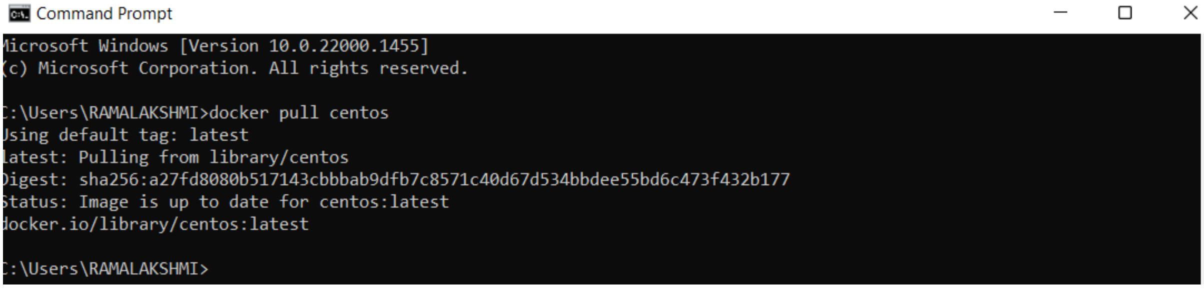
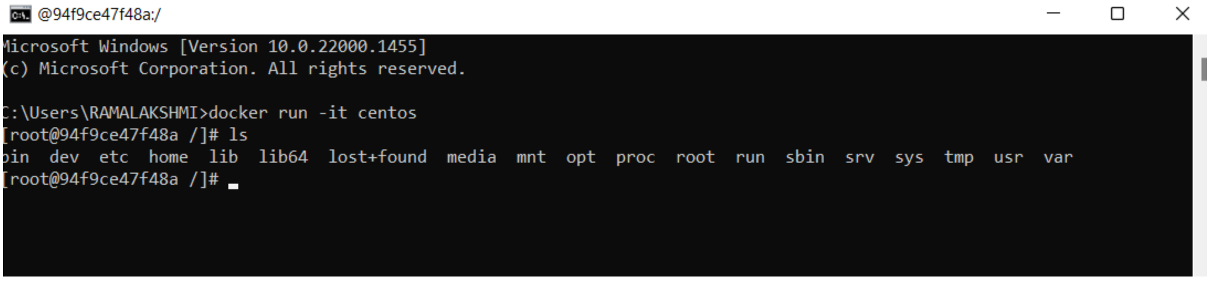
Q1) Pull any image from the docker hub, create its container, and execute it showing the output.

Pulling an image:



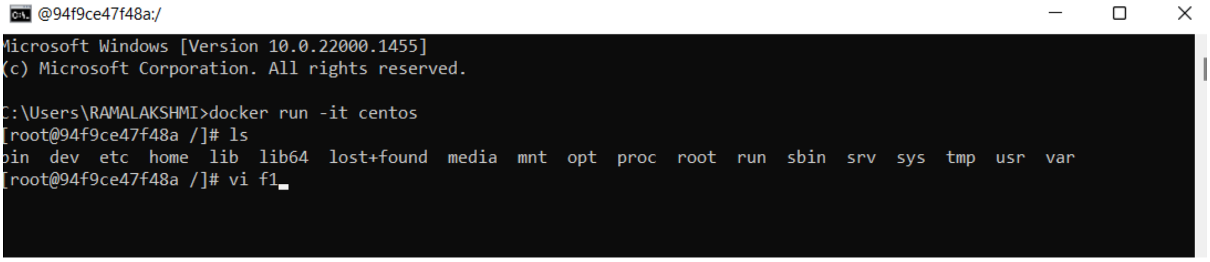
```
icrosoft Windows [Version 10.0.22000.1455]
(c) Microsoft Corporation. All rights reserved.

:\Users\RAMALAKSHMI>docker pull centos
Jsing default tag: latest
latest: Pulling from library/centos
)igest: sha256:a27fd8080b517143cbbbab9dfb7c8571c40d67d534bbdee55bd6c473f432b177
Status: Image is up to date for centos:latest
docker.io/library/centos:latest

:\Users\RAMALAKSHMI>
```

Running it:



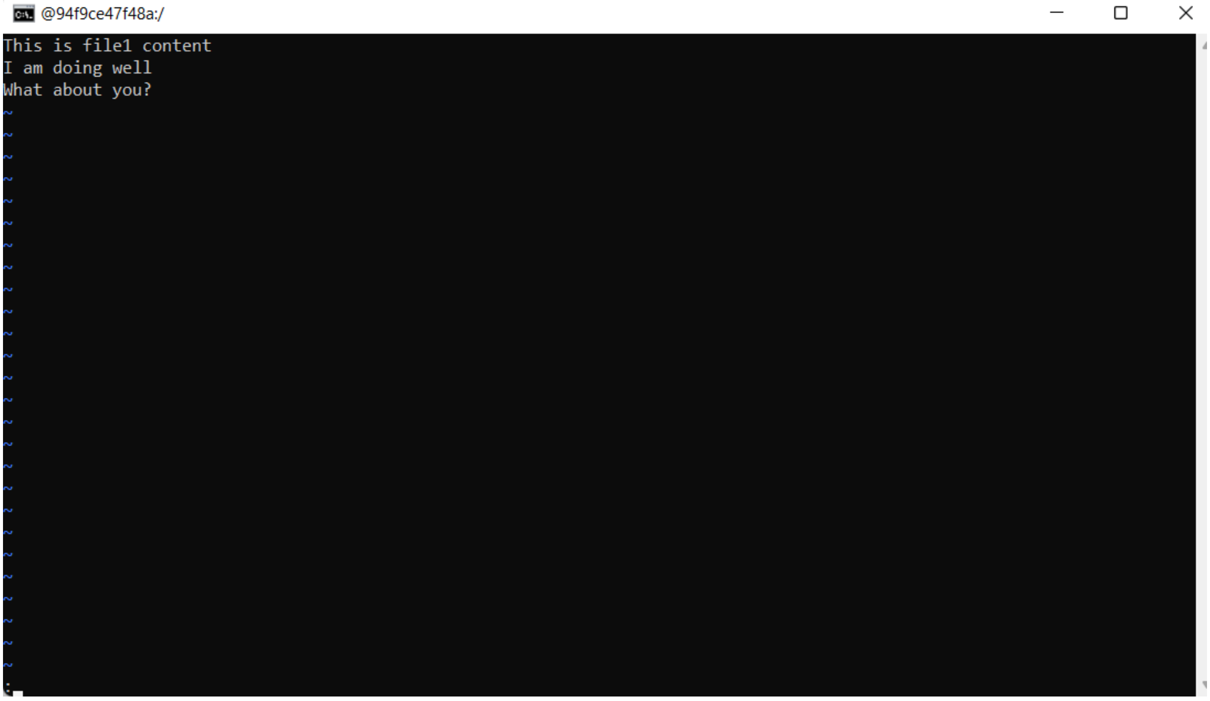
```
icrosoft Windows [Version 10.0.22000.1455]
(c) Microsoft Corporation. All rights reserved.

:\Users\RAMALAKSHMI>docker run -it centos
[root@94f9ce47f48a /]# ls
pin  dev  etc  home  lib  lib64  lost+found  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var
[root@94f9ce47f48a /]#
```

Creating a file named f1



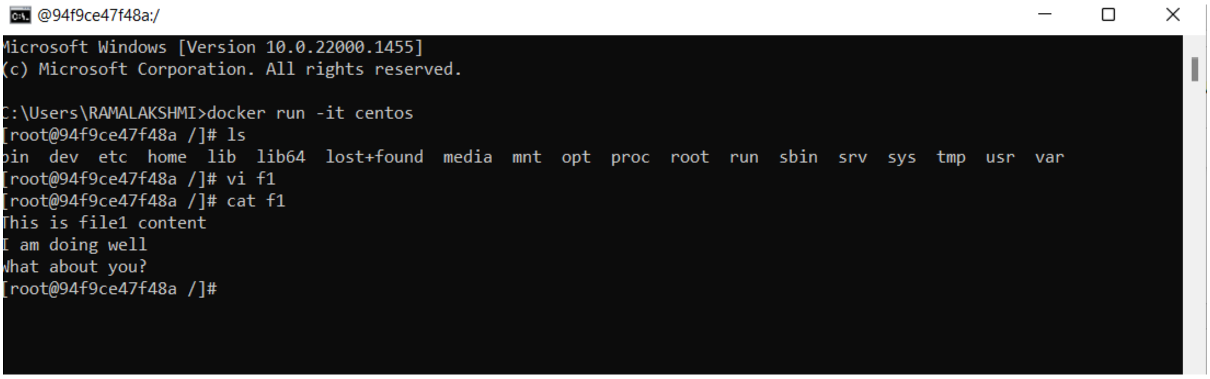
```
icrosoft Windows [Version 10.0.22000.1455]
(c) Microsoft Corporation. All rights reserved.

:\Users\RAMALAKSHMI>docker run -it centos
[root@94f9ce47f48a /]# ls
pin  dev  etc  home  lib  lib64  lost+found  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var
[root@94f9ce47f48a /]# vi f1
```

