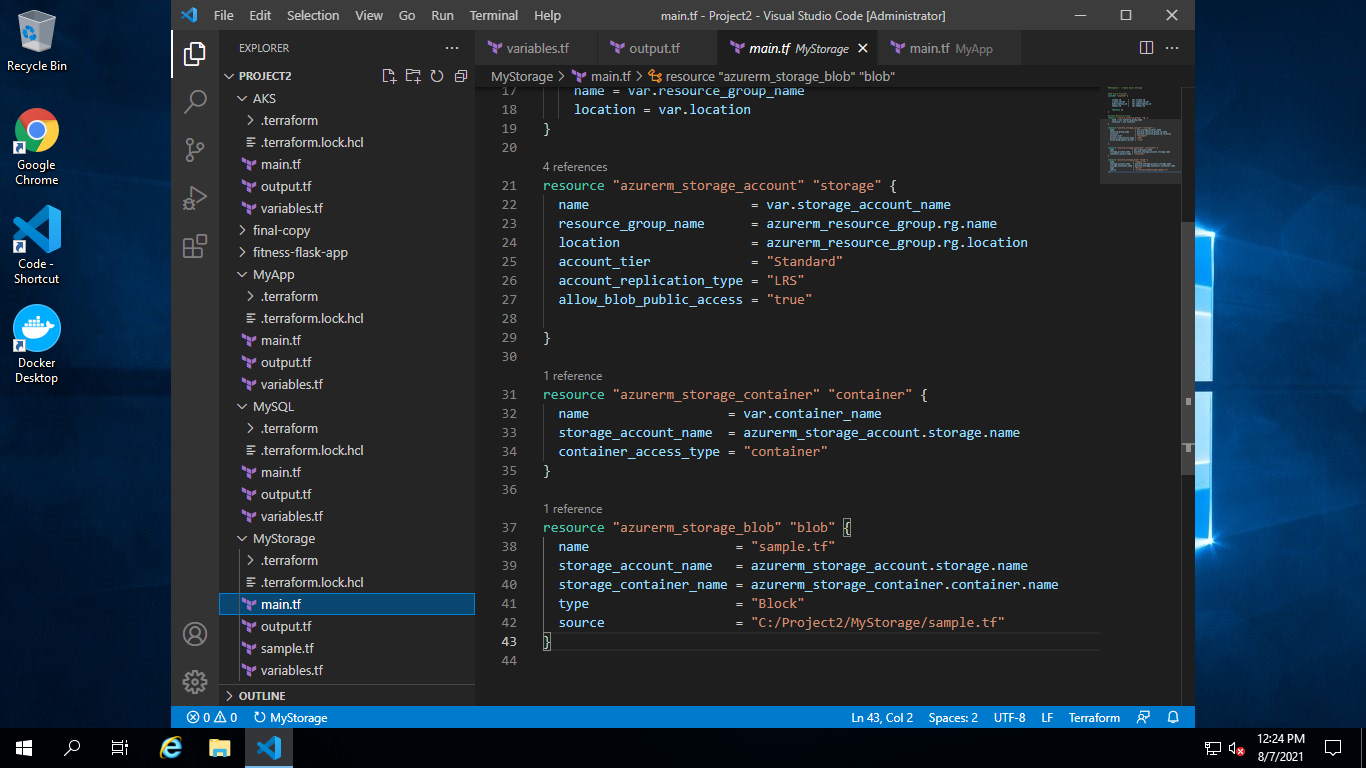
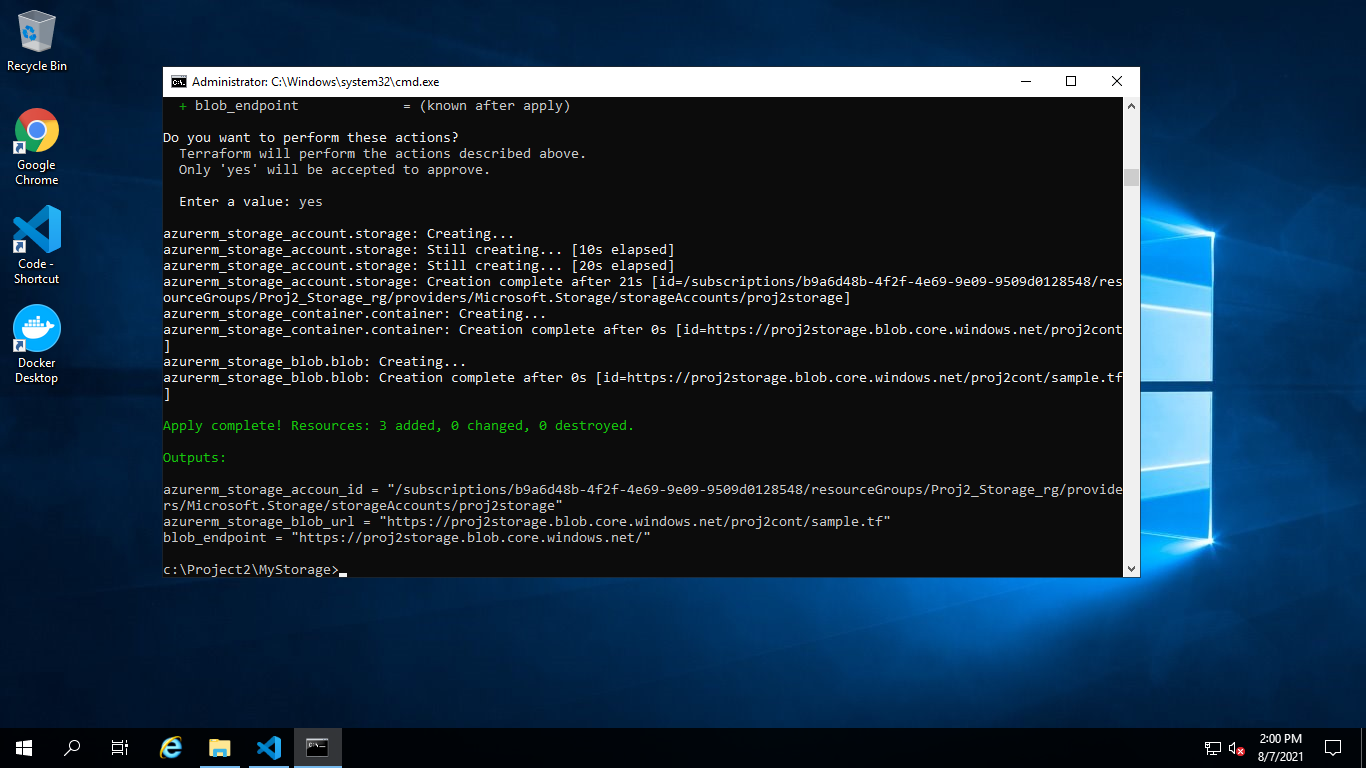
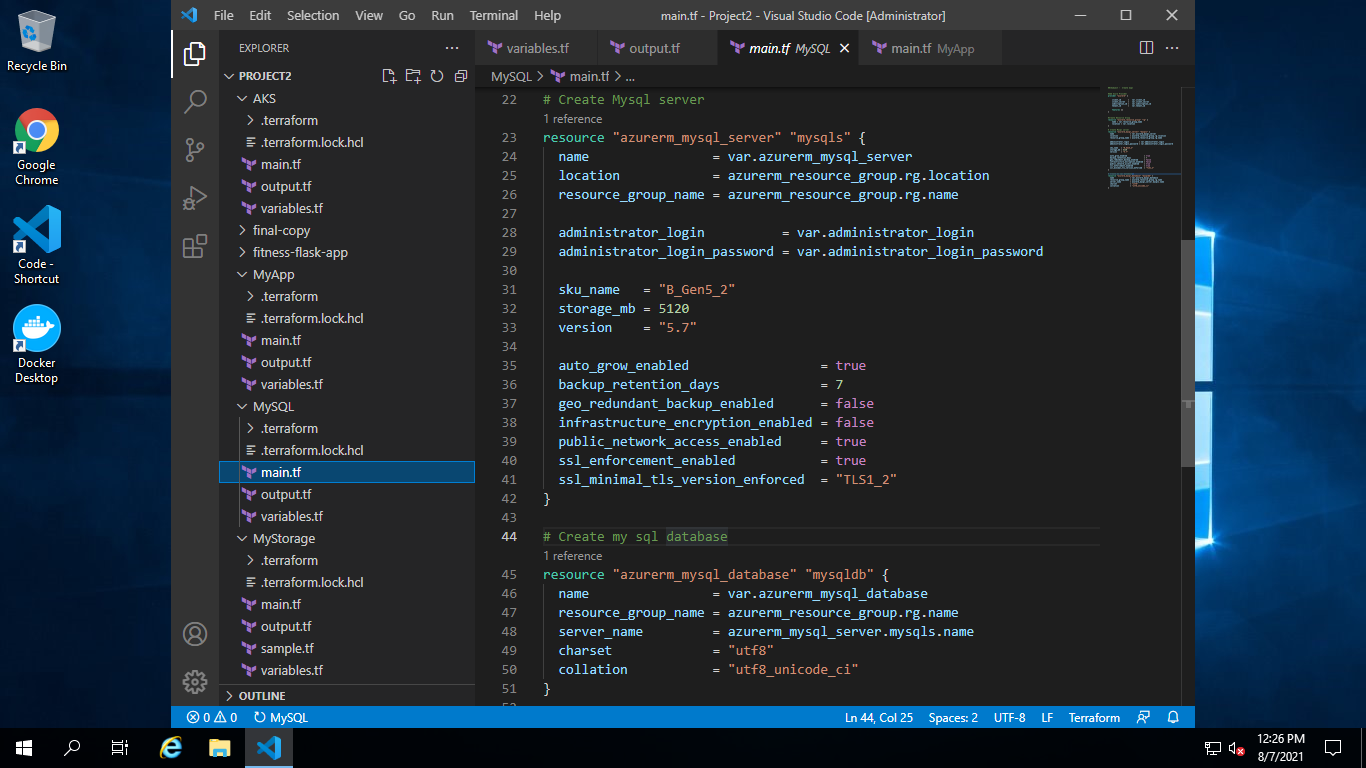
