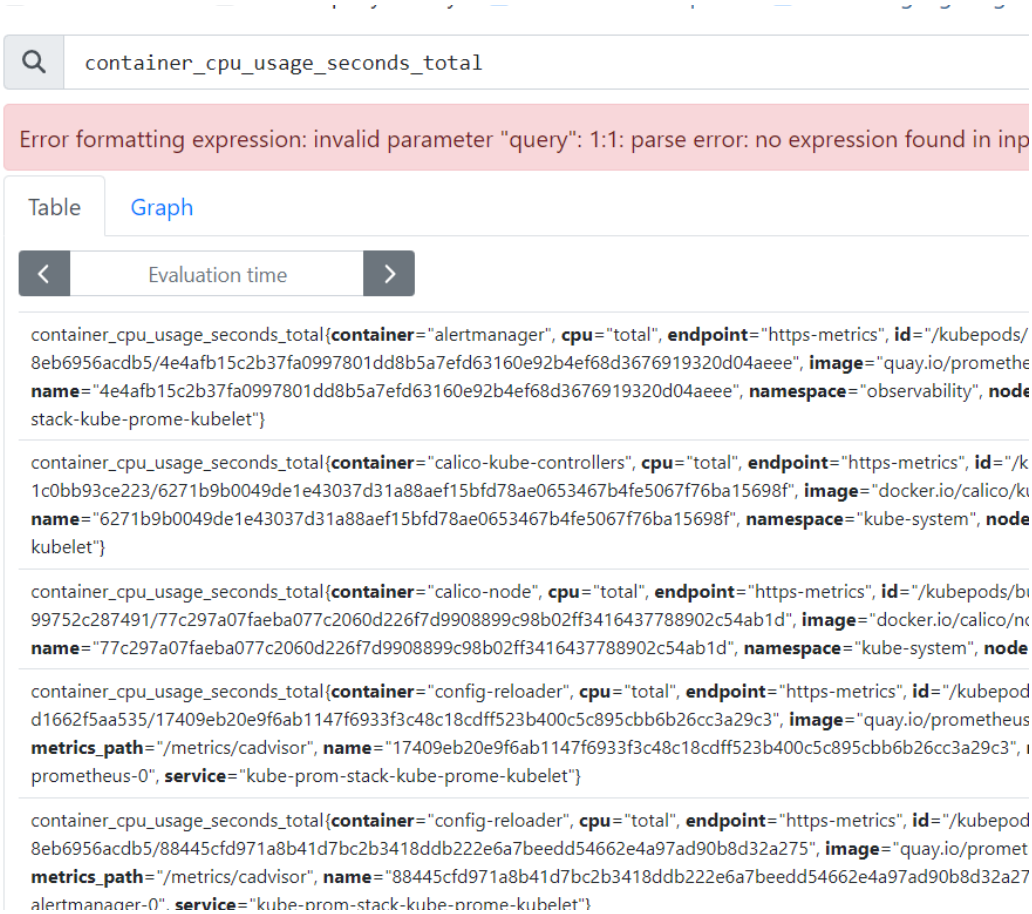
Then click  beside Blue Execute button to check available expressions and choose an expression