Entering contents into f1



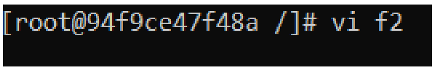
```
This is file1 content
I am doing well
What about you?
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
:
```

Displaying Contents of f1 using cat command



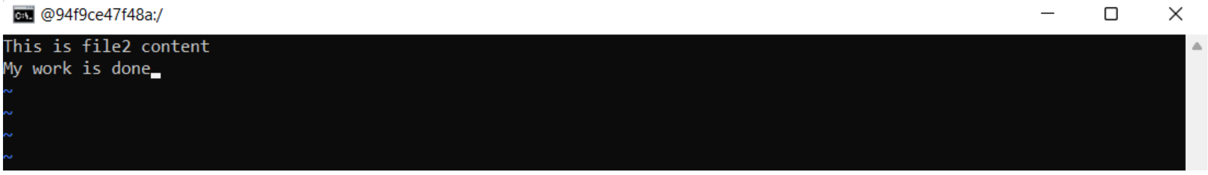
```
Microsoft Windows [Version 10.0.22000.1455]
(c) Microsoft Corporation. All rights reserved.

C:\Users\RAMALAKSHMI>docker run -it centos
[root@94f9ce47f48a /]# ls
bin  dev  etc  home  lib  lib64  lost+found  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var
[root@94f9ce47f48a /]# vi f1
[root@94f9ce47f48a /]# cat f1
This is file1 content
I am doing well
What about you?
[root@94f9ce47f48a /]#
```