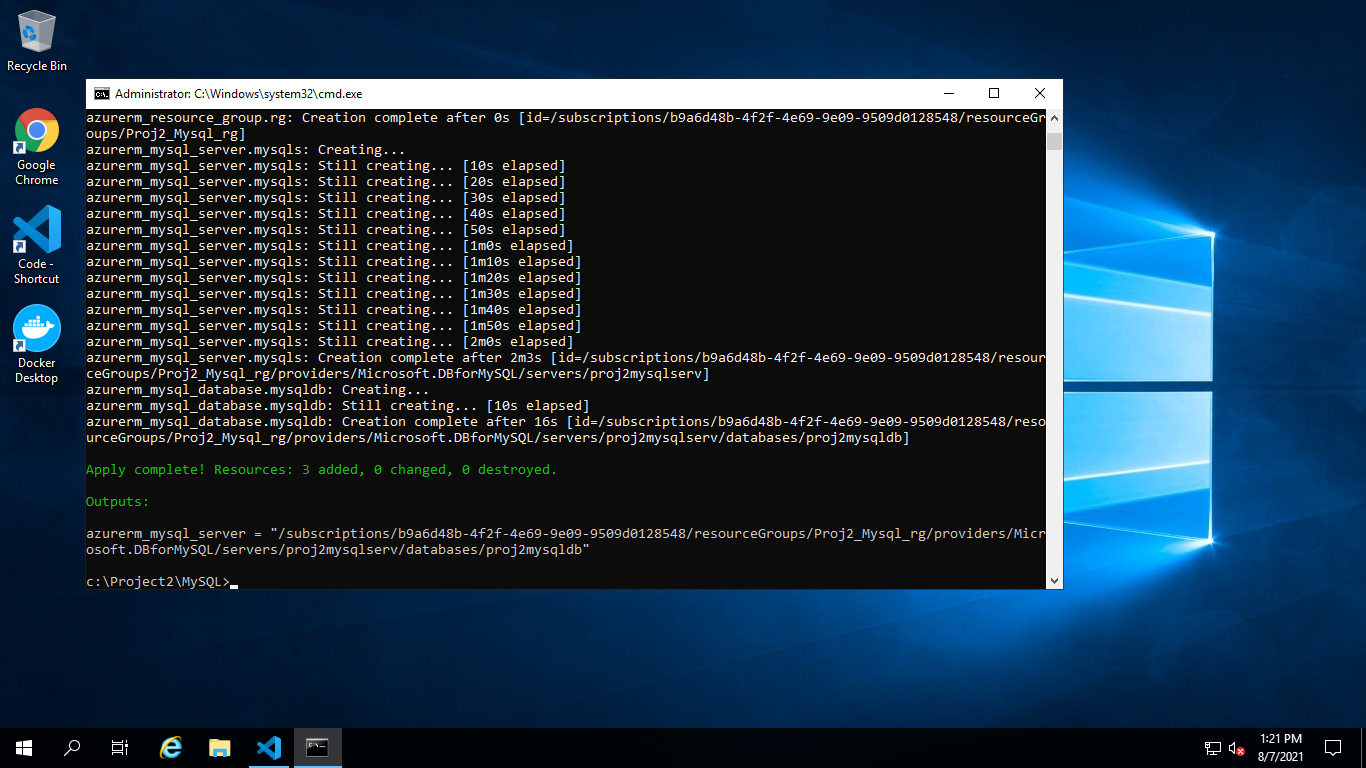
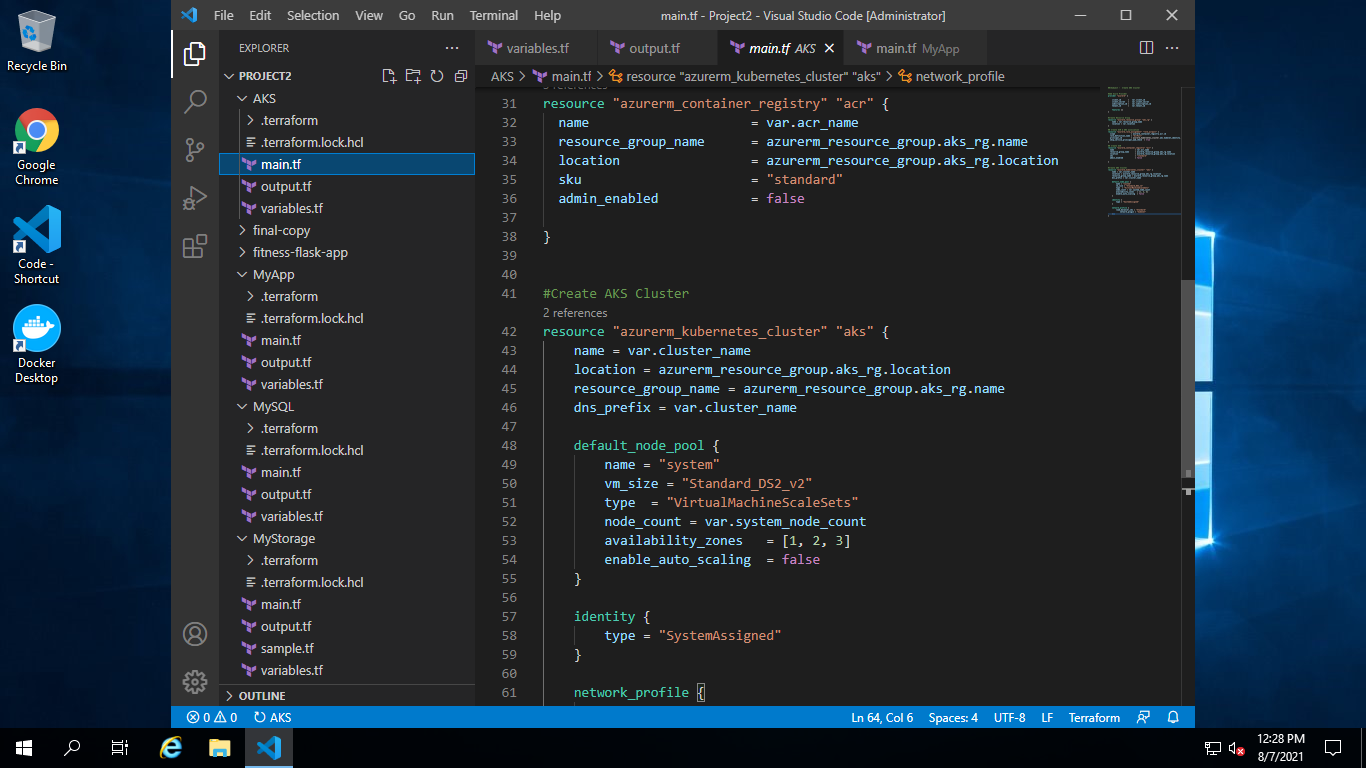
1. Created Storage Account via Terraform  
   

