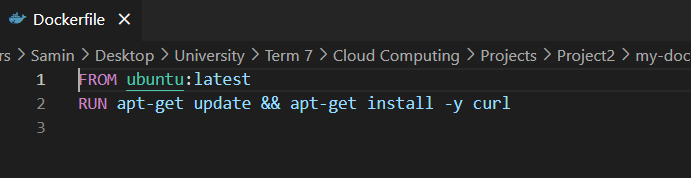
گزارش پروژه دوم رایانش ابری - داکر و مقدمات کوبرنتیز

گام اول-

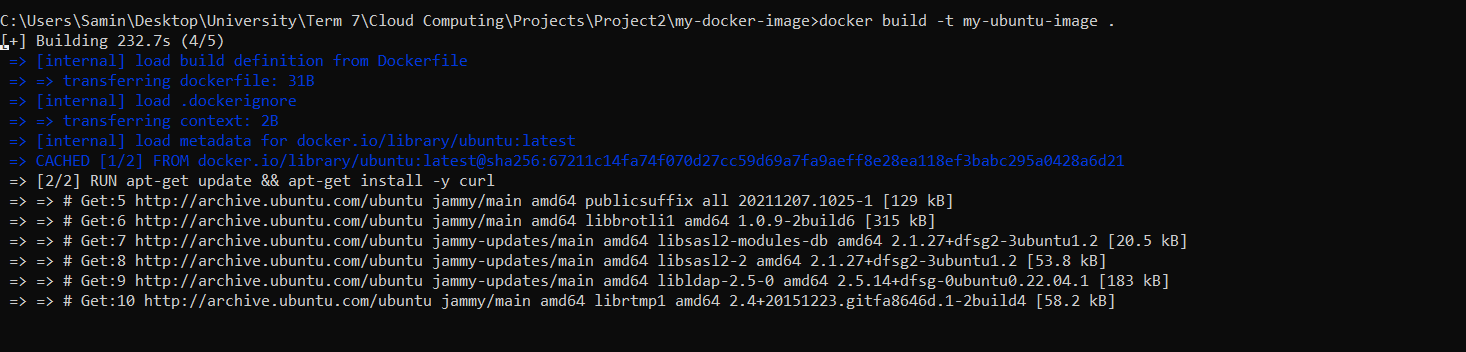
ابتدا یک دیرکتوری برای داکر فایل ساخته سپس یک داکر فایل در آن میسازیم:

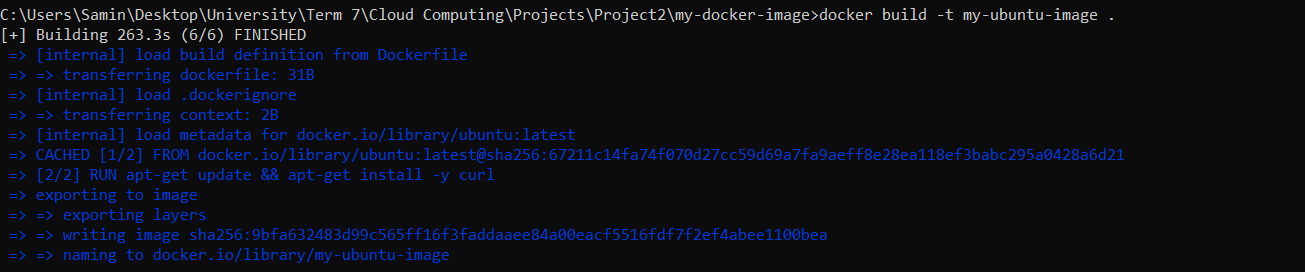


در داکر فایل عبارات زیر را مینویسم که برای نصب cURL در ubuntu است:

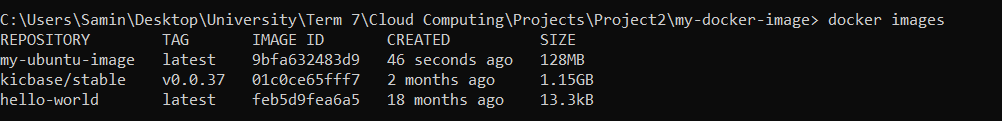


حالا داکر فایل را ذخیره کرده و docker image را میسازیم:

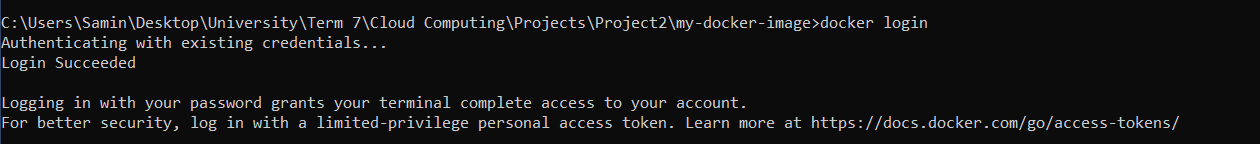
مدتی طول میکشد تا docker image ساخته شود:



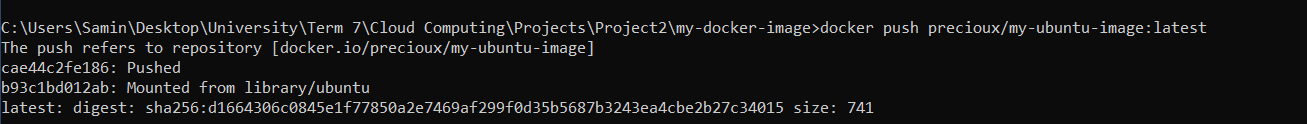
حالا آن را چک میکنیم:

همانطور که مشاهده میشود در لیست تصاویز داکری که در سیستم وجود دارد تصویر جدید هم وجود دارد پس به درستی ساخته شده است.