Creation of f2



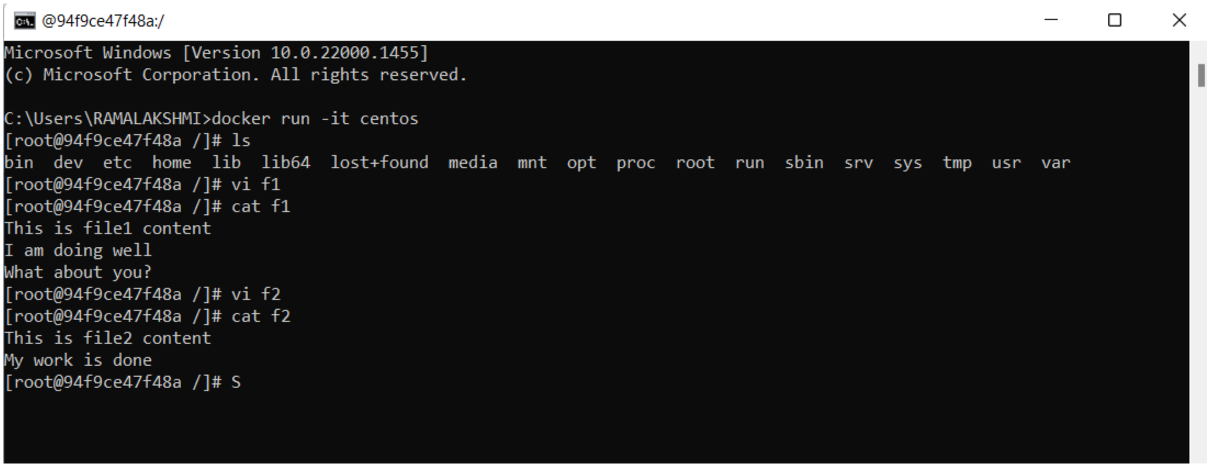
```
[root@94f9ce47f48a /]# vi f2
```

Entering contents into f2



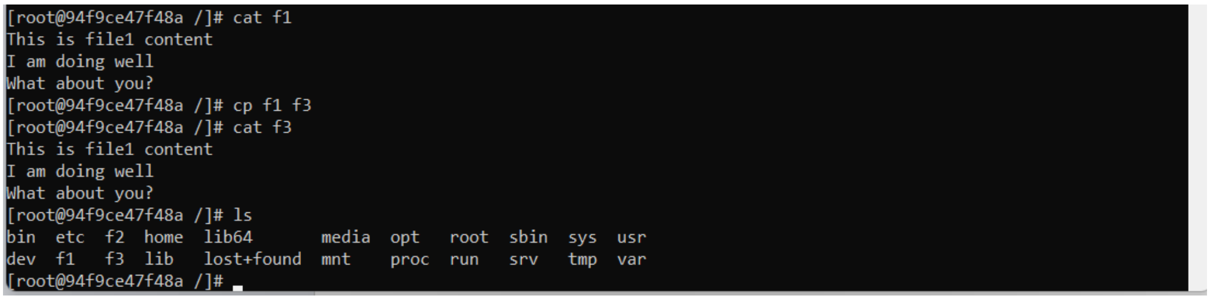
```
This is file2 content
My work is done
~
~
~
~
```

Displaying file f2



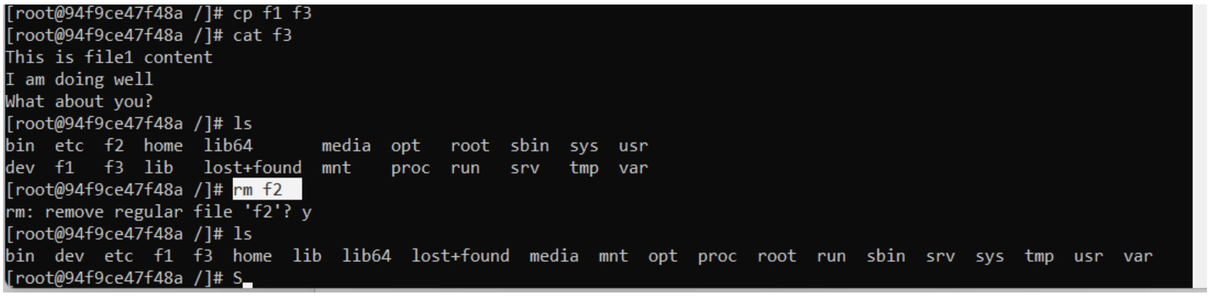
```
Microsoft Windows [Version 10.0.22000.1455]
(c) Microsoft Corporation. All rights reserved.

C:\Users\RAMALAKSHMI>docker run -it centos
[root@94f9ce47f48a /]# ls
bin  dev  etc  home  lib  lib64  lost+found  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var
[root@94f9ce47f48a /]# vi f1
[root@94f9ce47f48a /]# cat f1
This is file1 content
I am doing well
What about you?
[root@94f9ce47f48a /]# vi f2
[root@94f9ce47f48a /]# cat f2
This is file2 content
My work is done
[root@94f9ce47f48a /]# S
```

Coping contents of f1 to f3 using copy commands



```
[root@94f9ce47f48a /]# cat f1
This is file1 content
I am doing well
What about you?
[root@94f9ce47f48a /]# cp f1 f3
[root@94f9ce47f48a /]# cat f3
This is file1 content
I am doing well
What about you?
[root@94f9ce47f48a /]# ls
bin  etc  f2  home  lib64       media  opt   root  sbin  sys  usr
dev  f1   f3  lib   lost+found  mnt    proc  run   srv   tmp  var
[root@94f9ce47f48a /]# _
```