After choosing, click Execute button

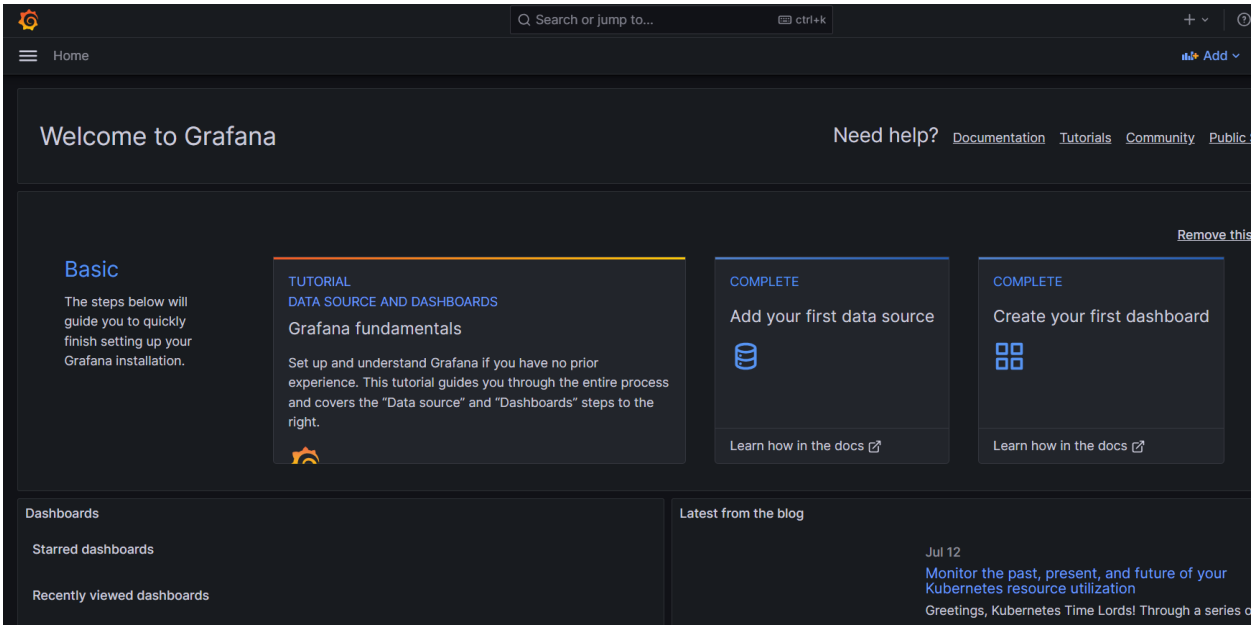


Since we are in Table tab then we can see data

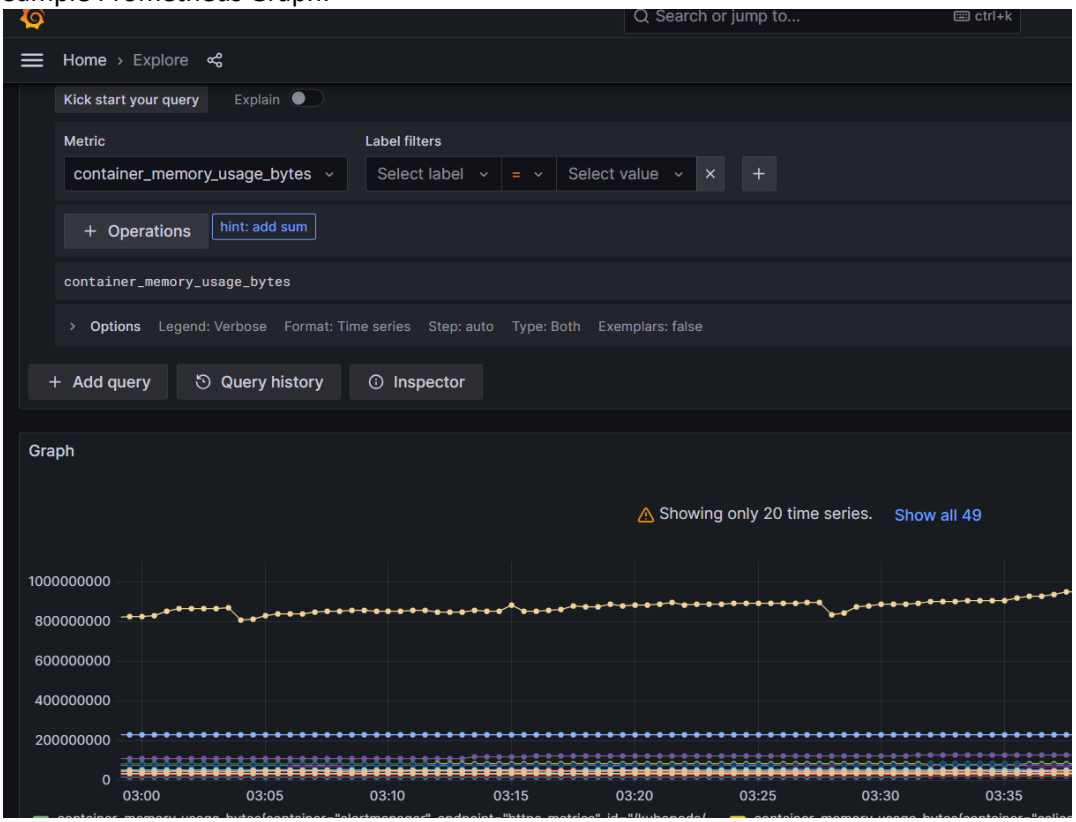


Please click Graph tab to make it in Graph form

6. Open link http://localhost:3000/ in browser. It will go to Grafana page. Please use the credentials from step 1. You will go to Grafana homepage



Sample Prometheus Graph:



7. Then initialize Selenium Grid via Kubernetes with namespace selenium so that we can properly check the graph in Grafana connected to Prometheus data source

Selenium Grid Kubernetes initializations:
microk8s kubectl create namespace selenium
microk8s kubectl create deployment selenium-hub --image selenium/hub:3.9.0 --port 4444 --namespace selenium
microk8s kubectl expose deployment selenium-hub --type=ClusterIP --namespace selenium
microk8s kubectl create deployment selenium-node-chrome --image selenium/node-chrome:3.9.1 --namespace selenium
microk8s kubectl expose deployment selenium-node-chrome --type=ClusterIP --port 4444 --namespace selenium
microk8s kubectl set env deployment/selenium-node-chrome --env="HUB_PORT_4444_TCP_ADDR=selenium-hub" --env="HUB_PORT_4444_TCP_PORT=4444" --namespace selenium
microk8s kubectl create deployment selenium-node-firefox --image selenium/node-firefox:3.9.1 --namespace selenium
microk8s kubectl expose deployment selenium-node-firefox --type=ClusterIP --port 4444 --namespace selenium
microk8s kubectl set env deployment/selenium-node-firefox --env="HUB_PORT_4444_TCP_ADDR=selenium-hub" --env="HUB_PORT_4444_TCP_PORT=4444" --namespace selenium
microk8s kubectl scale deployment --replicas 2 selenium-node-chrome --namespace selenium
microk8s kubectl scale deployment --replicas 2 selenium-node-firefox --namespace selenium