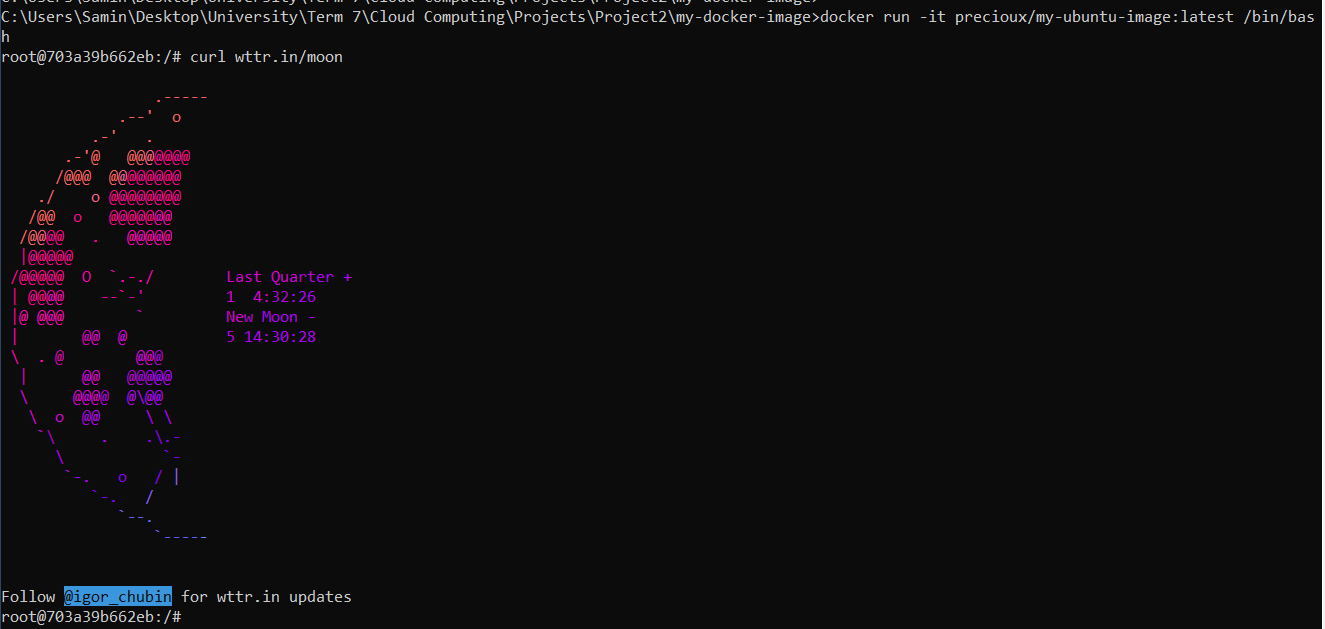
میخواهیم این تصویر را روی داکرهاب قرار دهیم:





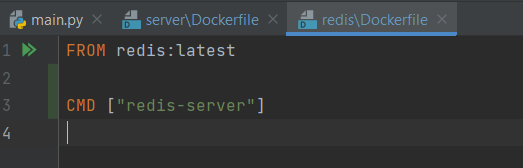


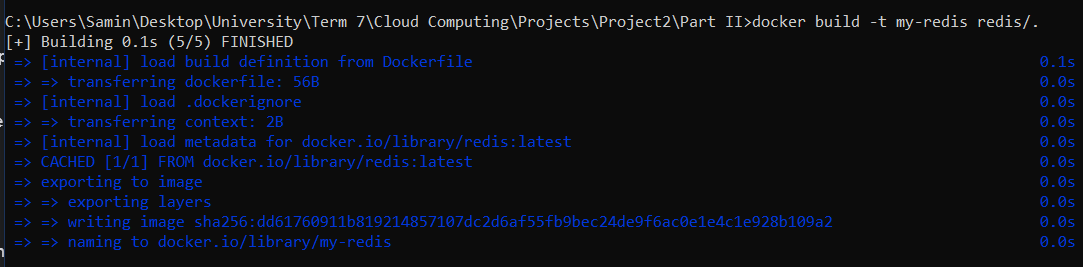
حالا برای تست آن را دانلود و ران میکنیم:

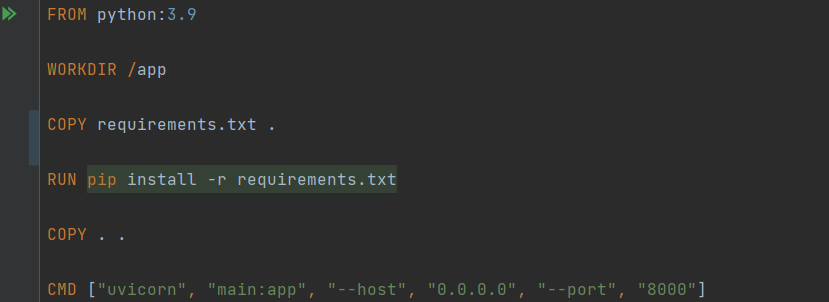


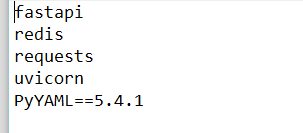
همانطور که مشاهده میشود یک بش بر پایه لینوکس بالا امد که با استفاده از آن کرل مورد نظر را زدیم و پاسخ گرفتیم.