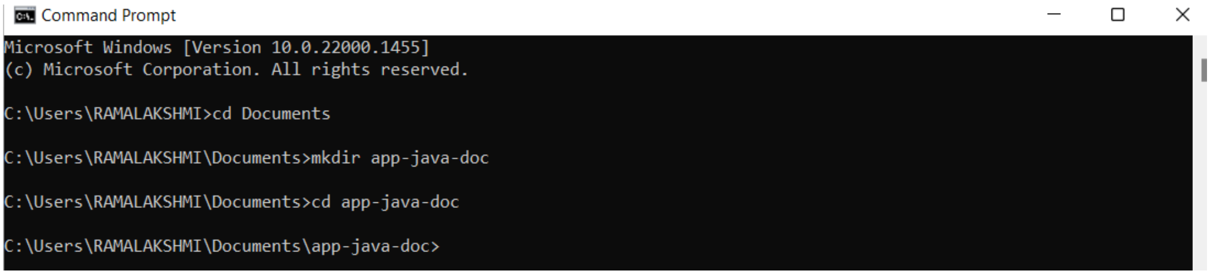
Removing file f2



```
[root@94f9ce47f48a /]# cp f1 f3
[root@94f9ce47f48a /]# cat f3
This is file1 content
I am doing well
What about you?
[root@94f9ce47f48a /]# ls
bin  etc  f2  home  lib64       media  opt   root  sbin  sys  usr
dev  f1   f3  lib   lost+found  mnt    proc  run   srv   tmp  var
[root@94f9ce47f48a /]# rm f2
rm: remove regular file 'f2'? y
[root@94f9ce47f48a /]# ls
bin  dev  etc  f1  f3  home  lib  lib64  lost+found  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var
[root@94f9ce47f48a /]# S_
```

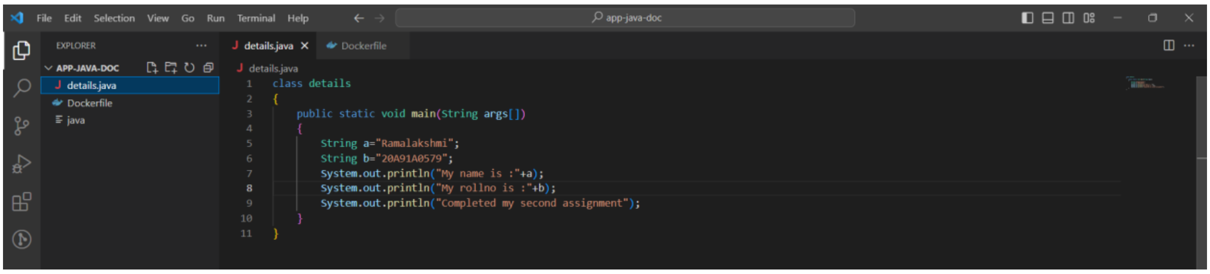
Q2) Create the basic java application, generate its image with necessary files, and execute it with docker.

- I have created a folder app-java-doc in documents folder and switched to that using cd command.



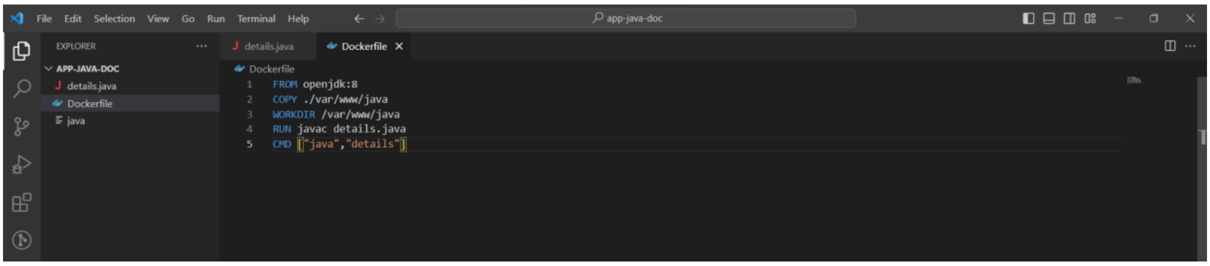
```
Microsoft Windows [Version 10.0.22000.1455]
(c) Microsoft Corporation. All rights reserved.

C:\Users\RAMALAKSHMI>cd Documents

C:\Users\RAMALAKSHMI\Documents>mkdir app-java-doc

C:\Users\RAMALAKSHMI\Documents>cd app-java-doc

C:\Users\RAMALAKSHMI\Documents\app-java-doc>
```

- Opened that folder in Visual studio code and created a java file named “details”.