Below is output:

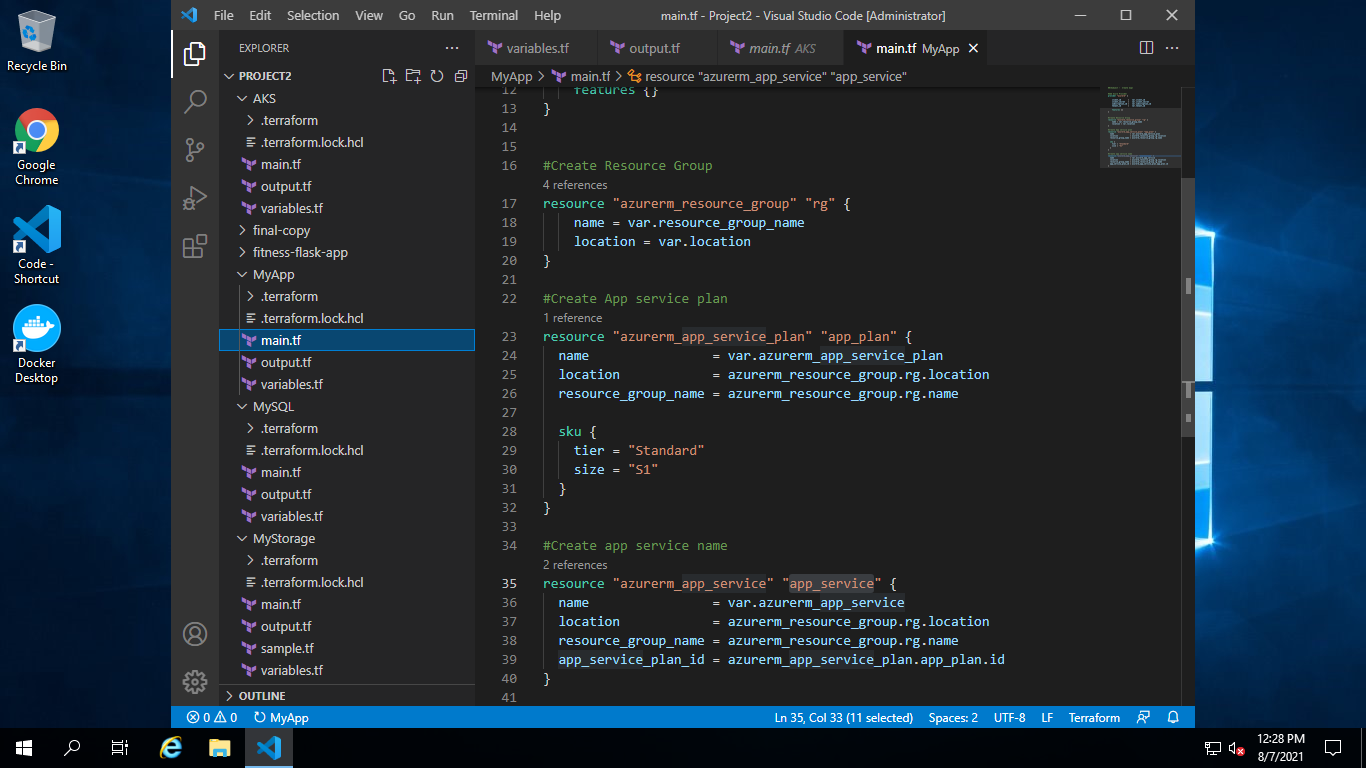


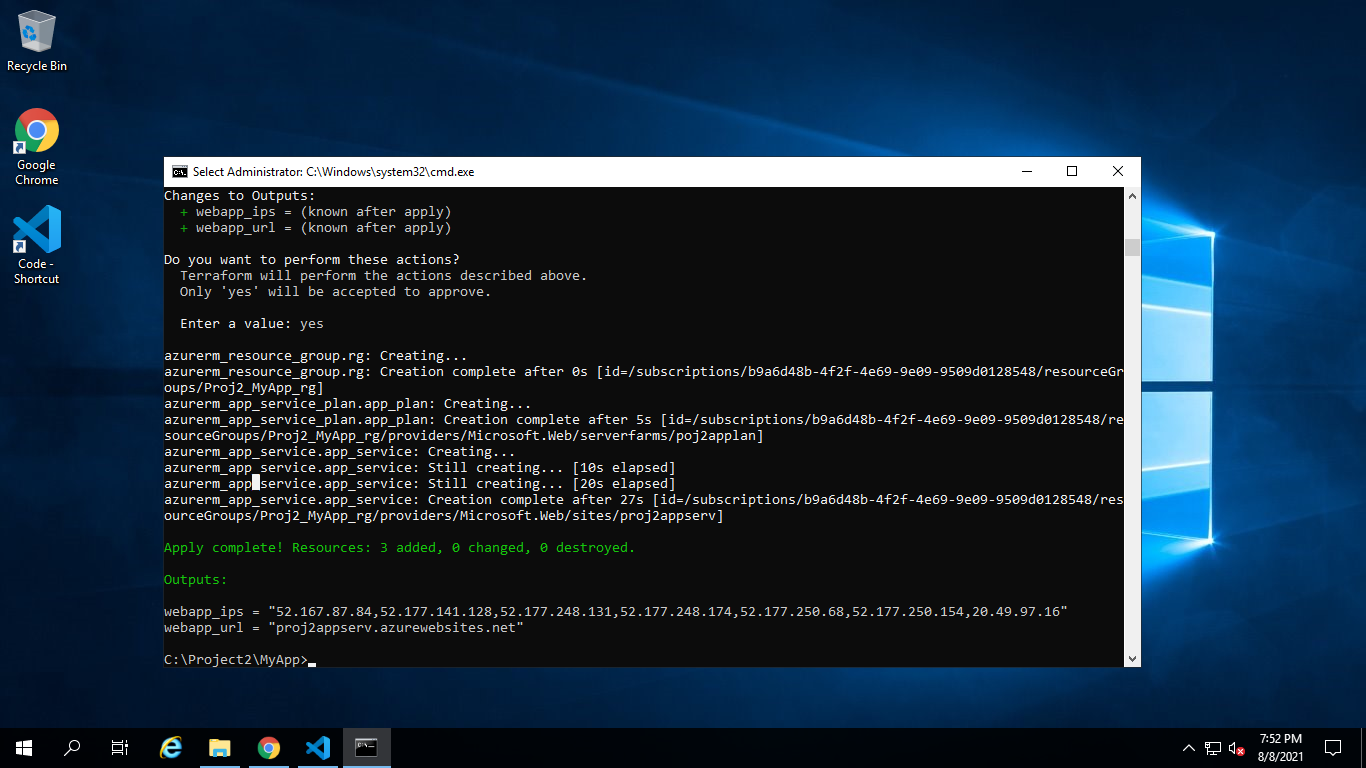
1. Created MySQL DB Server via Terraform  
     
   Below is output



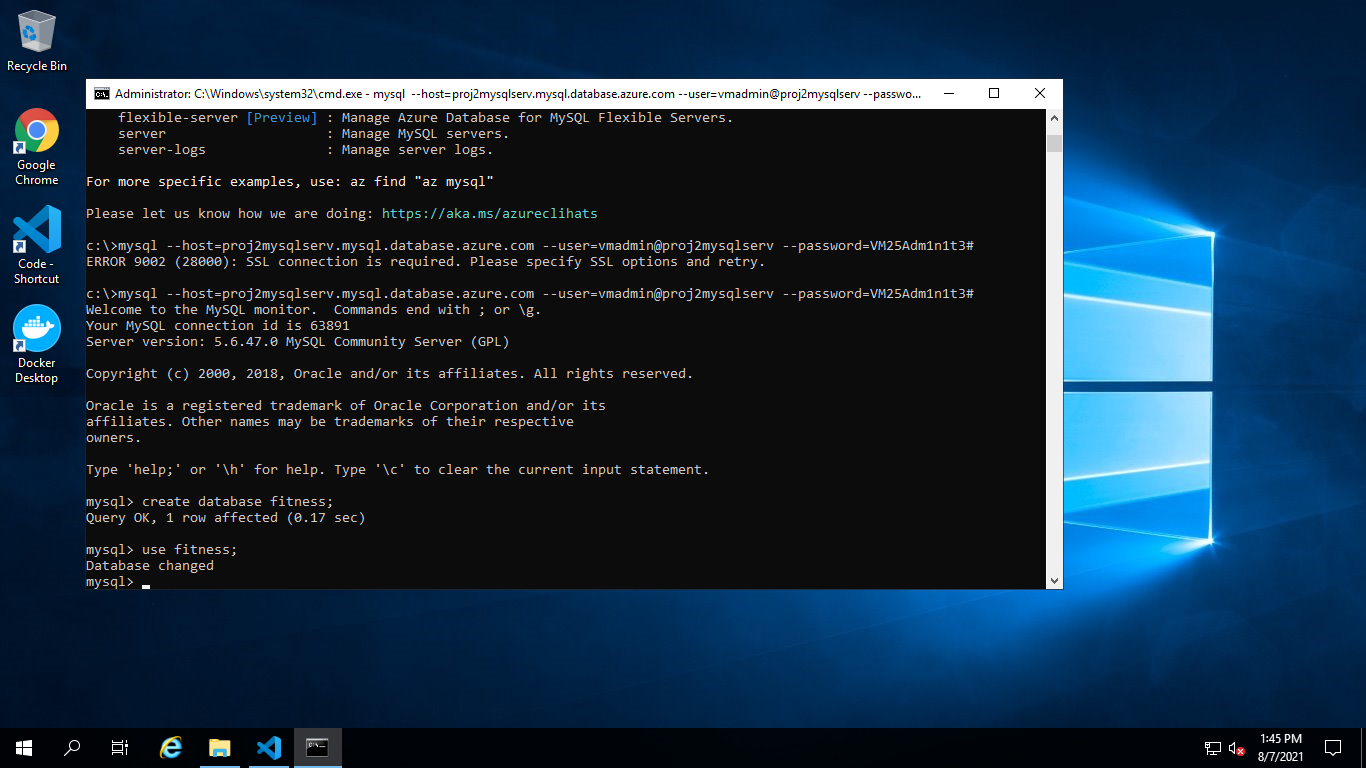
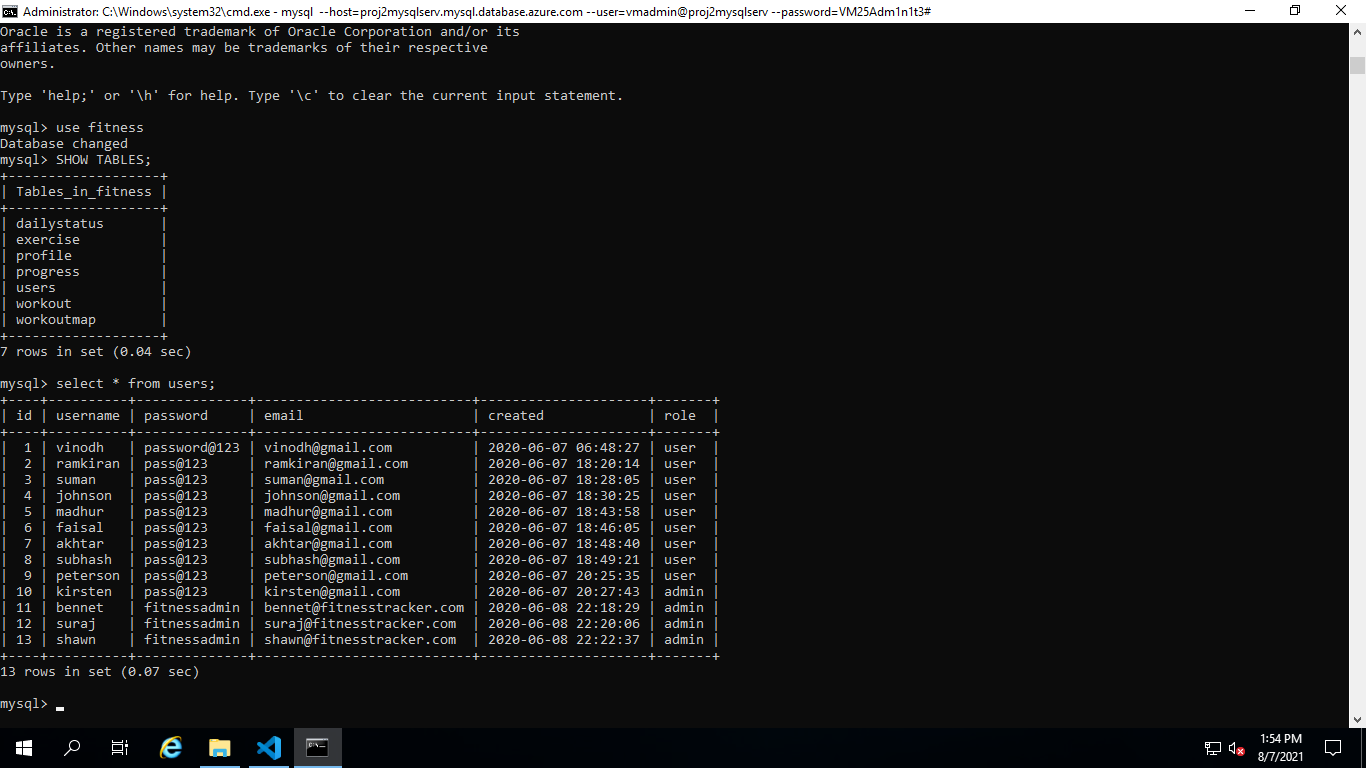
1. Created Azure ACR AND AKS via Terraform  
     
     
     
   Below is output:

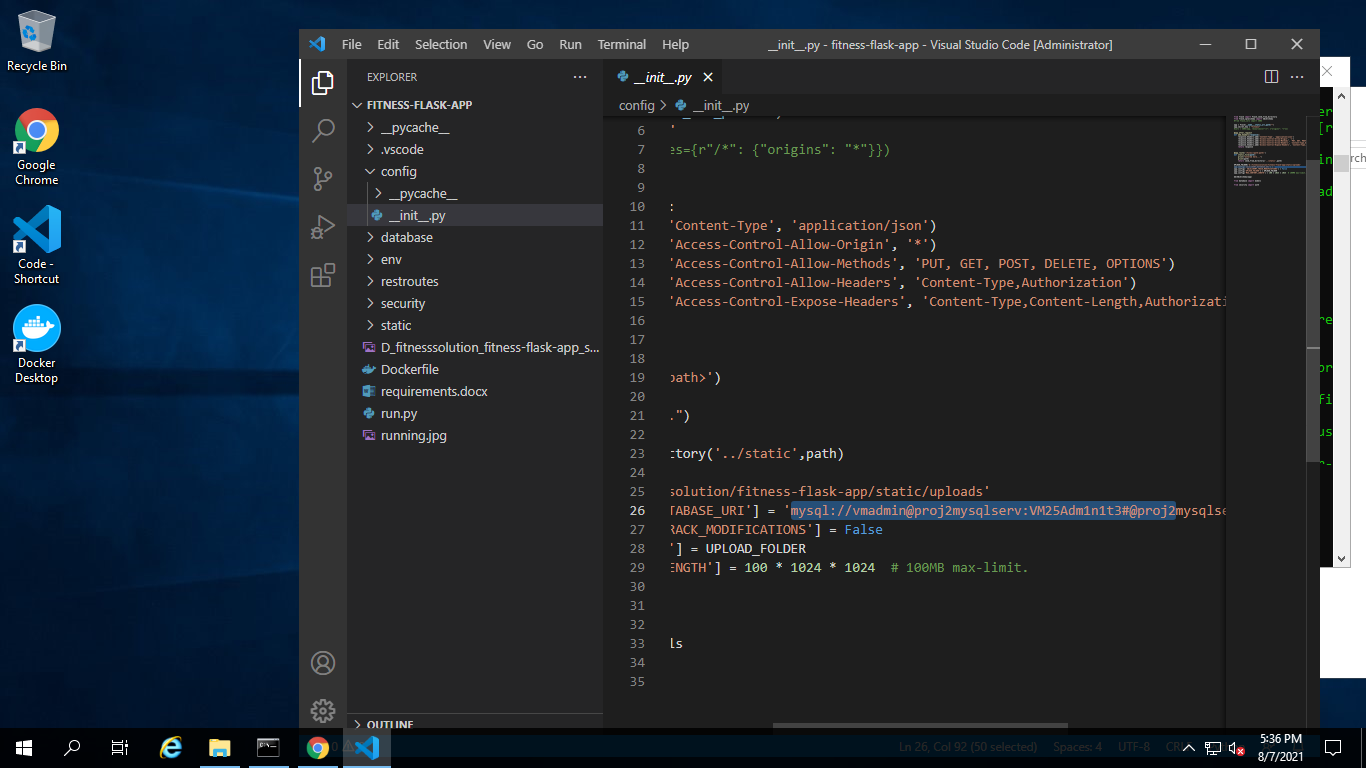


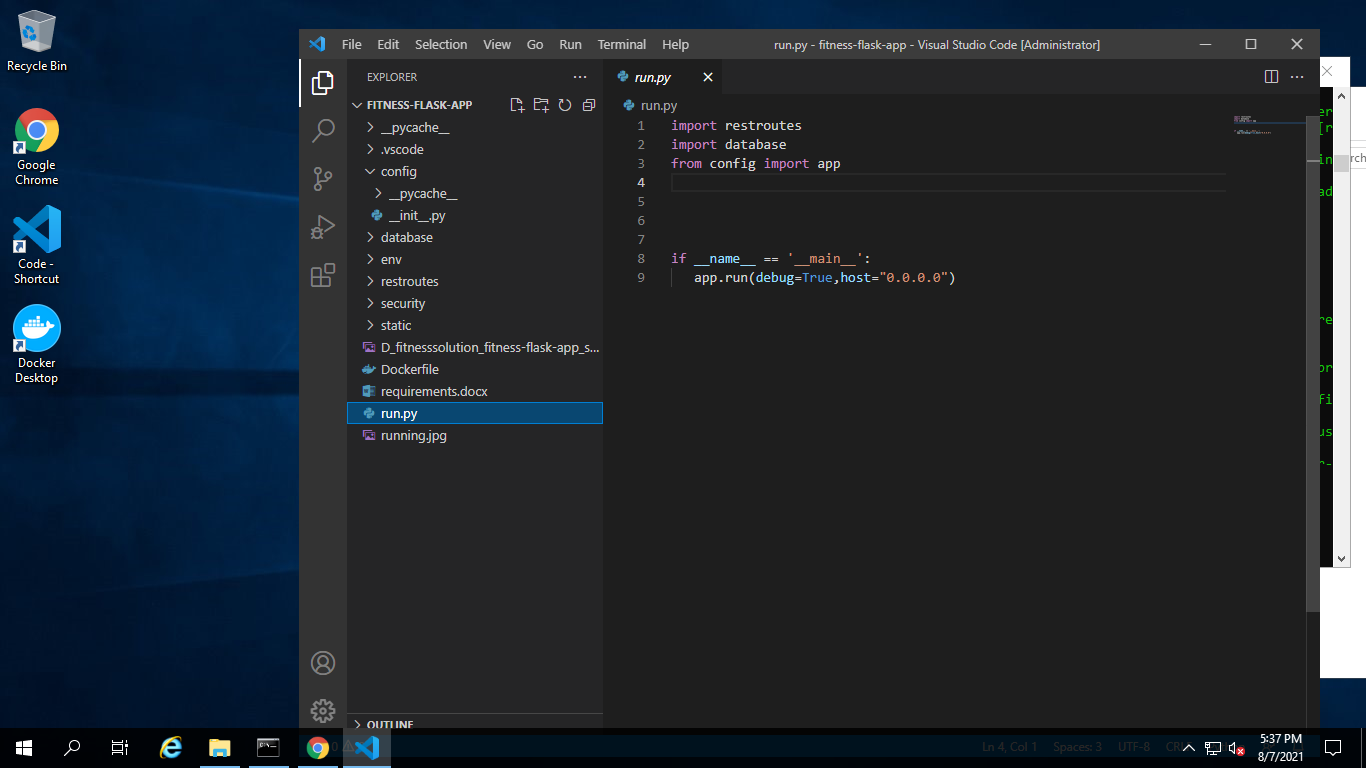
1. Created App service with Terraform  
     
     
     
   Below is output:



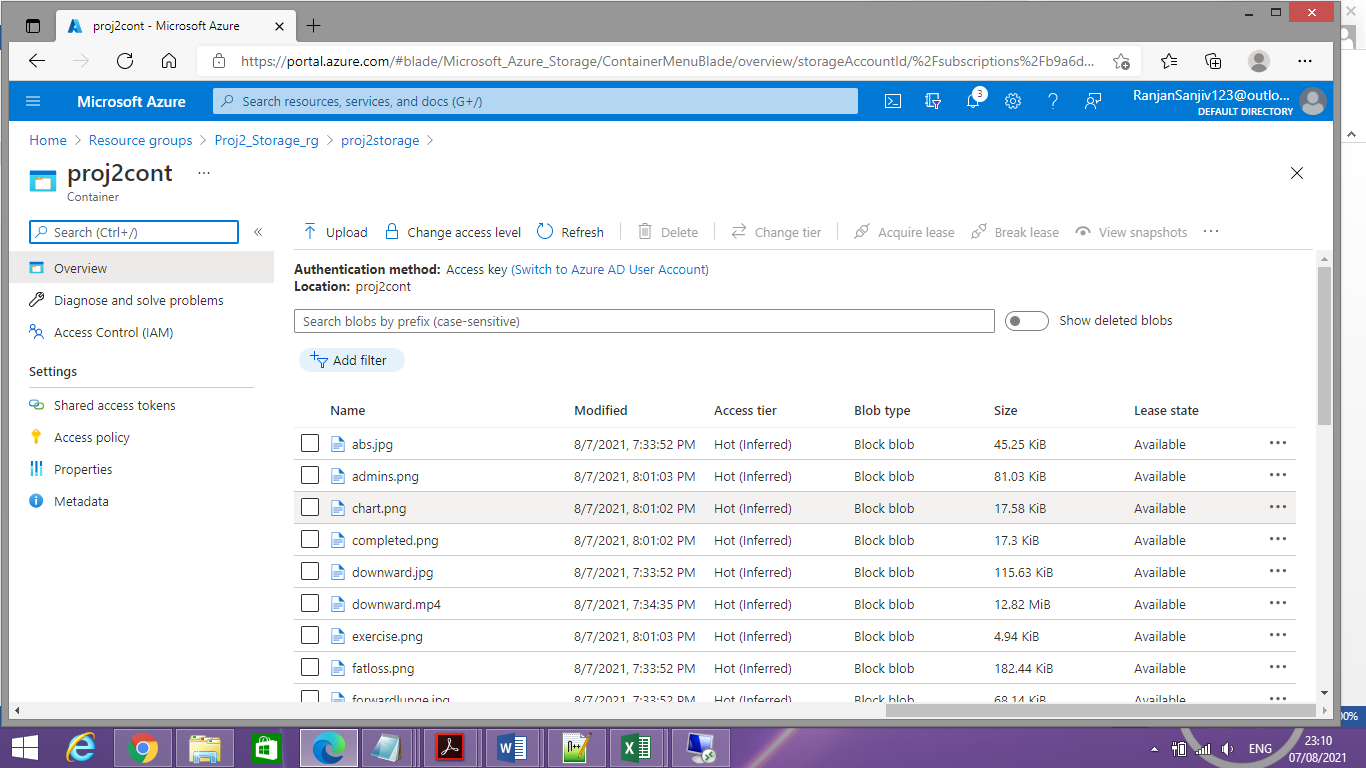
1. Azure Database Service for MYSQL with Data Restored

Created “Fitness” database:  
  
  
Imported data form fitnessbackup.sql:  


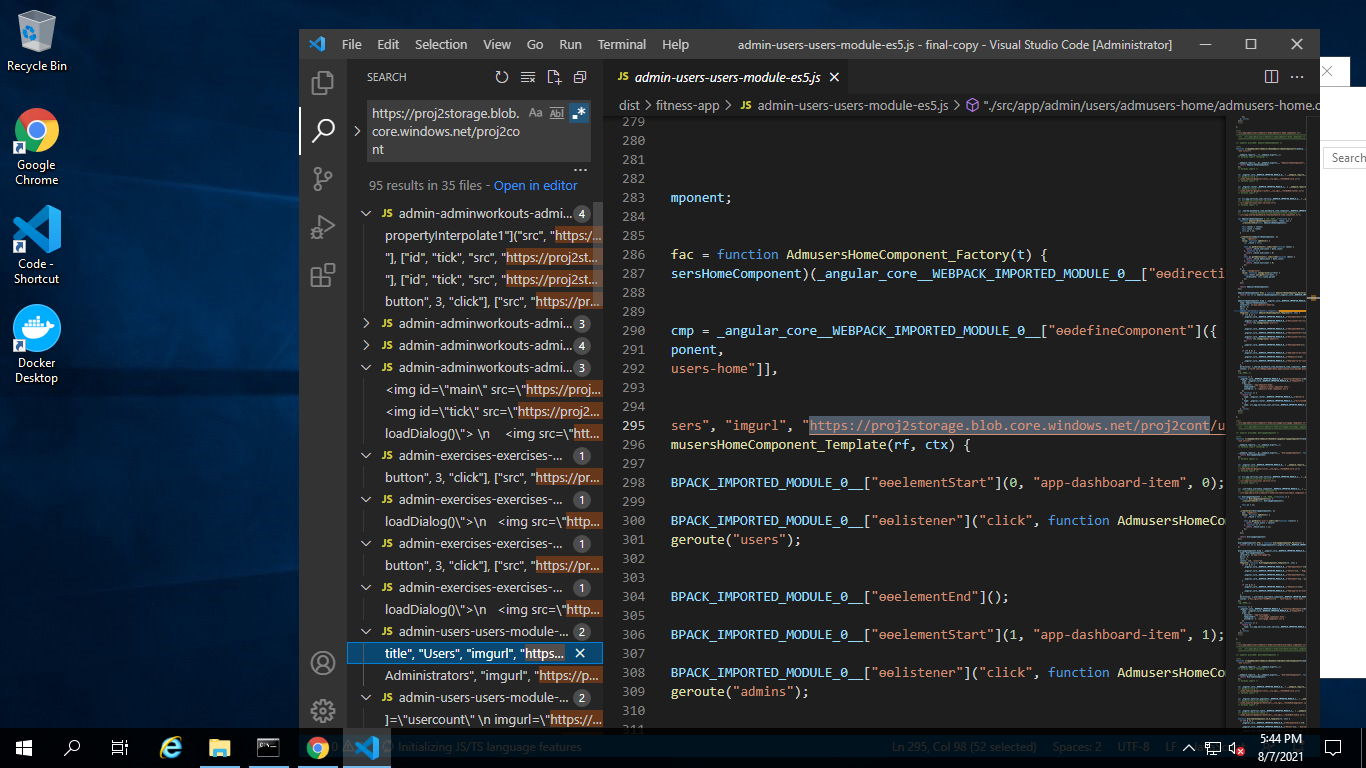
1. Changed DB Connection string on init.py and updated run.py with suggested changes  
     
   init.py:  
   