گام دوم

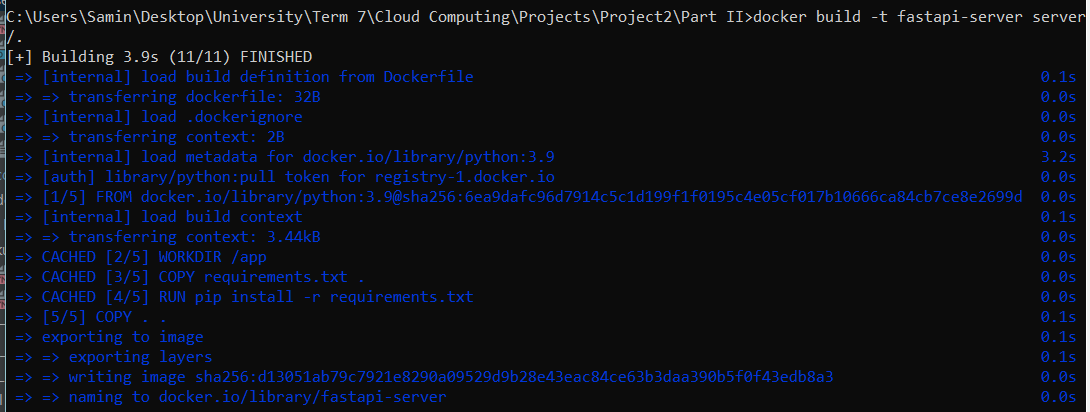
برای انجام این بخش از APILayer استفاده میکنیم. برای بالا آوردن یک کانتینر از کش ردیس بصورت زیر عمل میکنیم:

حالا داکرفایل سرور را مینویسیم:

برای این بخش نیاز به requirements.txt داریم:



حالا تصویر سرور را ایجاد میکنیم:

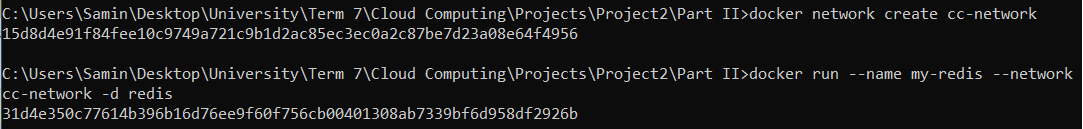


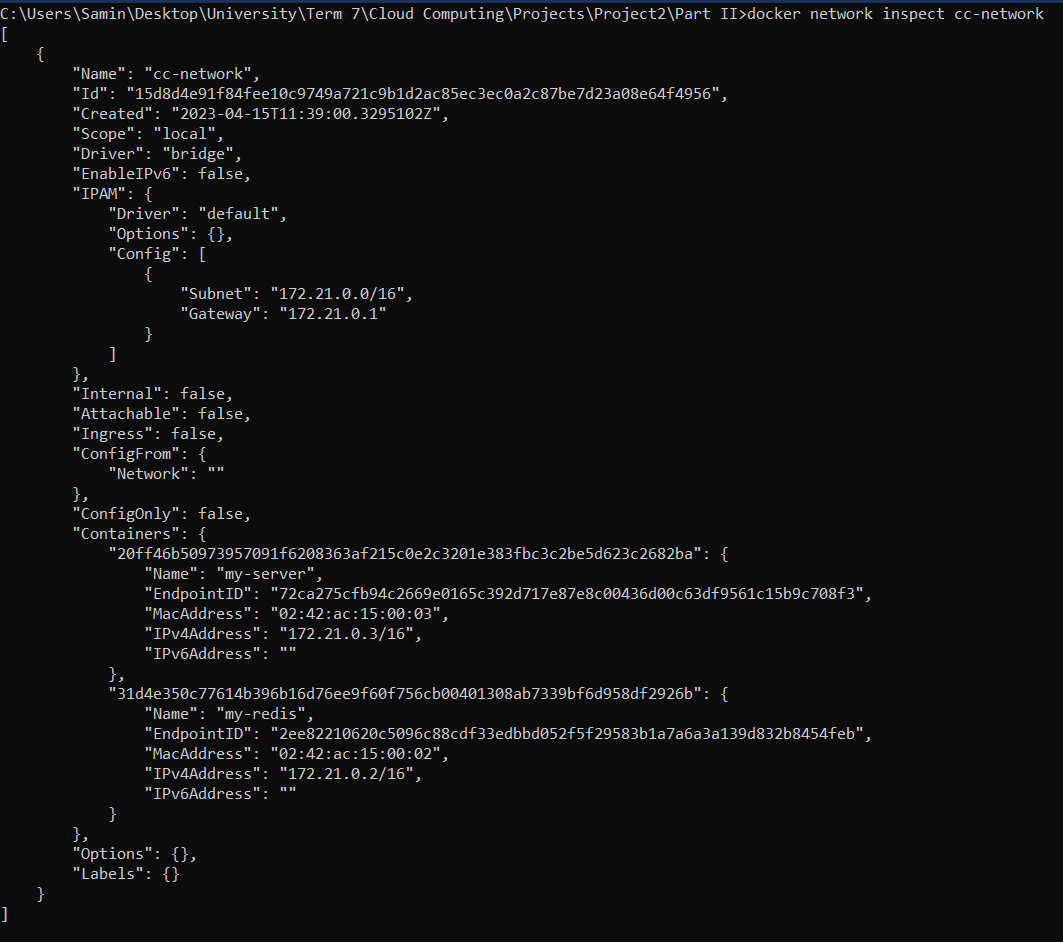
حالا این تصویر را روی داکر هاب قرار میدهیم