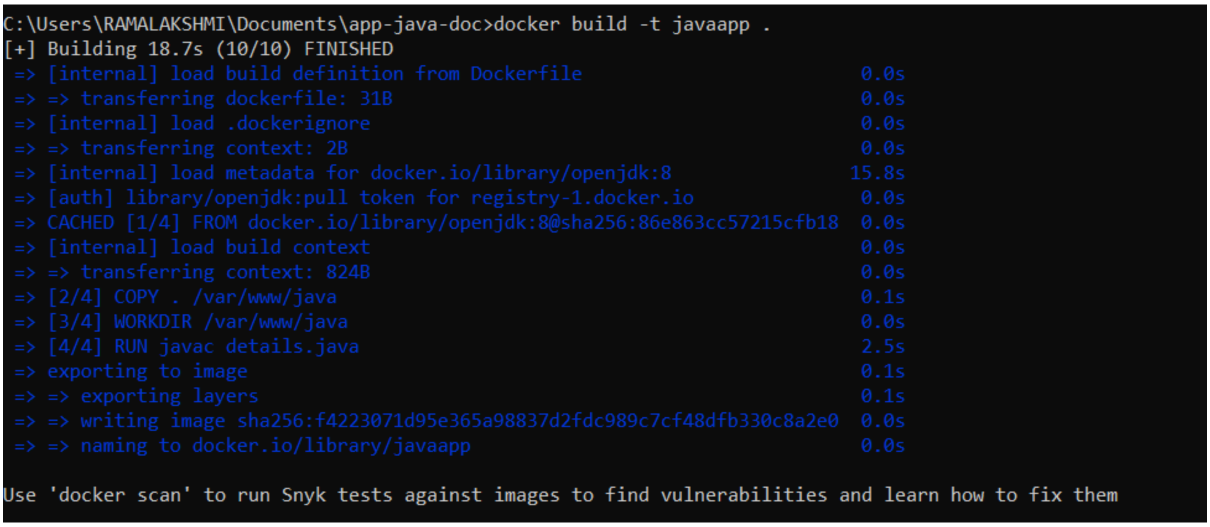


```java
class details
{
    public static void main(String args[])
    {
        String a="Ramalakshmi";
        String b="20A91A0579";
        System.out.println("My name is :"+a);
        System.out.println("My rollno is :"+b);
        System.out.println("Completed my second assignment");
    }
}
```

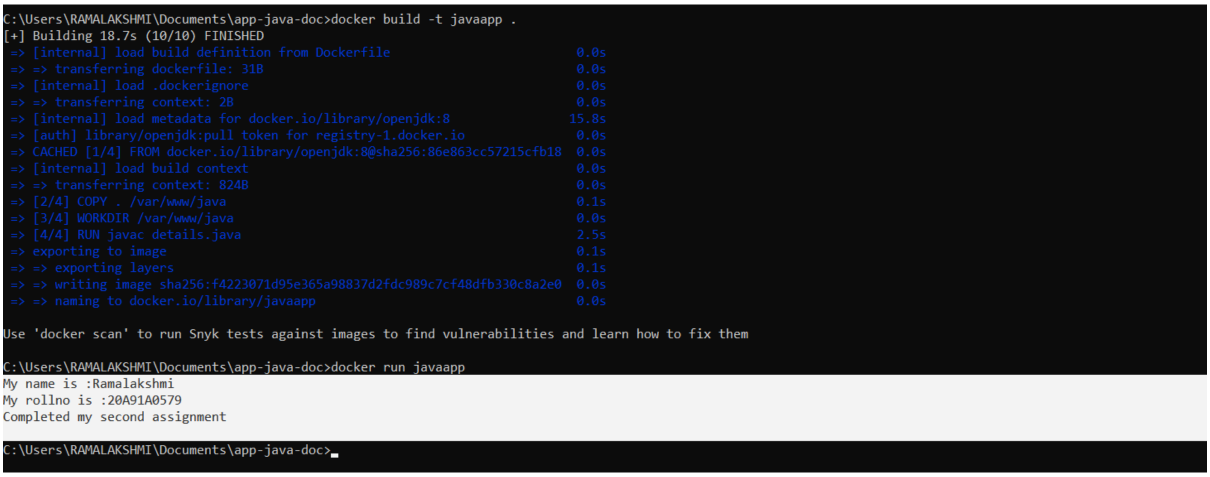
- Created a Dockerfile



```
FROM openjdk:8
COPY ./var/www/java
WORKDIR /var/www/java
RUN javac details.java
CMD ["java","details"]
```