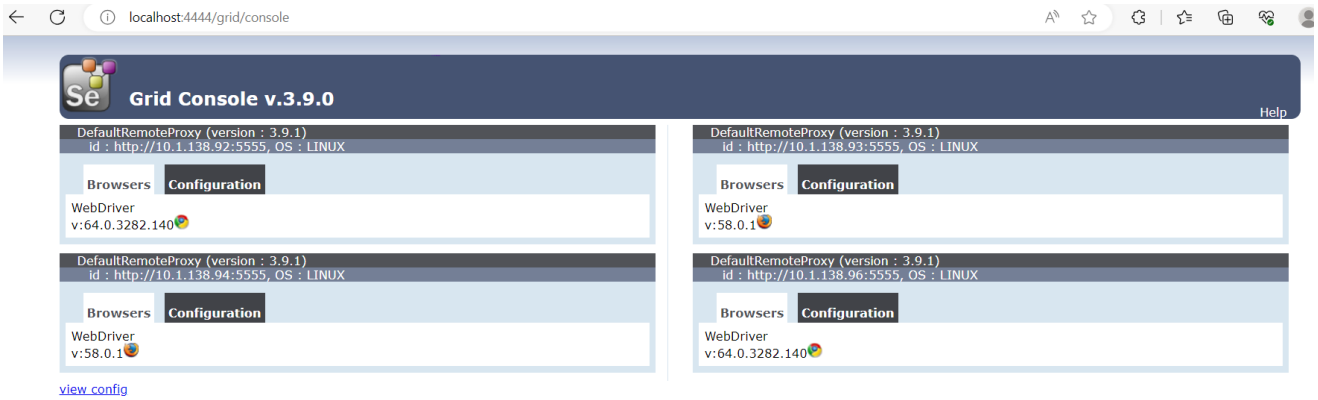
and then port forward 4444 to see if Selenium Hub contains 2 nodes for Chrome and 2 nodes for Firefox
microk8s kubectl port-forward -n selenium service/selenium-hub --address 0.0.0.0 4444:4444

Sample:

```
root@DESKTOP-JLEGGJC:~# microk8s kubectl port-forward -n selenium service/selenium-hub --address 0.0.0.0 4444:4444
Forwarding from 0.0.0.0:4444 -> 4444
```

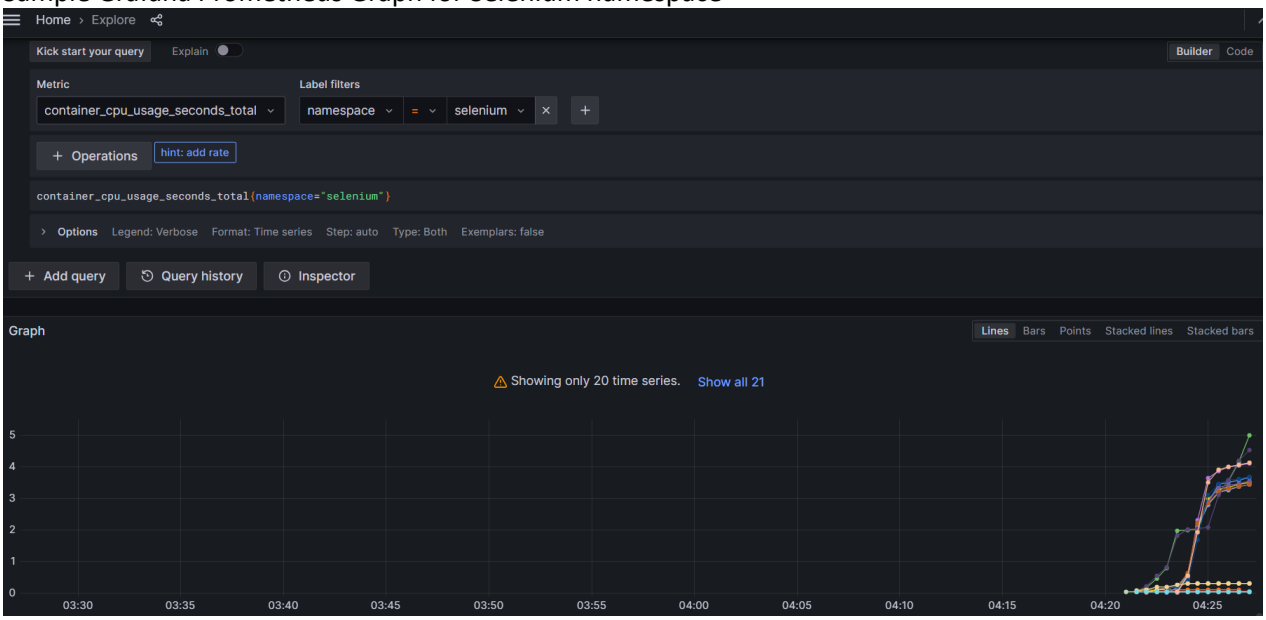
8. Check that the grid console is working and has 2 nodes chrome and 2 nodes firefox

   Link: http://localhost:4444/grid/console



9. Check Grafana if can monitor selenium namespace

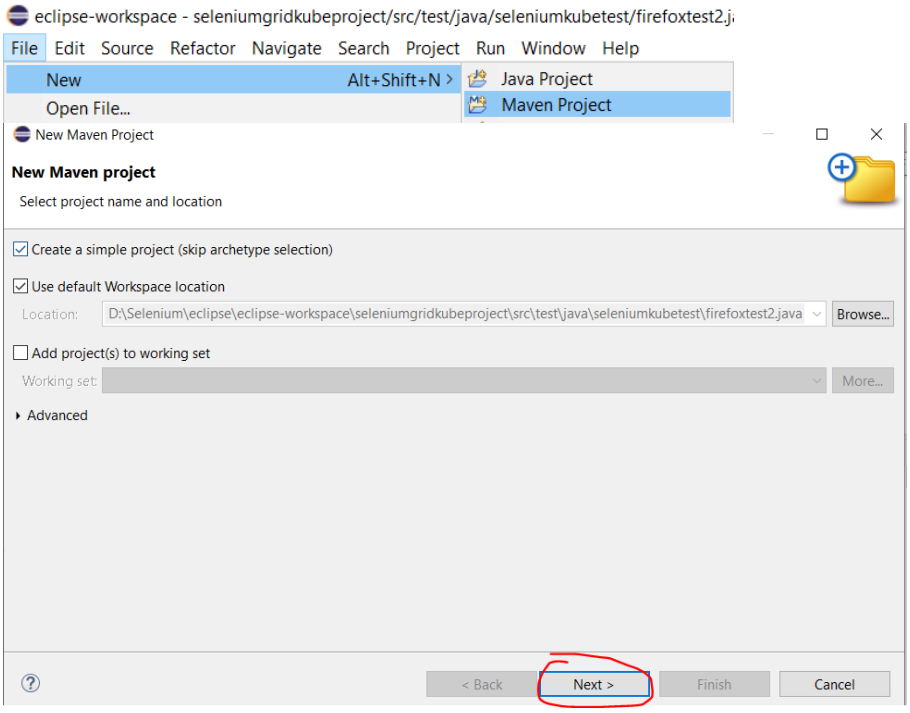
Sample Grafana Prometheus Graph for Selenium namespace