برای ردیس یک volume تعریف میکنیم:

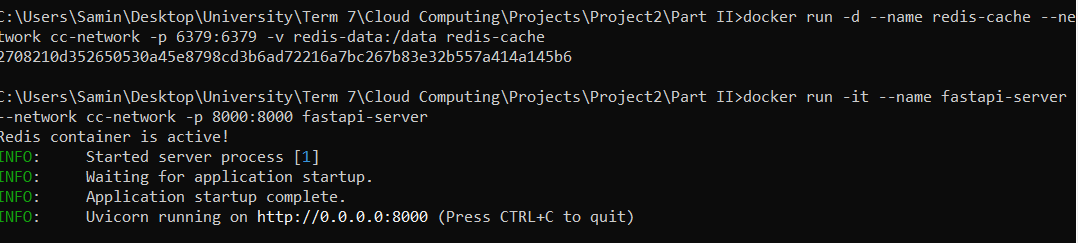


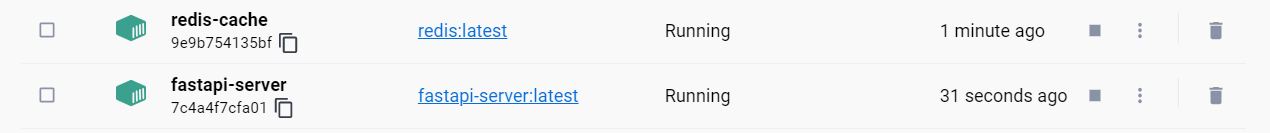
حالا لازم داریم که شبکه ای تعریف کنیم که این دو کانتینر باهم در ارتباط باشند:



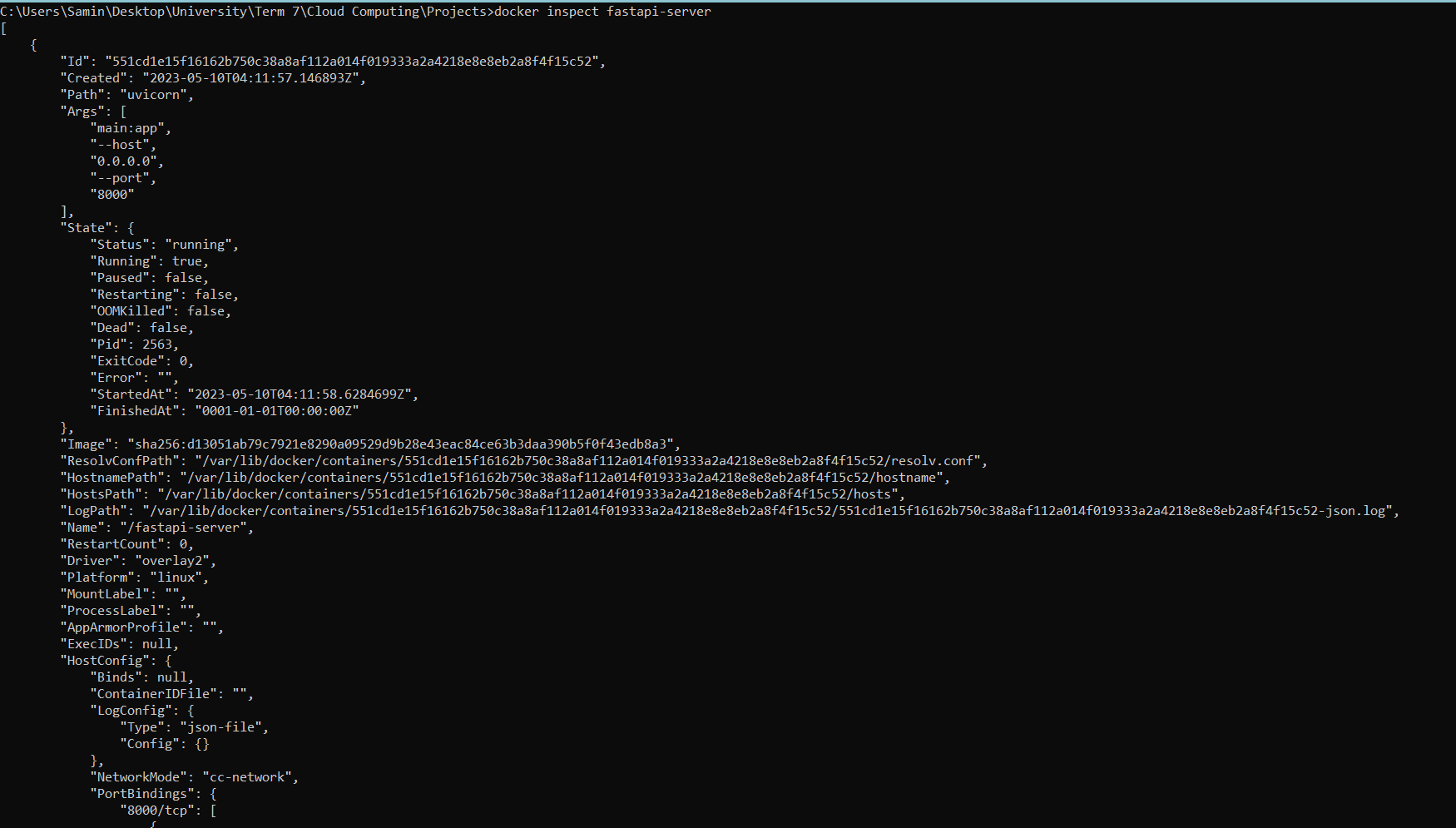
همانطور که مشاهده میشود شبکه تشکیل شده و دو کانتینر به آن متصل شده اند. کانتینر سرور روی 172.21.0.3 در حال شنیدن و کانتینر ردیس روی 172.21.0.2 در حال شنیدن میباشد.