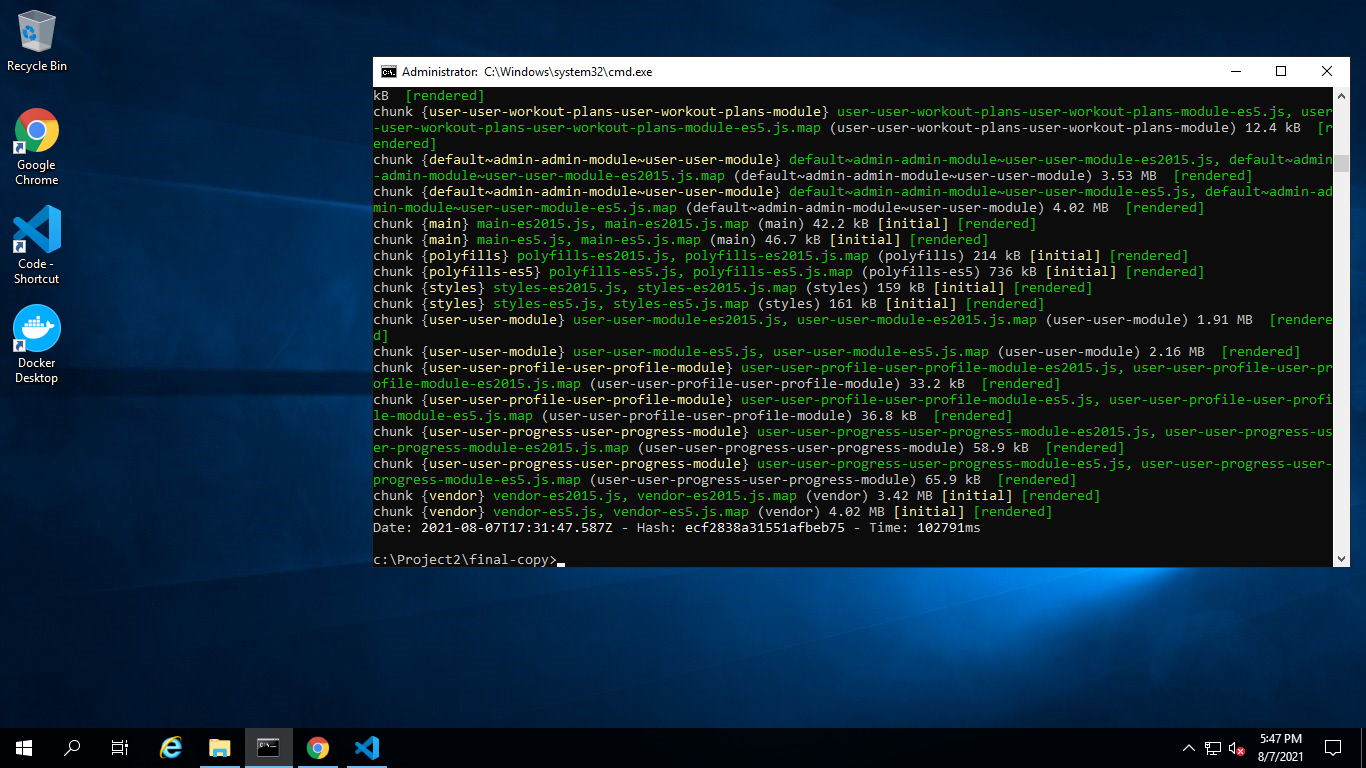
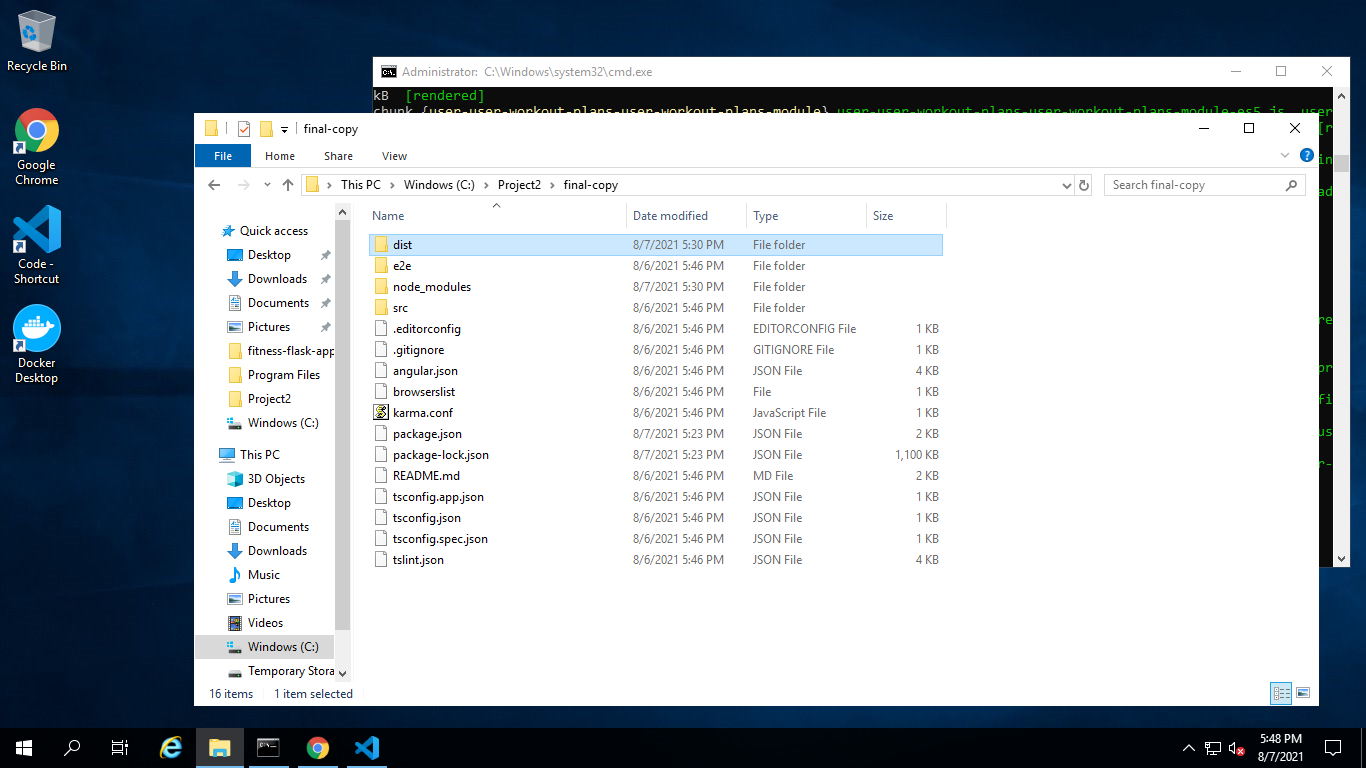
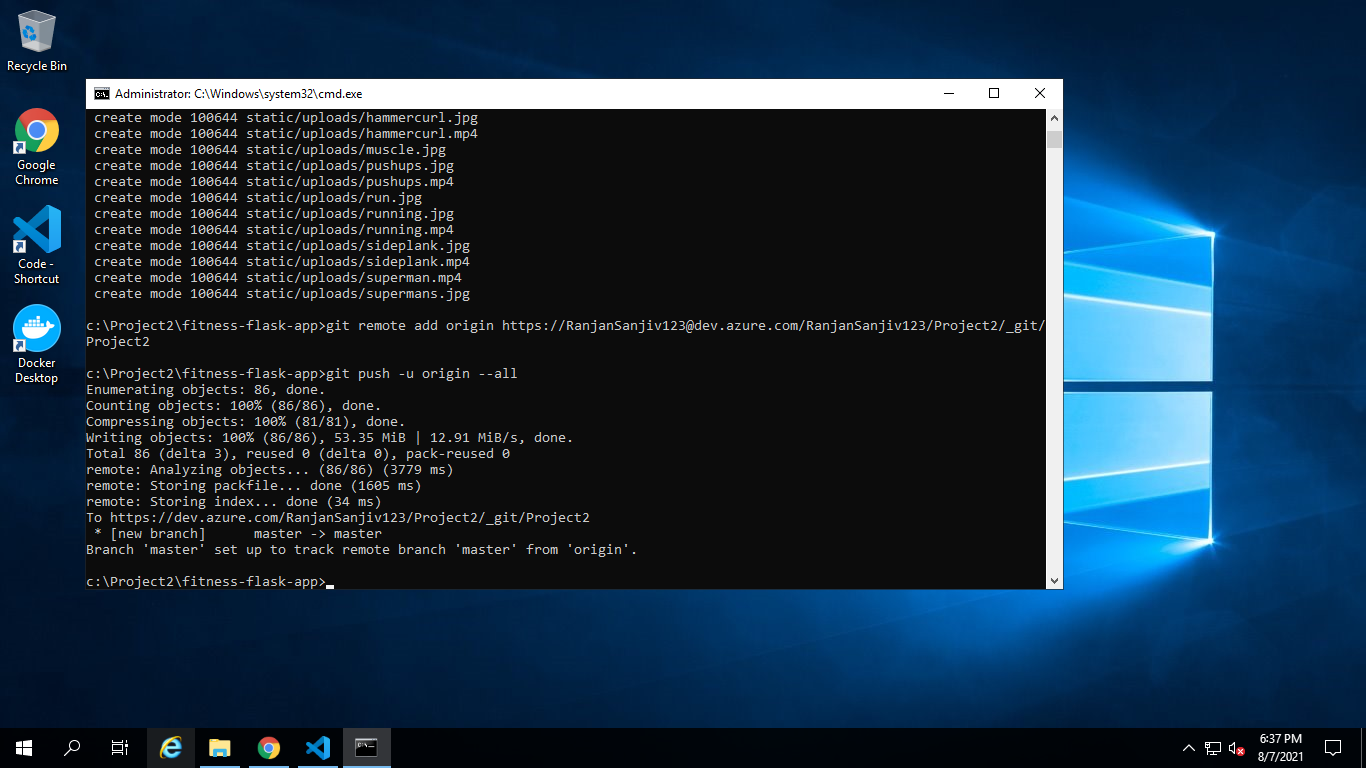
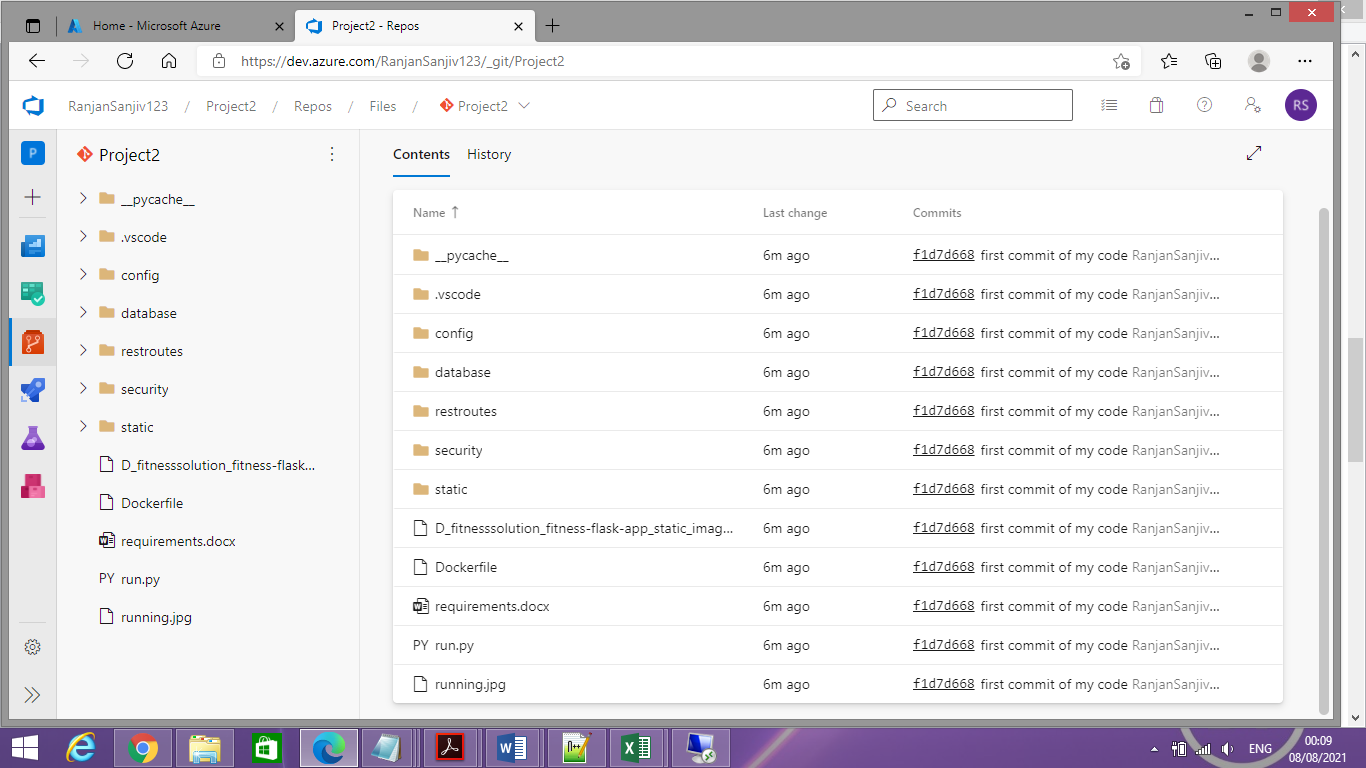
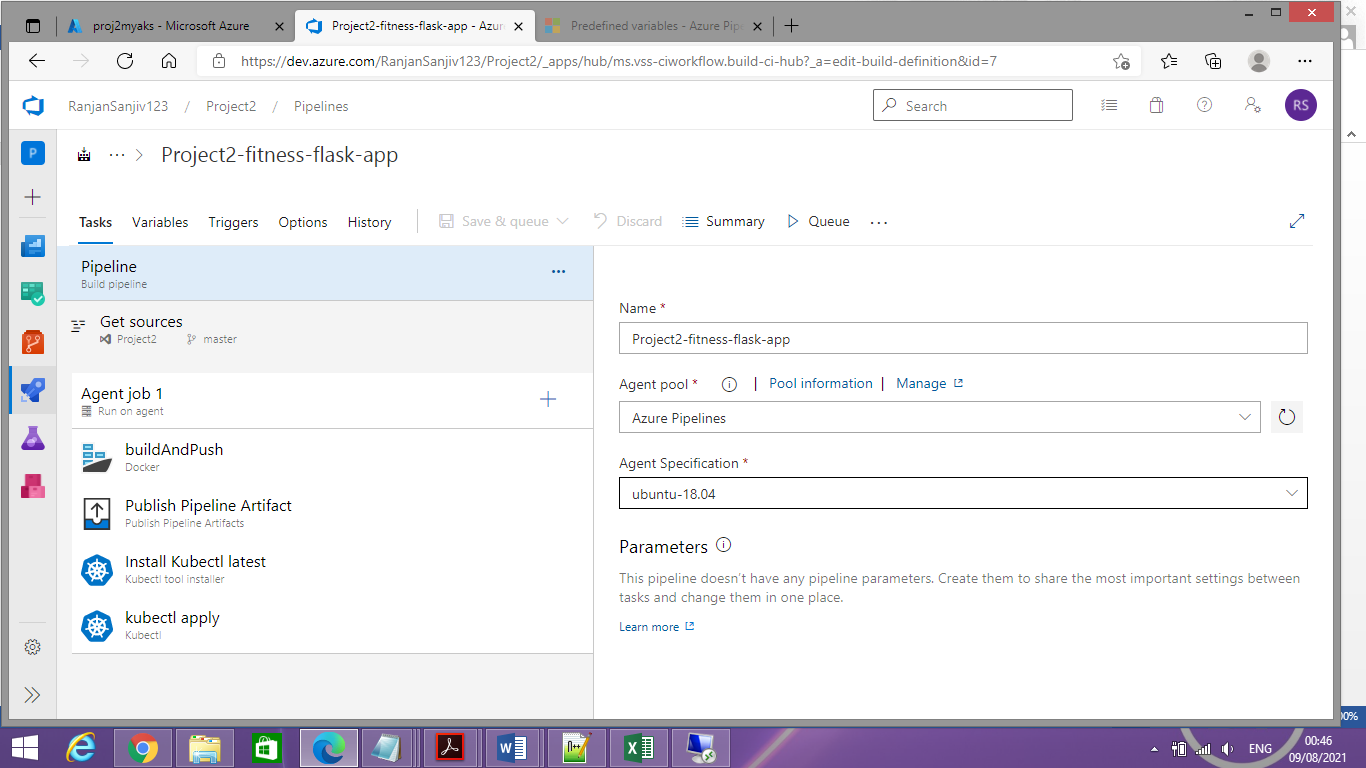
run.py:  