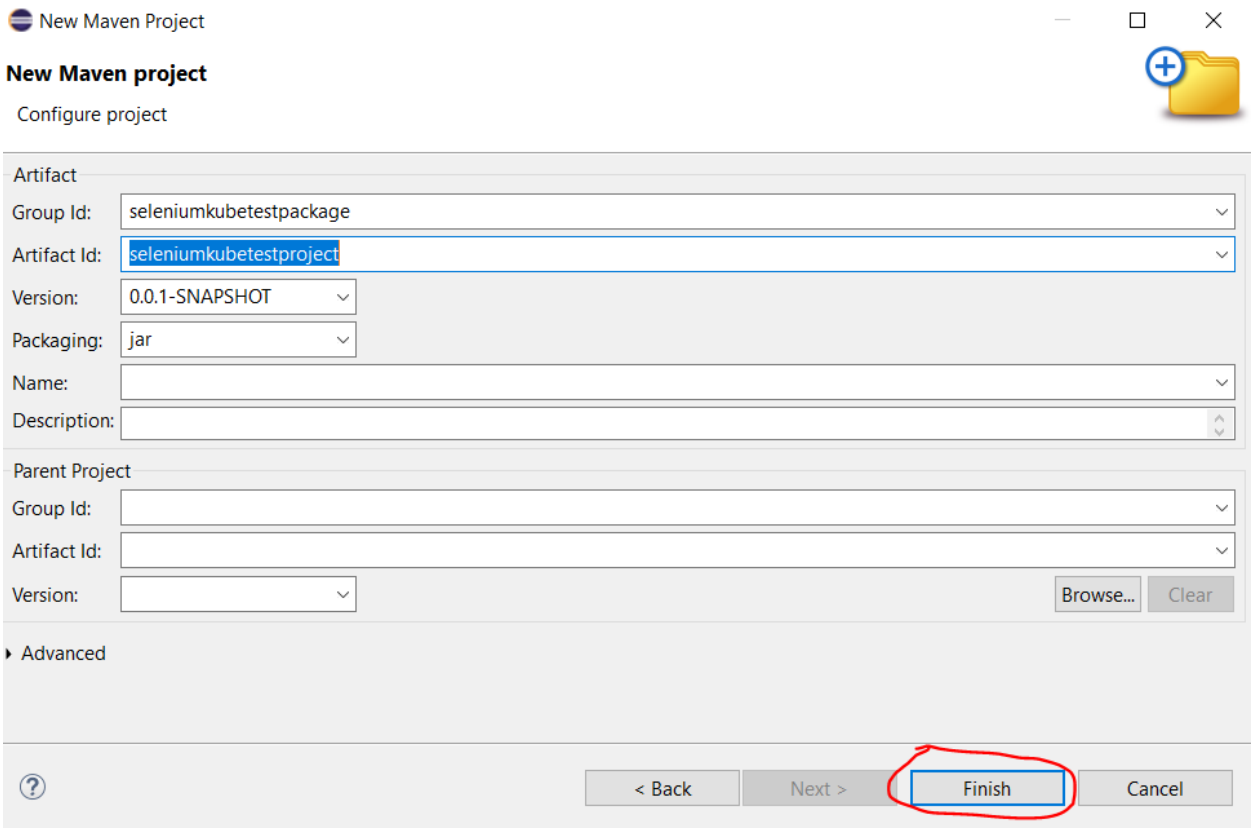


10. Then build a selenium project to test the selenium grids

    Need to Specify a new Workspace folder.

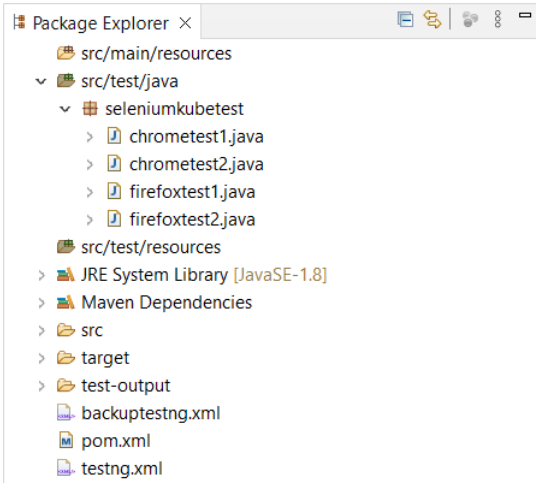
    Create a Maven Project





Note: Please see structure and codes below. Specify your own Project

Structure:



Codes:

Pom.xml (for dependencies)

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
https://maven.apache.org/xsd/maven-4.0.0.xsd">
```