کانتینر هارا بالا می آوریم:

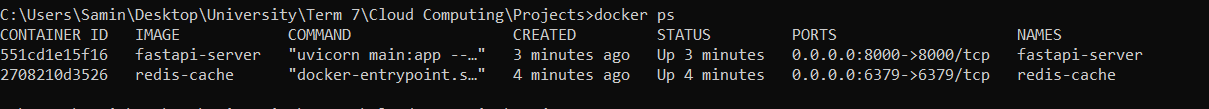




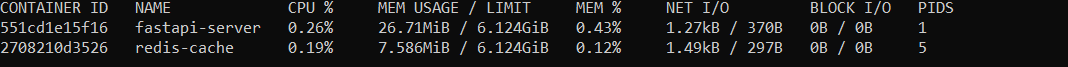
با دستور docker inspect تصویر fastapi-server را بررسی میکنیم:



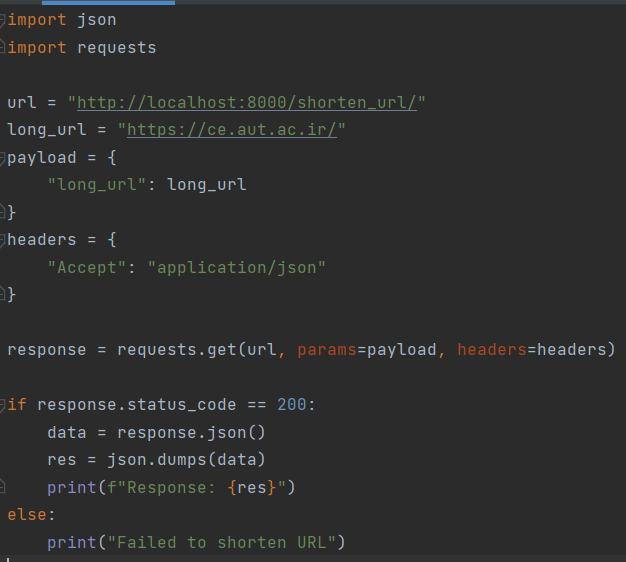
کانتینر های موجود در سیستم:



منابع مورد استفاده این کانتینرها بصورت زیر است:



حالا از برنامه تست میگیریم و آدرس <https://ce.aut.ac.ir/> را برای آن ارسال میکنیم:



نتیجه بصورت زیر خواهد بود:

Response:

{\"longUrl\": \"https://ce.aut.ac.ir/\",

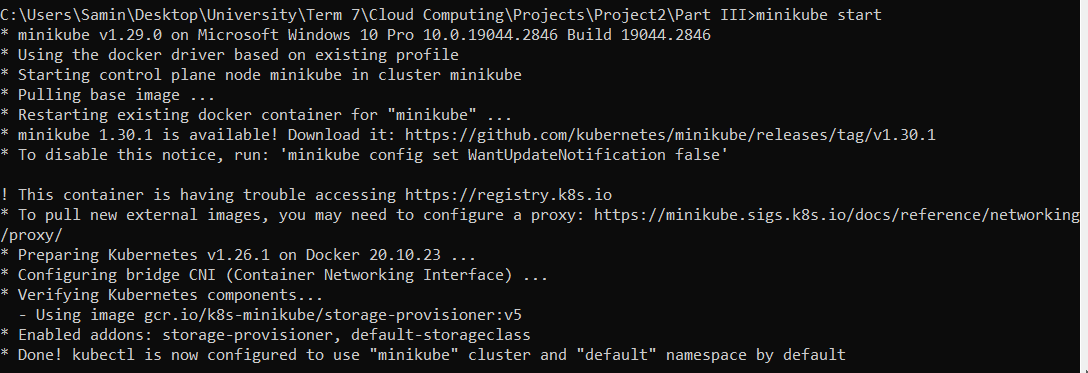
\"shortUrl\": \"https://p1.rs/Ohh5v\",

\"isCached\": false,

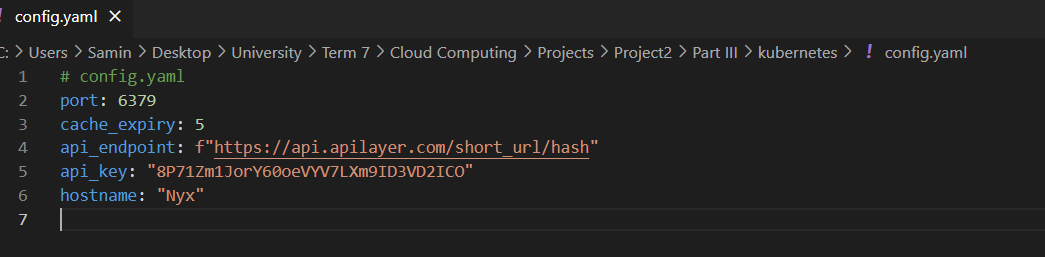
\"hostname\": \"Nyx\"}

گام سوم

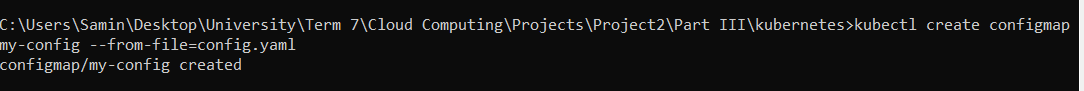
ابتدا مینیکوب را روی داکر بالا می آوریم:

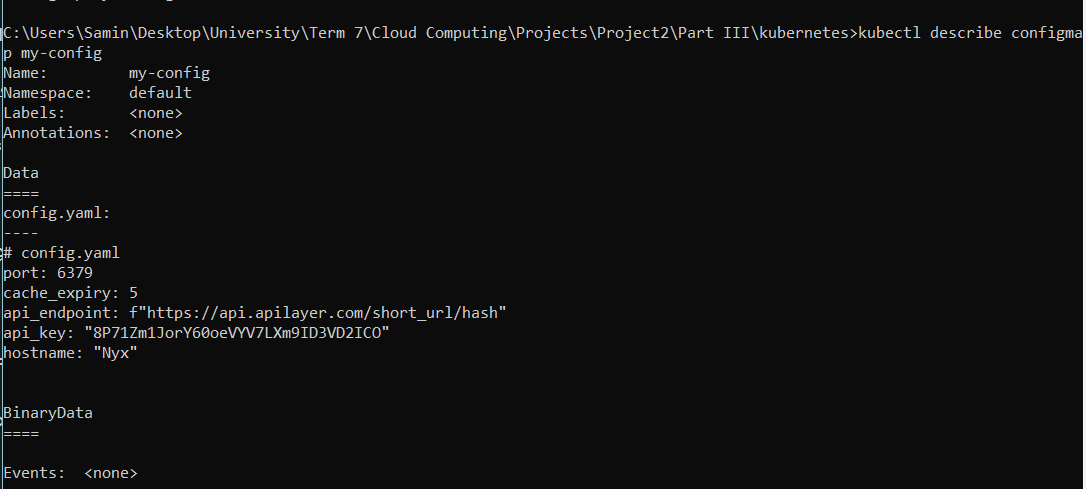


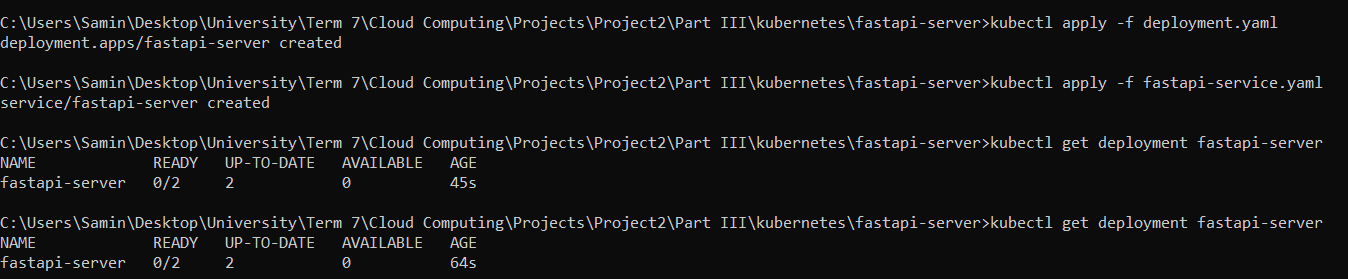
* برای ساخت configmap از فایل کانفیگ زیر استفاده میکنیم:



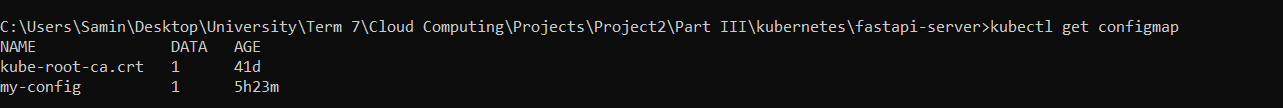
حالا از این فایل کانفیگ مپ میسازیم:



آن را بررسی میکنیم:

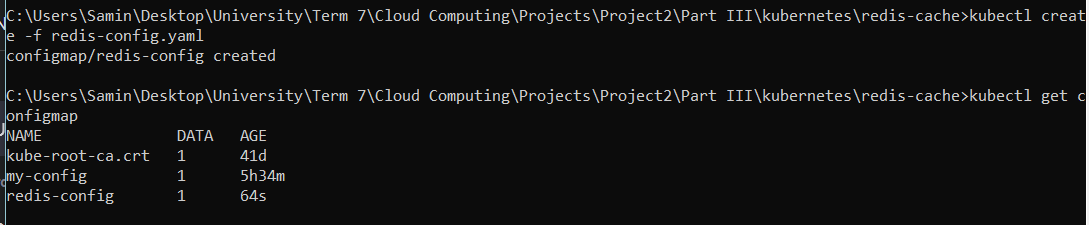
با انجام این مراحل کانفیگ مپ اپلای میشود، حالا فایل های خواسته شده دیگر را هم تعریف کرده سپس به ترتیب اپلای میکنیم:

با بررسی مشاهده میشود که فایل ها به درستی ساخته شده اند.