1. Uploaded all Static images on Azure storage container



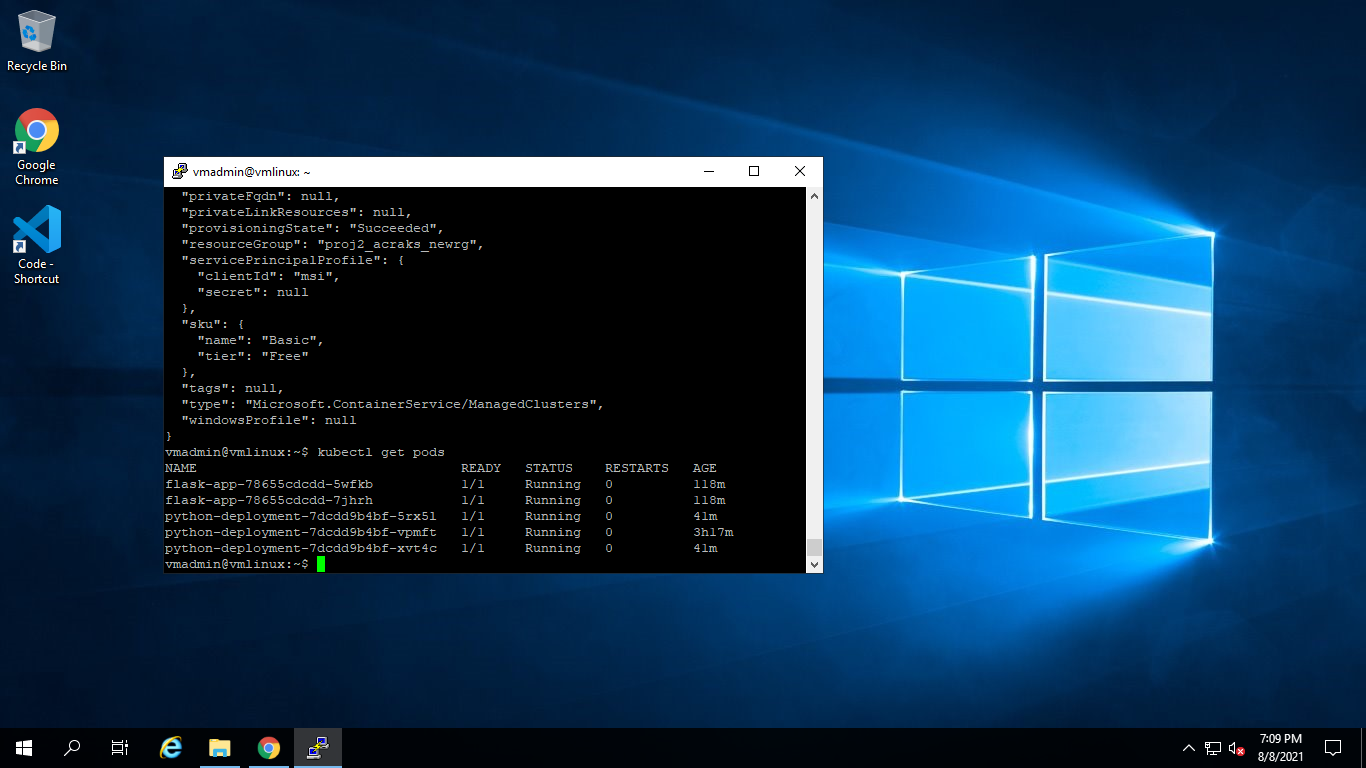
1. Then changed image URL - <http://localhost:5000/files/images> to Blob end point - <https://proj2storage.blob.core.windows.net/proj2cont> on final-copy folder of Angular App