```
  <modelVersion>4.0.0</modelVersion>
  <groupId>seleniummavengridpackage</groupId>
  <artifactId>seleniummavengridproject</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <dependencies>
       <dependency>
             <groupId>org.seleniumhq.selenium</groupId>
             <artifactId>selenium-java</artifactId>
             <version>3.8.0</version>
       </dependency>
       <dependency>
             <groupId>org.testng</groupId>
             <artifactId>testng</artifactId>
             <version>6.5.1</version>
       </dependency>
       <dependency>
        <groupId>org.slf4j</groupId>
        <artifactId>slf4j-api</artifactId>
        <version>1.7.5</version>
       </dependency>
       <dependency>
        <groupId>org.slf4j</groupId>
        <artifactId>slf4j-log4j12</artifactId>
        <version>1.7.5</version>
       </dependency>
  </dependencies>
  <build>
       <plugins>
             <plugin>
                   <groupId>org.apache.maven.plugins</groupId>
             <artifactId>maven-compiler-plugin</artifactId>
             <version>3.9.0</version>
             <configuration>
                   <source>1.8</source>
                   <target>1.8</target>
             </configuration>
             </plugin>

             <plugin>
                   <groupId>org.apache.maven.plugins</groupId>
             <artifactId>maven-surefire-plugin</artifactId>
             <version>2.20</version>
             <configuration>
                   <suiteXmlFiles>
                         <suiteXmlFile>testng.xml</suiteXmlFile>
                         <!--
<suiteXmlFile>src/main/resources/testng.xml</suiteXmlFile> -->
                   </suiteXmlFiles>
             </configuration>
             </plugin>
       </plugins>
  </build>
</project>
```

Testng.xml (Note: Parallelization may not work when executing pom.xml test in Jenkins)

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE suite SYSTEM "https://testng.org/testng-1.0.dtd">
<suite parallel="classes" name="Suite">
  <test thread-count="4" parallel="classes" name="Test">
    <classes>
      <class name="seleniumkubetest.chrometest1"/>
      <class name="seleniumkubetest.chrometest2"/>
      <class name="seleniumkubetest.firefoxtest1"/>
      <class name="seleniumkubetest.firefoxtest2"/>
    </classes>
  </test> <!-- Test -->
</suite> <!-- Suite -->
```

chrometest1.java

```
package seleniumkubetest;

import org.testng.annotations.Test;
import java.net.MalformedURLException;
import java.net.URL;
import java.util.concurrent.TimeUnit;

import org.openqa.selenium.UnexpectedAlertBehaviour;
import org.openqa.selenium.chrome.ChromeOptions;
import org.openqa.selenium.remote.CapabilityType;
import org.openqa.selenium.remote.RemoteWebDriver;

public class chrometest1 {


	@Test
	public void test1() throws MalformedURLException
	{
		ChromeOptions cap = new ChromeOptions();
		cap.setCapability(CapabilityType.UNEXPECTED_ALERT_BEHAVIOUR,
				UnexpectedAlertBehaviour.IGNORE);

		RemoteWebDriver driver = new RemoteWebDriver(new
URL("http://localhost:4444/wd/hub"),cap);
		driver.manage().timeouts().implicitlyWait(180, TimeUnit.SECONDS);
		driver.get("http://yahoo.com");
		System.out.println("Chrome Title = " + driver.getTitle());

		driver.quit();
	}
}
```

chrometest2.java

```
package seleniumkubetest;
```

```
import org.testng.annotations.Test;
import java.net.MalformedURLException;
import java.net.URL;
import java.util.concurrent.TimeUnit;

import org.openqa.selenium.UnexpectedAlertBehaviour;
import org.openqa.selenium.chrome.ChromeOptions;
import org.openqa.selenium.remote.CapabilityType;
import org.openqa.selenium.remote.RemoteWebDriver;

public class chrometest2 {


	@Test
	public void test2() throws MalformedURLException
	{
		ChromeOptions cap = new ChromeOptions();
		cap.setCapability(CapabilityType.UNEXPECTED_ALERT_BEHAVIOUR,
				UnexpectedAlertBehaviour.IGNORE);

		RemoteWebDriver driver = new RemoteWebDriver(new
URL("http://localhost:4444/wd/hub"),cap);
		driver.manage().timeouts().implicitlyWait(180, TimeUnit.SECONDS);
		driver.get("http://google.com");
		System.out.println("Chrome Title = " + driver.getTitle());

		driver.quit();
	}
}
```

firefoxtest1.java

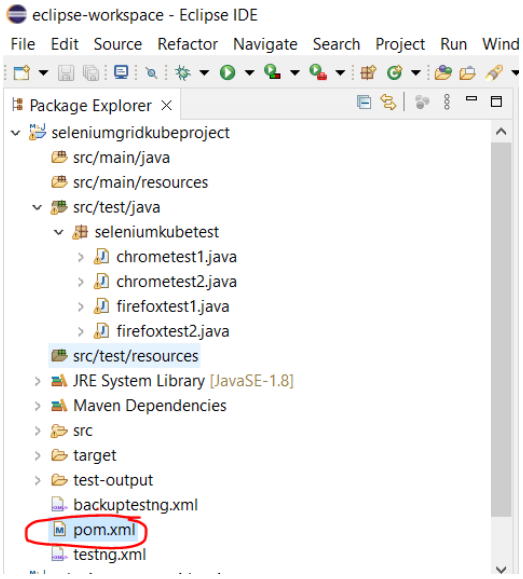
```
package seleniumkubetest;

import org.testng.annotations.Test;
import java.net.MalformedURLException;
import java.net.URL;
import java.util.concurrent.TimeUnit;

import org.openqa.selenium.UnexpectedAlertBehaviour;
import org.openqa.selenium.firefox.FirefoxOptions;
import org.openqa.selenium.remote.CapabilityType;
import org.openqa.selenium.remote.RemoteWebDriver;

public class firefoxtest1 {


	@Test
```