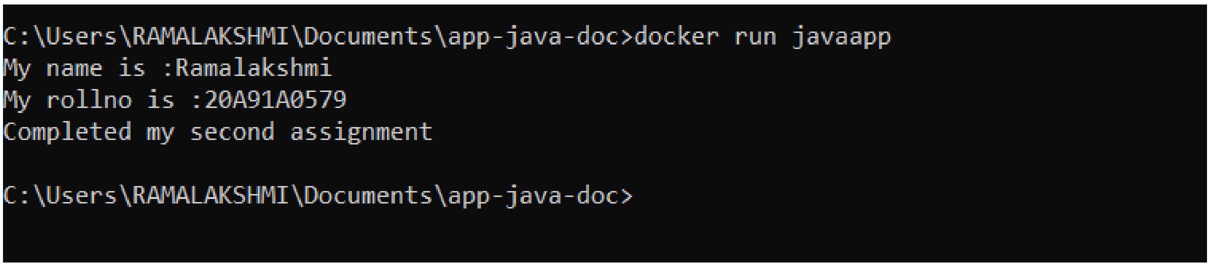
```
C:\Users\RAMALAKSHMI\Documents\app-java-doc>docker build -t javaapp .
[+] Building 18.7s (10/10) FINISHED
 => [internal] load build definition from Dockerfile                        0.0s
 => => transferring dockerfile: 31B                                         0.0s
 => [internal] load .dockerignore                                           0.0s
 => => transferring context: 2B                                             0.0s
 => [internal] load metadata for docker.io/library/openjdk:8               15.8s
 => [auth] library/openjdk:pull token for registry-1.docker.io              0.0s
 => CACHED [1/4] FROM docker.io/library/openjdk:8@sha256:86e863cc57215cfb18  0.0s
 => [internal] load build context                                           0.0s
 => => transferring context: 824B                                           0.0s
 => [2/4] COPY . /var/www/java                                              0.1s
 => [3/4] WORKDIR /var/www/java                                             0.0s
 => [4/4] RUN javac details.java                                            2.5s
 => exporting to image                                                      0.1s
 => => exporting layers                                                     0.1s
 => => writing image sha256:f4223071d95e365a98837d2fdc989c7cf48dfb330c8a2e0  0.0s
 => => naming to docker.io/library/javaapp                                  0.0s

Use 'docker scan' to run Snyk tests against images to find vulnerabilities and learn how to fix them
```



```
C:\Users\RAMALAKSHMI\Documents\app-java-doc>docker build -t javaapp .
[+] Building 18.7s (10/10) FINISHED
 => [internal] load build definition from Dockerfile                        0.0s
 => => transferring dockerfile: 31B                                         0.0s
 => [internal] load .dockerignore                                           0.0s
 => => transferring context: 2B                                             0.0s
 => [internal] load metadata for docker.io/library/openjdk:8               15.8s
 => [auth] library/openjdk:pull token for registry-1.docker.io              0.0s
 => CACHED [1/4] FROM docker.io/library/openjdk:8@sha256:86e863cc57215cfb18  0.0s
 => [internal] load build context                                           0.0s
 => => transferring context: 824B                                           0.0s
 => [2/4] COPY . /var/www/java                                              0.1s
 => [3/4] WORKDIR /var/www/java                                             0.0s
 => [4/4] RUN javac details.java                                            2.5s
 => exporting to image                                                      0.1s
 => => exporting layers                                                     0.1s
 => => writing image sha256:f4223071d95e365a98837d2fdc989c7cf48dfb330c8a2e0  0.0s
 => => naming to docker.io/library/javaapp                                  0.0s

Use 'docker scan' to run Snyk tests against images to find vulnerabilities and learn how to fix them
```

```
C:\Users\RAMALAKSHMI\Documents\app-java-doc>docker run javaapp
My name is :Ramalakshmi
My rollno is :20A91A0579
Completed my second assignment

C:\Users\RAMALAKSHMI\Documents\app-java-doc>
```

- Finally program executed successfully and below is the respective output.



```
C:\Users\RAMALAKSHMI\Documents\app-java-doc>docker run javaapp
My name is :Ramalakshmi
My rollno is :20A91A0579
Completed my second assignment

C:\Users\RAMALAKSHMI\Documents\app-java-doc>
```