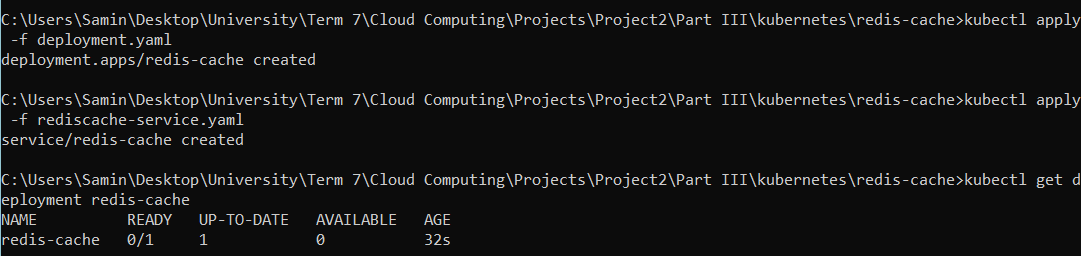
همچنین کانفیگ مپ را مجددا چک میکنیم:

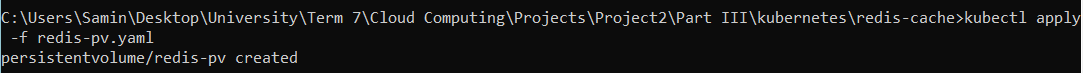
سه فایل قبلی را برای ردیس کش میسازیم:

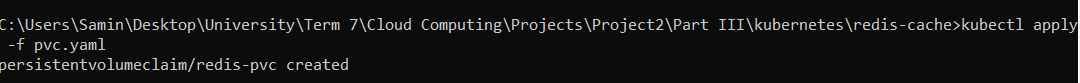
Configmap



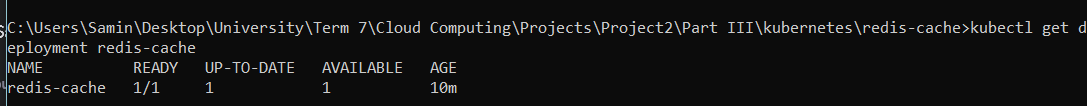
Deployment and service

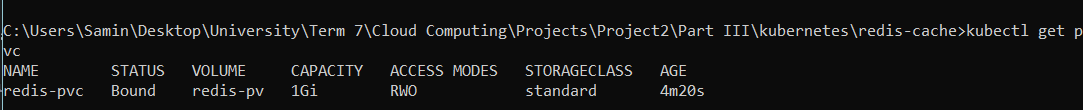


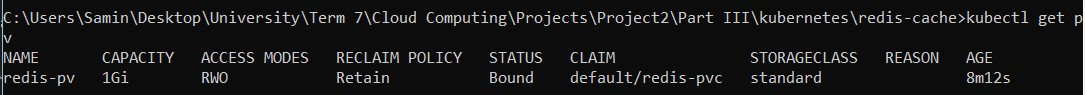
دو فایل دیگر را میسازیم:



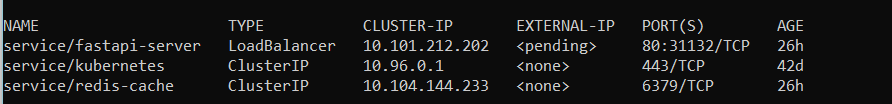
حالا فایل های قرار داده شده را بررسی میکنیم:



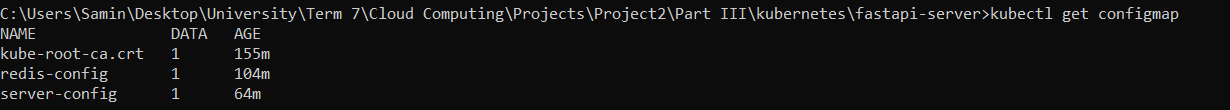




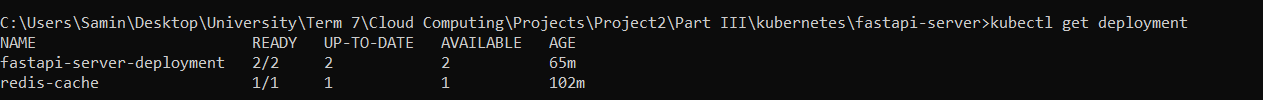
برای بررسی آدرس ها بصورت زیر عمل میکنیم:



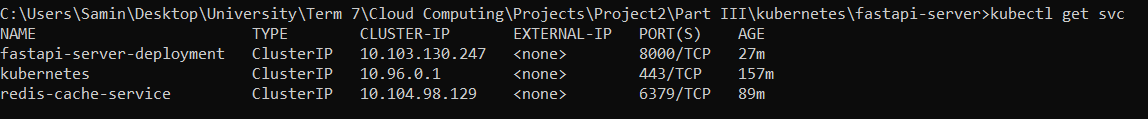
نهایتا مجددا منابع را با دستور kubectl get بررسی میکنیم:

کانفیگ مپ ها:

همانطور که مشاهده میشود کانفیگ مپ برای سرور و ردیس به درستی ایجاد شده اند.