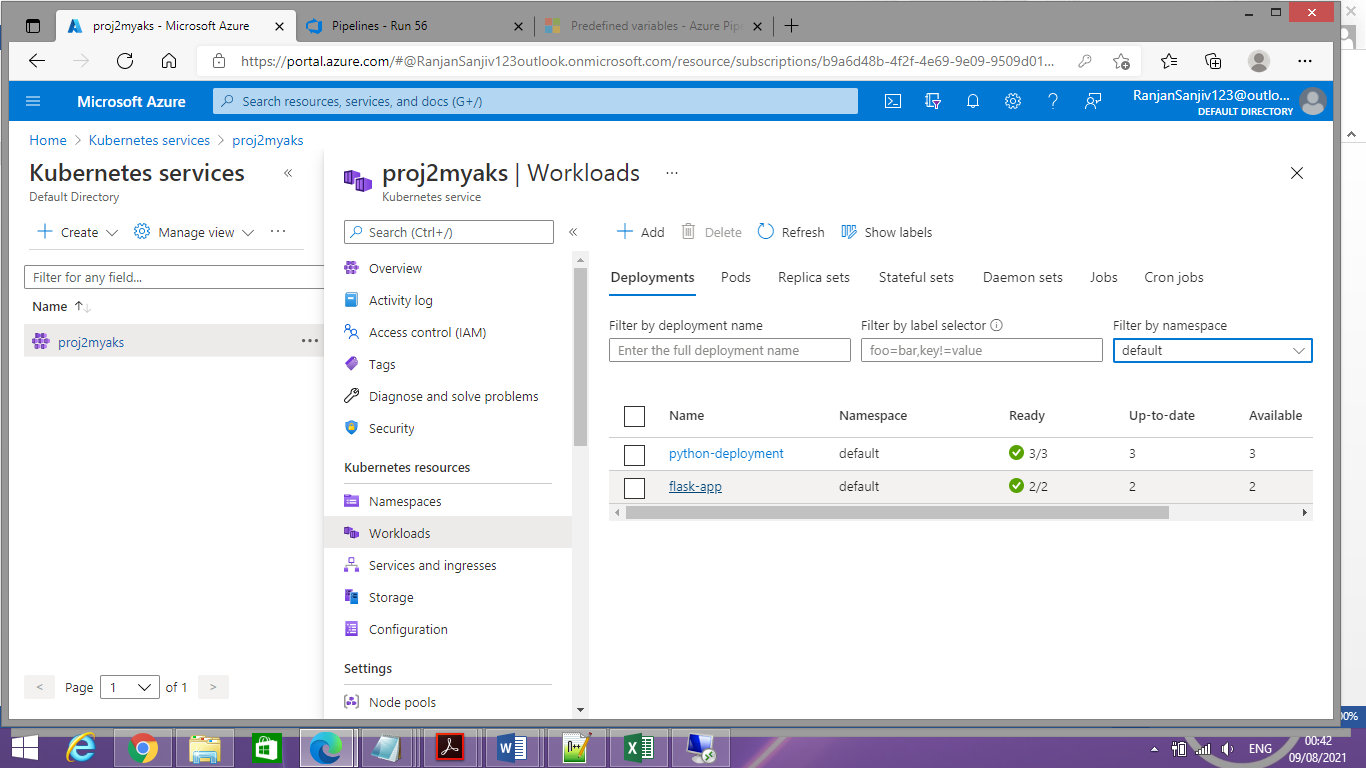


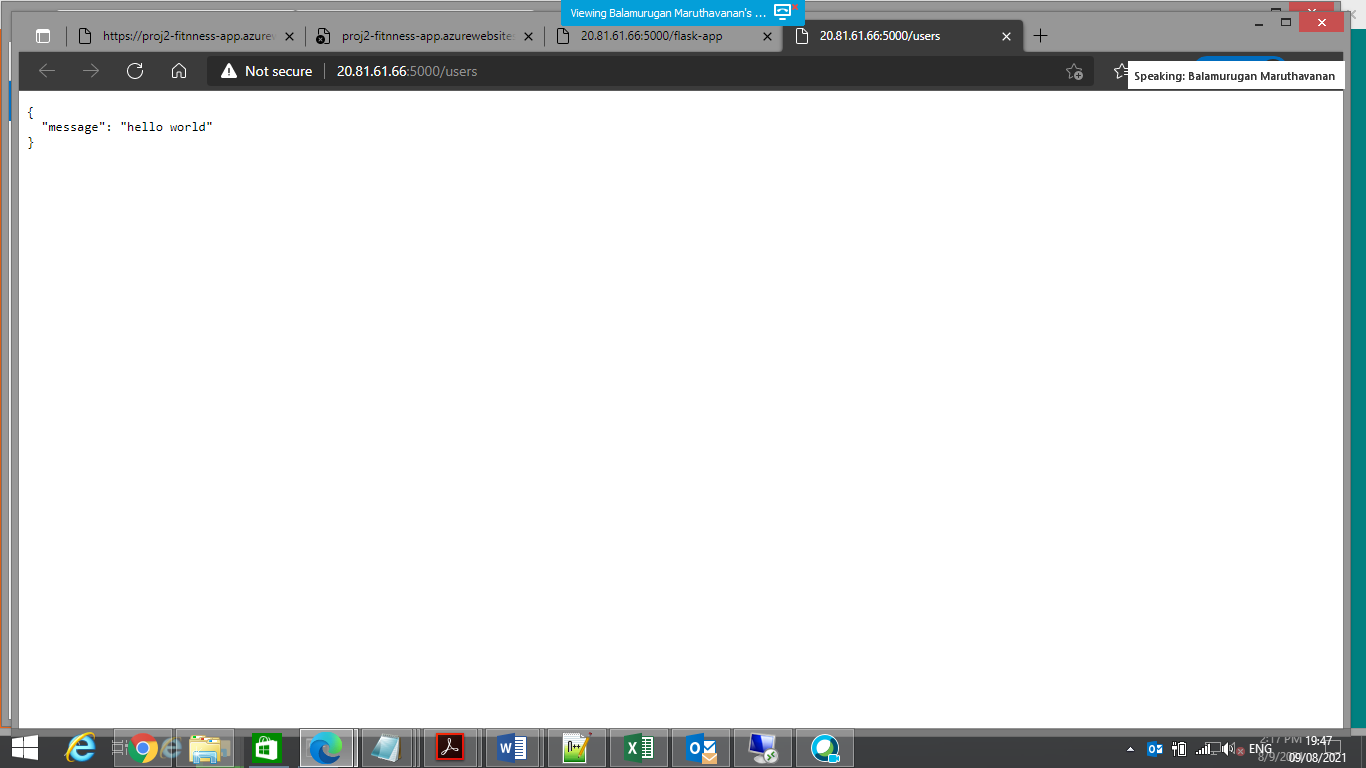
1. Then performed   
    - npm install -g @angular/cli  
   - ng build  
     
     
   And I can see <dist> folder inside final-copy folder:  
   
2. To Azure repo push the code downloaded from given code link  
     
   Using git, push the fitness-flask code to Azure repo:  
   
3. Verified Devops repo for the same:  
   
4. Created Pipeline 1 for AKS deployment:  
   

Received External Load balancer IP

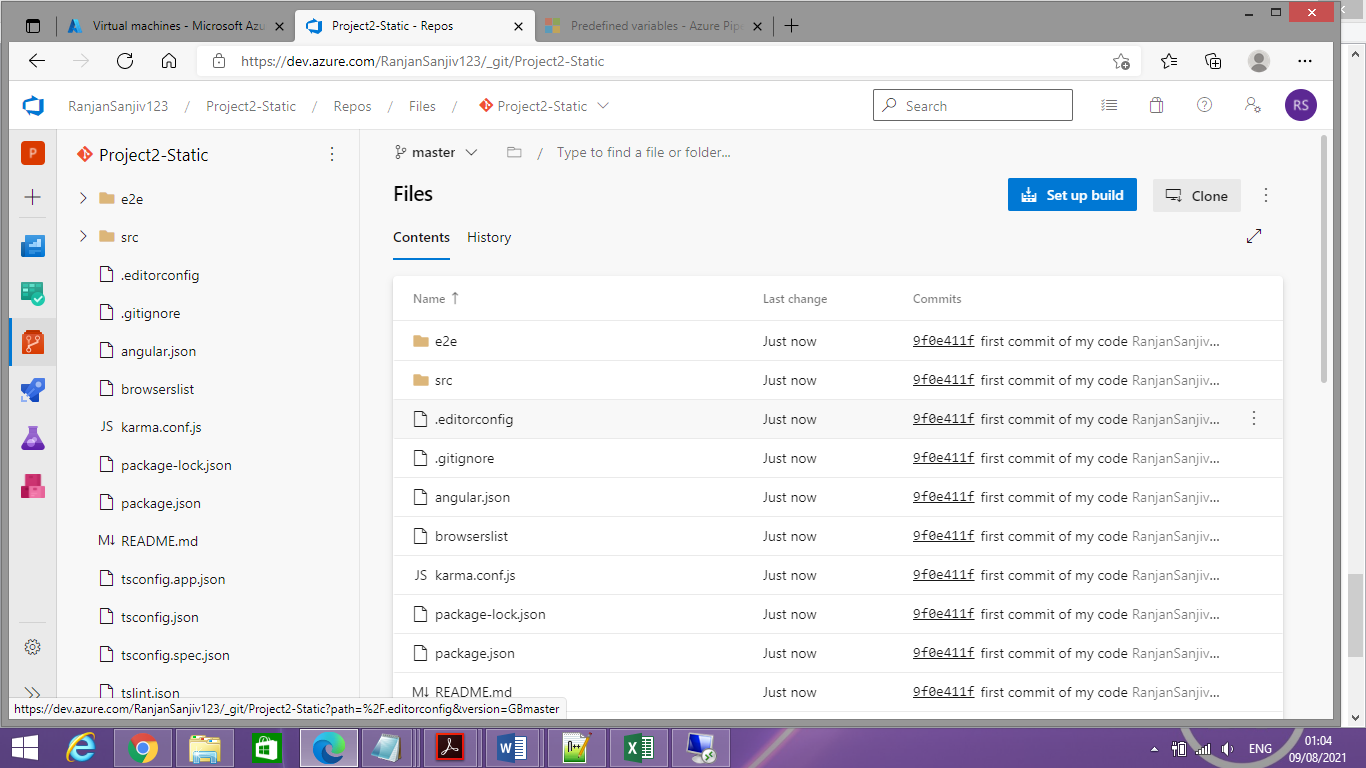
1. Successfully pulled images from ACR in AKS