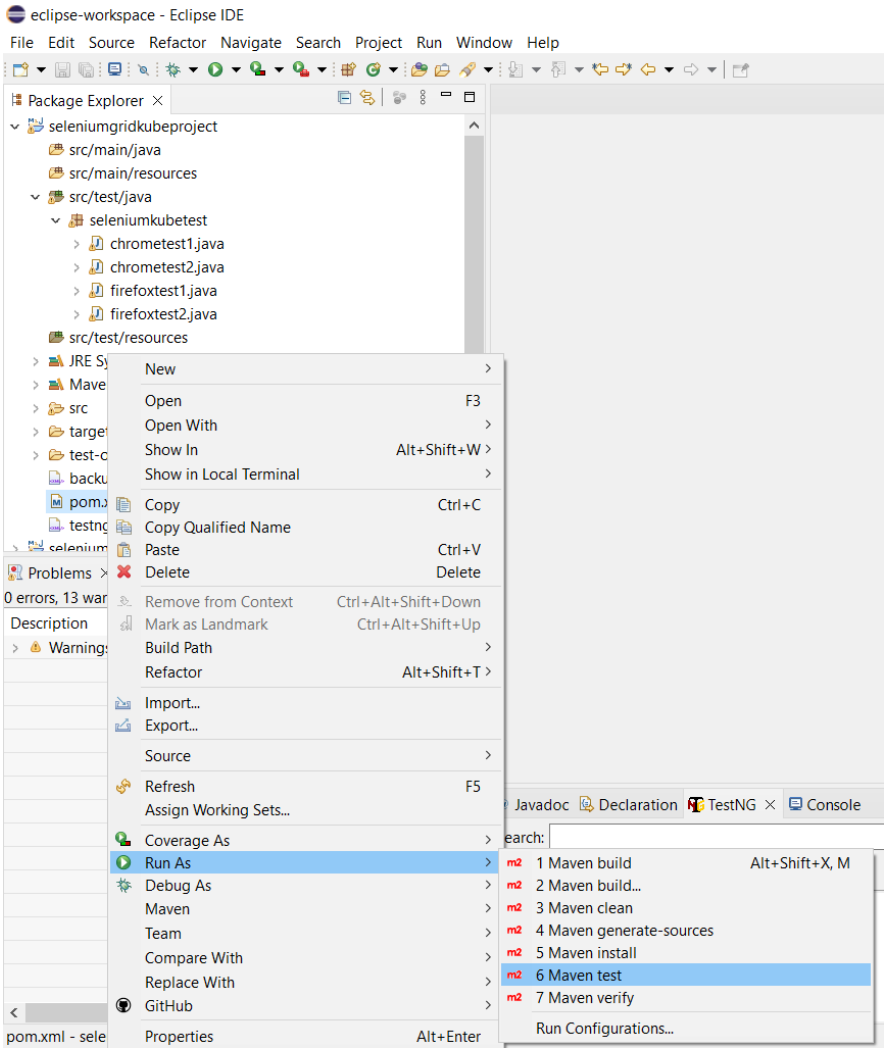
```
	public void test3() throws MalformedURLException
	{
		FirefoxOptions cap = new FirefoxOptions();
		cap.setCapability(CapabilityType.UNEXPECTED_ALERT_BEHAVIOUR,
			UnexpectedAlertBehaviour.DISMISS);

		RemoteWebDriver driver = new RemoteWebDriver(new
URL("http://localhost:4444/wd/hub"),cap);
		driver.manage().timeouts().implicitlyWait(180, TimeUnit.SECONDS);
		driver.get("http://youtube.com");
		System.out.println("Firefox Title = " + driver.getTitle());

		driver.quit();
	}
}
```

firefoxtest2.java

```
package seleniumkubetest;

import org.testng.annotations.Test;
import java.net.MalformedURLException;
import java.net.URL;
import java.util.concurrent.TimeUnit;

import org.openqa.selenium.UnexpectedAlertBehaviour;
import org.openqa.selenium.firefox.FirefoxOptions;
import org.openqa.selenium.remote.CapabilityType;
import org.openqa.selenium.remote.RemoteWebDriver;

public class firefoxtest2 {


	@Test
	public void test4() throws MalformedURLException
	{
		FirefoxOptions cap = new FirefoxOptions();
		cap.setCapability(CapabilityType.UNEXPECTED_ALERT_BEHAVIOUR,
		   UnexpectedAlertBehaviour.DISMISS);

		RemoteWebDriver driver = new RemoteWebDriver(new
URL("http://localhost:4444/wd/hub"),cap);
		driver.manage().timeouts().implicitlyWait(180, TimeUnit.SECONDS);
		driver.get("http://instagram.com");
		System.out.println("Firefox Title = " + driver.getTitle());

		driver.quit();
	}
}
```