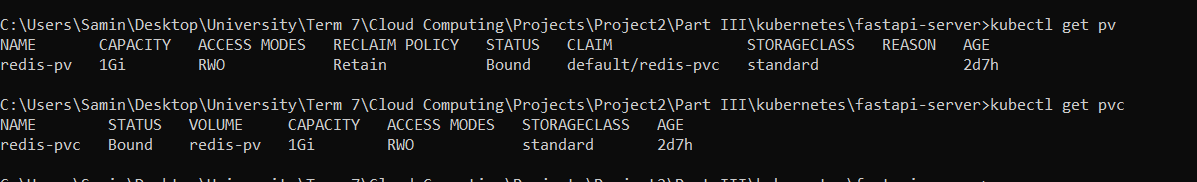
دیپلویمنت ها:

همانطور که مشاهده میشود فایل های دیپلویمنت به درستی ساخته شده اند

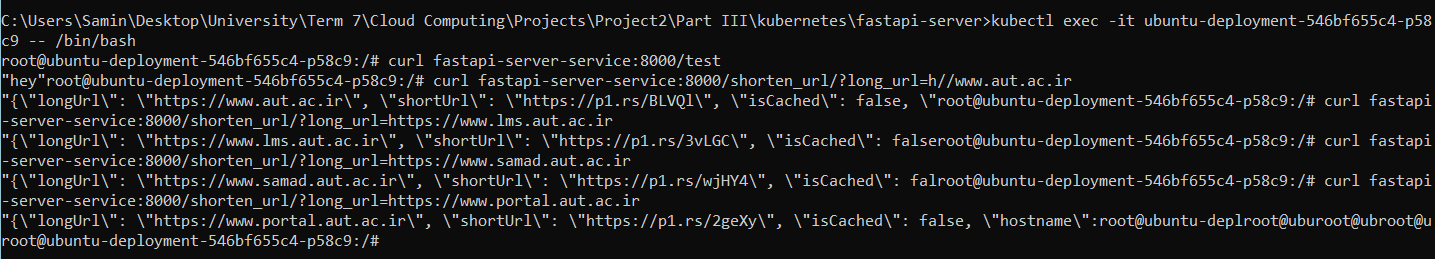
سرویس ها:

همانطور که مشاهده میشود سرویس ها به درستی ساخته شده اند.

Pv,pvc :

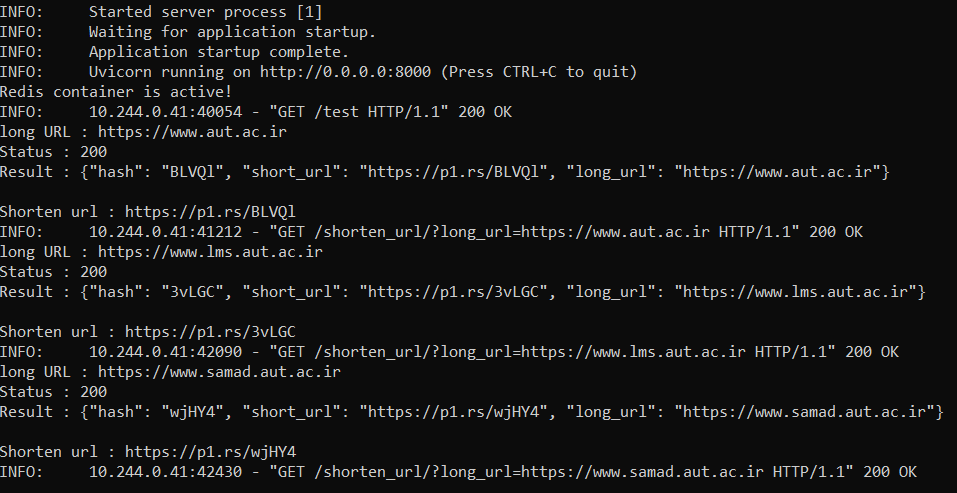


همانطور که مشاهده میشود فایل ها به درستی تعریف و ایجاد شده اند.

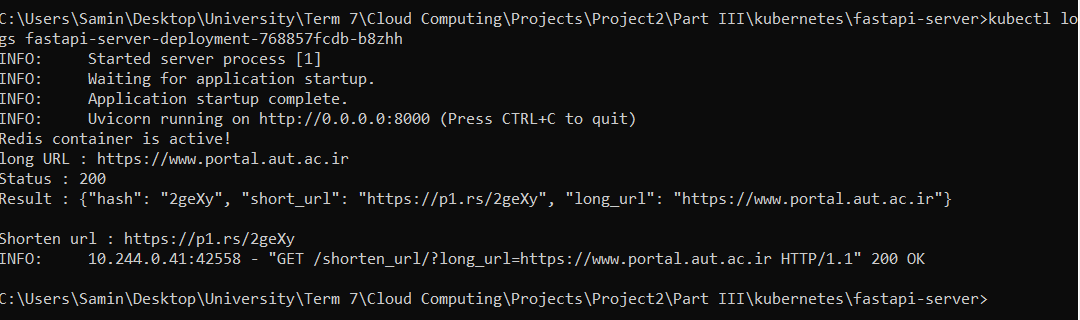
برای نمایش توزیع درخواست ها بین دو پاد سرور چندین درخواست به سرویس سرور میزنیم:

حالا با لاگ گرفتن از پاد ها بررسی میکنیم این درخواست هارا چه کسی پاسخ گفته:

پاد اول سه درخواست را جواب داده:



و پاد دوم یکی از درخواست هارا:



بنابراین درخواست ها بین این دو پاد سرور توزیع شده بوده است.