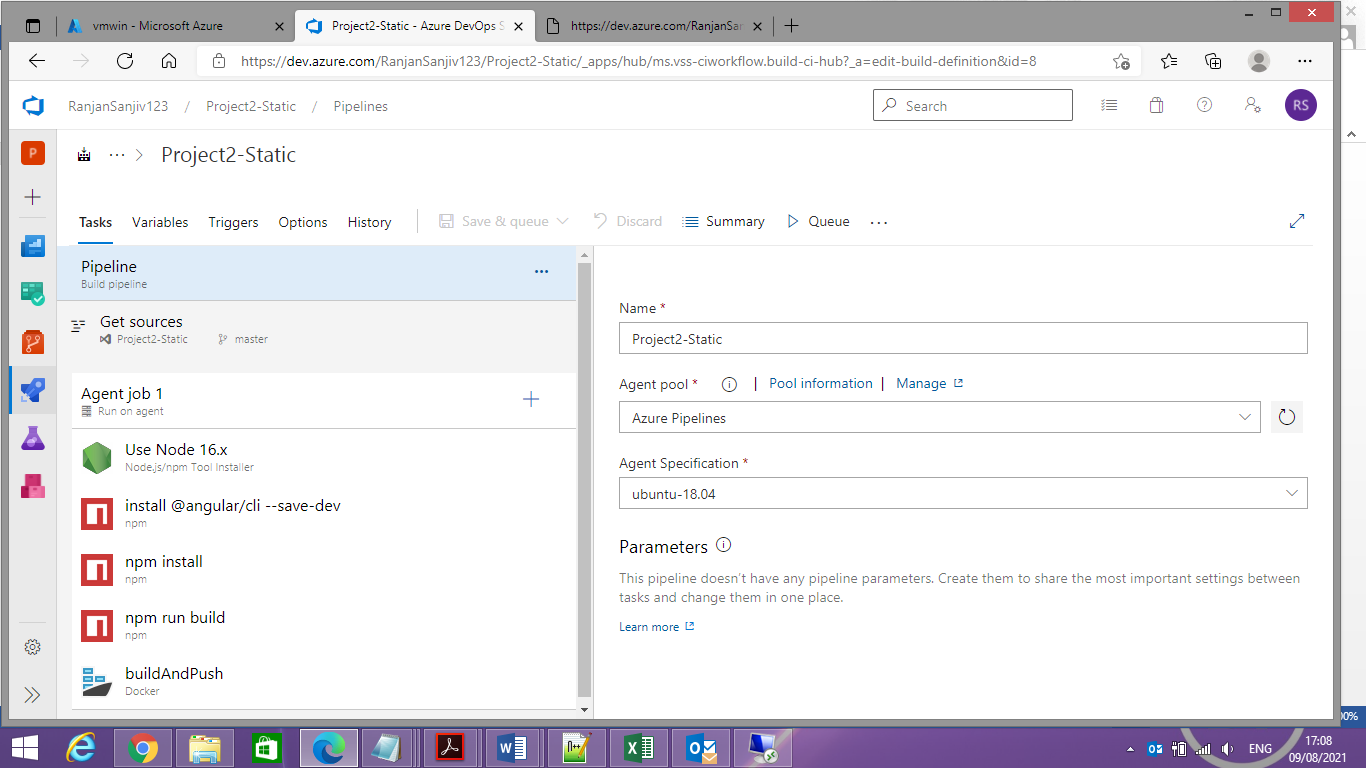
15. On Portal  


16. Access Page on External Load balancer at the port 5000:  


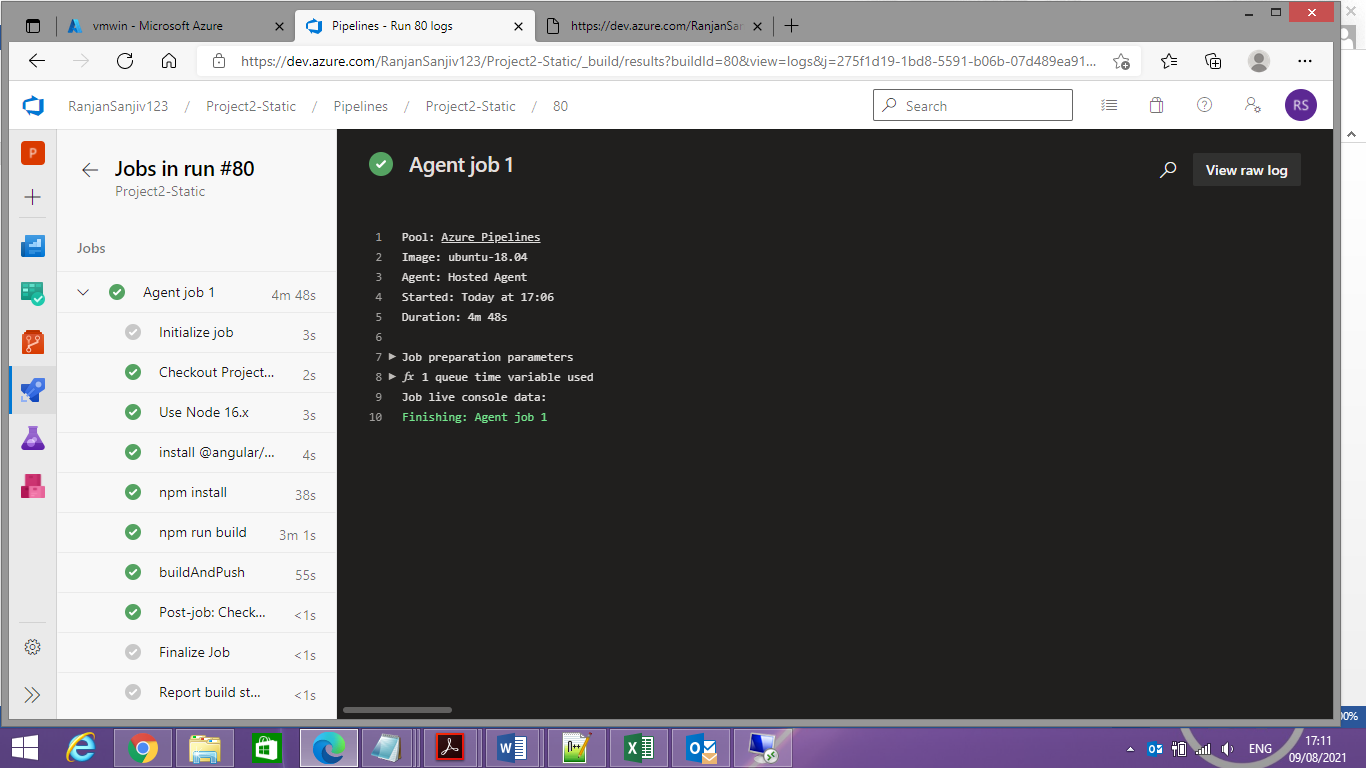
16. Created new repo for static page:



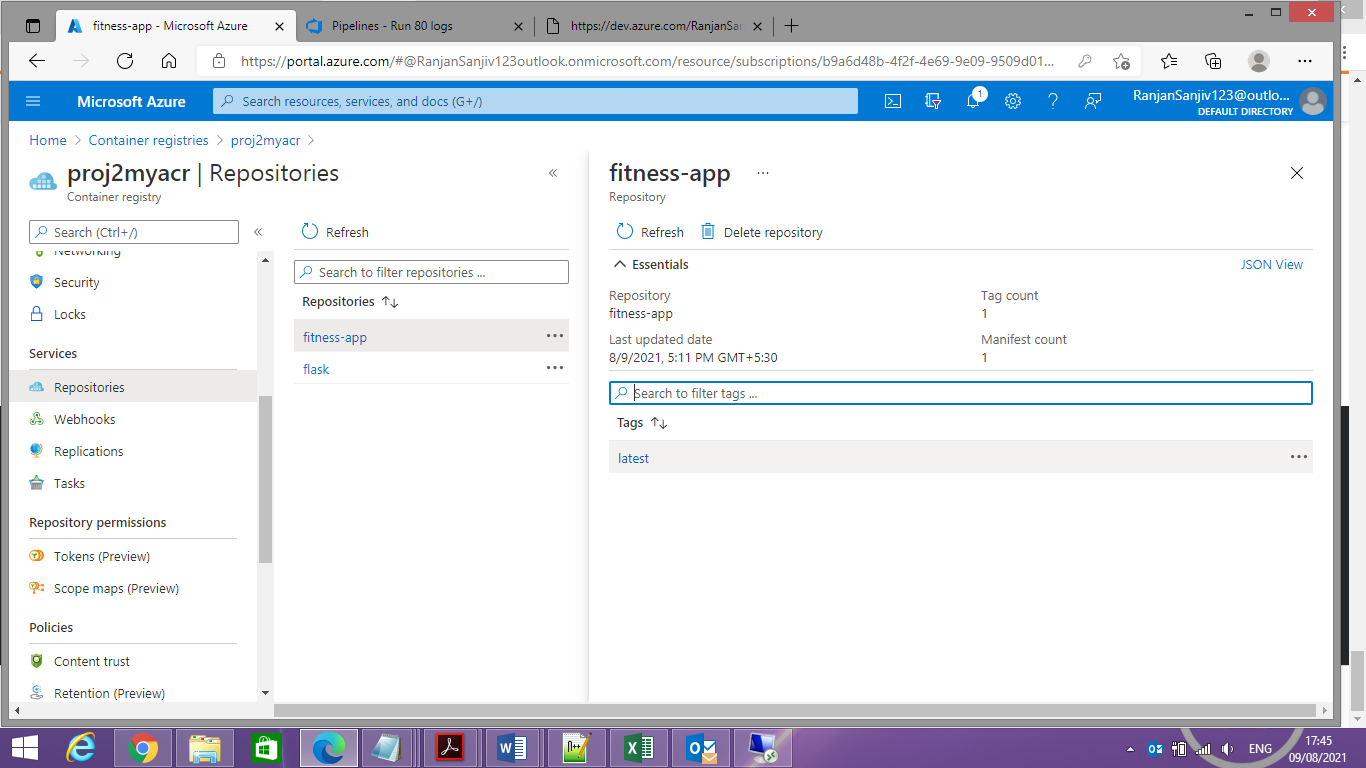
17. Created pipeline 2 for Static site



18. Pipeline executed successfully



18. Image got pushed successfully to ACR – fitness-app: latest -:



19: Now pushed this image Azure APP   
  