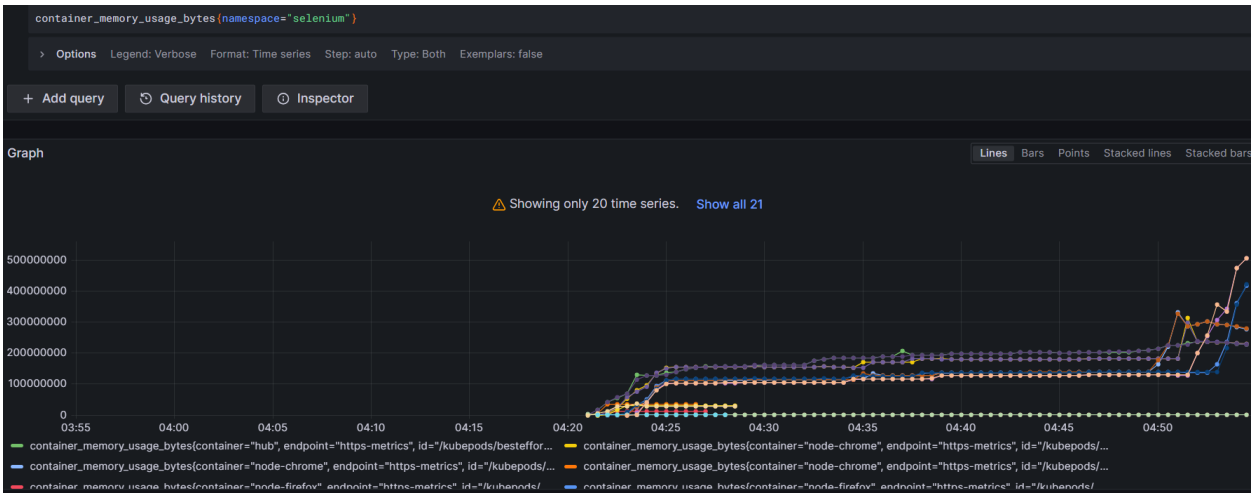
11. After creating project then test the project by right clicking pom.xml and Run as Maven test



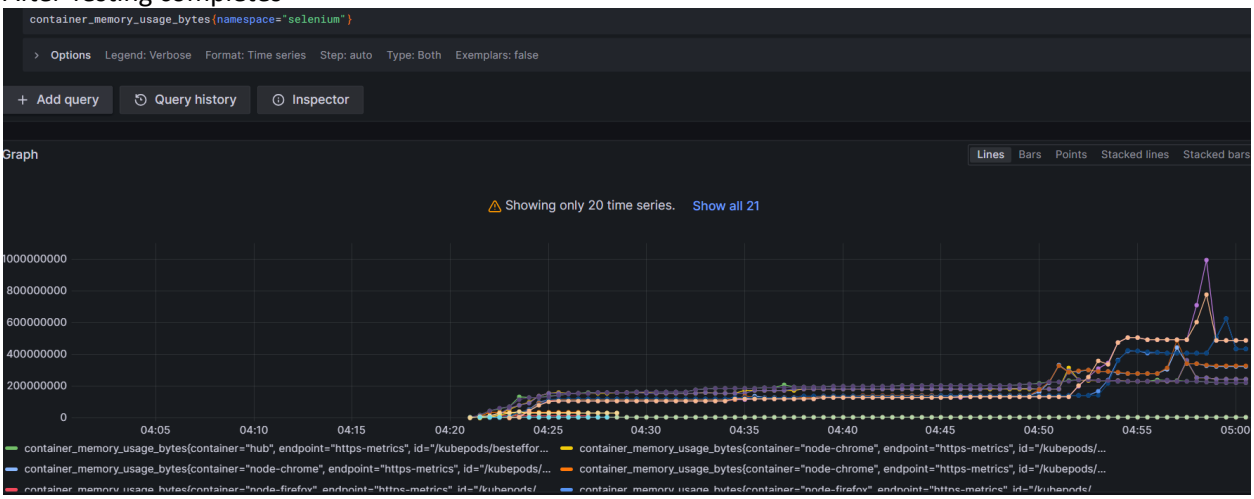


12. While testing, check the Grafana real-time monitoring (e.g. Container Memory)

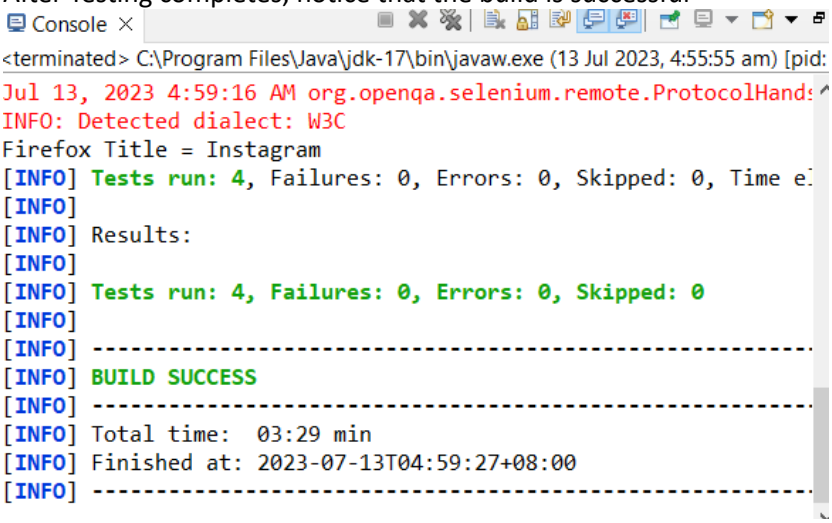
Increasing memory due to Selenium Testing in progress:



After Testing completes

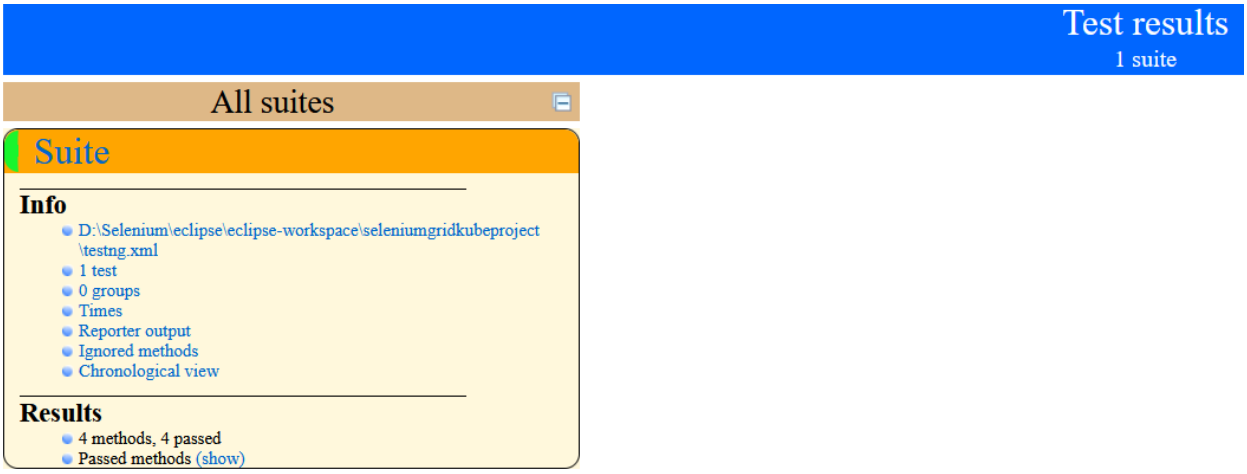


13. After Testing completes, notice that the build is successful



14. Then go to <workspace directory>/project/target/surefire-reports and open both index.html and emailable-report.html to check the test results

index.html



emailable-report.html

Test	Methods Passed	Scenarios Passed	# skipped	# failed	Total Time	Included Groups	Excluded Groups
Test	4	4	0	0	203.5 seconds		

Class	Method	# of Scenarios	Start	Time (ms)
Test — passed				
seleniumkubetest.chrometest1	test1	1	1689195362667	57371
seleniumkubetest.chrometest2	test2	1	1689195420039	25628
seleniumkubetest.firefoxtest1	test3	1	1689195445668	67057
seleniumkubetest.firefoxtest2	test4	1	1689195512726	53448

Test

seleniumkubetest.firefoxtest2:test4

---

seleniumkubetest.firefoxtest1:test3

---

seleniumkubetest.chrometest1:test1

---

seleniumkubetest.chrometest2:test2

15. Then perform cleanup by Ctrl + C all port-forwards, logout to Grafana and disable dashboard and observability and delete selenium hub and node grids for firefox and chrome and delete selenium namespace

    Selenium Cleanup scripts:
    microk8s kubectl delete service selenium-node-chrome --namespace selenium
    microk8s kubectl delete service selenium-node-firefox --namespace selenium
    microk8s kubectl delete service selenium-hub --namespace selenium
    microk8s kubectl delete deployment selenium-node-chrome --namespace selenium
    microk8s kubectl delete deployment selenium-node-firefox --namespace selenium
    microk8s kubectl delete deployment selenium-hub --namespace selenium

    Disable Dashboard, Observability, and DNS

    microk8s disable dashboard observability dns
    OR
    microk8s disable dashboard
    microk8s disable observability
    microk8s disable dns

    Sample Output for disabling Dashboard, Observability, and DNS:
    root@DESKTOP-JLEGGJC:~# microk8s disable dashboard observability dns
    Infer repository core for addon dashboard
    Infer repository core for addon observability
    Infer repository core for addon dns
    WARNING: Do not enable or disable multiple addons in one command.
        This form of chained operations on addons will be DEPRECATED in the future.
        Please, disable one addon at a time: 'microk8s disable <addon>'
    Disabling Dashboard
    serviceaccount "kubernetes-dashboard" deleted
    service "kubernetes-dashboard" deleted
    secret "kubernetes-dashboard-certs" deleted
    secret "kubernetes-dashboard-csrf" deleted
    secret "kubernetes-dashboard-key-holder" deleted
    configmap "kubernetes-dashboard-settings" deleted
    role.rbac.authorization.k8s.io "kubernetes-dashboard" deleted
    clusterrole.rbac.authorization.k8s.io "kubernetes-dashboard" deleted
    rolebinding.rbac.authorization.k8s.io "kubernetes-dashboard" deleted
    clusterrolebinding.rbac.authorization.k8s.io "kubernetes-dashboard" deleted
    deployment.apps "kubernetes-dashboard" deleted
    service "dashboard-metrics-scraper" deleted
    deployment.apps "dashboard-metrics-scraper" deleted
    Dashboard is disabled
    Disabling observability
    service "kube-prom-stack-kube-prome-kubelet" deleted
    namespace "observability" deleted

Disabling DNS
Reconfiguring kubelet
Removing DNS manifest
deployment.apps "coredns" deleted
pod/coredns-7745f9f87f-d4lk8 condition met
serviceaccount "coredns" deleted
configmap "coredns" deleted
service "kube-dns" deleted
clusterrole.rbac.authorization.k8s.io "coredns" deleted
clusterrolebinding.rbac.authorization.k8s.io "coredns" deleted
DNS is disabled

Reference: https://www.server-world.info/en/note?os=Ubuntu_22.04&p=microk8s&f=7
Reference to fix mount error because it is not shared
https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/6/html/storage_administration_guide/sect-using_the_mount_command-mounting-bind