

Bulut Bilişime Giriş Final Projesi - Başkenti Tahmin Et



Gerekli tahmini süre: 30 dakika

Bu final projesinde, "Başkenti Tahmin Et" uygulamasını bulutta dağıttıcasınız. Bu, size 4 seçenekten bir ülkenin başkentini tahmin etmenizi isteyen bir web uygulamasıdır.

Bir uygulamanın bulutta nasıl geliştirileceğini ve dağıtılamasını pratik yaparak öğrenmek için sağlanan kaynak kodunu ve adımları kullanacaksınız.

Hedefler:

1. Kaynak kodunu klonlayın
2. Docker imajı oluşturun
3. Docker üzerinde dağıtın
4. IBM Cloud'a imajı etiketleyin ve itiniz
5. IBM Code Engine üzerinde dağıtın

Arka Plan

Docker

Konteynerler, uygulamaları ve bunların bağımlılıklarını paketleyen izole ortamlardır. Her bir konteyner, ana işletim sisteminde izole bir işlem olarak çalışır.

[Docker](#), geliştiricilerin hafif, izole konteynerler içinde uygulamaların dağıtımını ve yönetimini otomatikleştirmesine olanak tanıyan açık kaynaklı bir platformdur.

IBM Cloud

[IBM Cloud](#), IBM tarafından sunulan bir bulut bilişim platformu ve bulut tabanlı hizmetler paketidir. Bulutta çeşitli uygulama ve iş yüklerinin geliştirilmesi, dağıtım ve yönetimini desteklemek için bir dizi altyapı, platform ve yazılım hizmeti sunar.

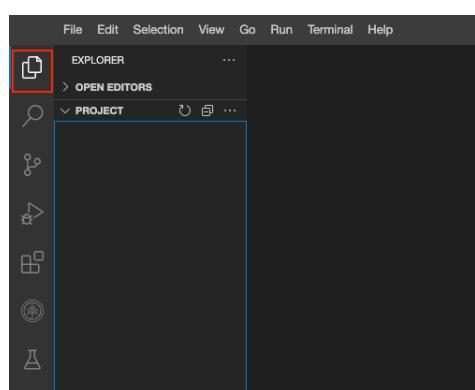
IBM Code Engine

[IBM Cloud Code Engine](#), IBM Cloud tarafından sağlanan sunucusuz bir hesaplama platformudur. Geliştiricilerin, temel altyapıyı yönetme ihtiyacı olmadan konteynerleştirilmiş uygulamaları dağıtmaması ve çalıştırmasına olanak tanır. Sunucu sağlama, ölçeklendirme ve bakımın karmaşıklıklarını soyutlayarak, geliştiricilerin kod yazmaya ve uygulama geliştirmeye odaklanmasına sağlar.

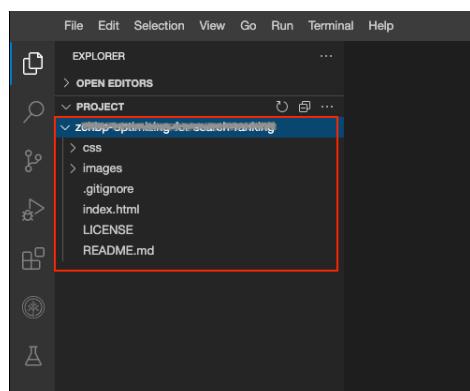
Cloud IDE'de dosyalarla çalışma

Cloud IDE'ye yenişeniz, bu bölüm size Cloud IDE'de projenizin bir parçası olan dosyaları nasıl oluşturup düzenleyeceğini gösterecektir.

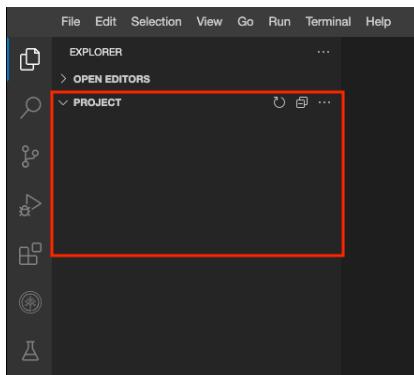
Cloud IDE içinde dosyalarınızı ve dizinlerinizi görüntülemek için bu dosya simgesine tıklayarak açabilirsiniz.



Eğer başlangıç kodunu git clone komutıyla klonladığınız, aşağıdaki gibi görünecektir:

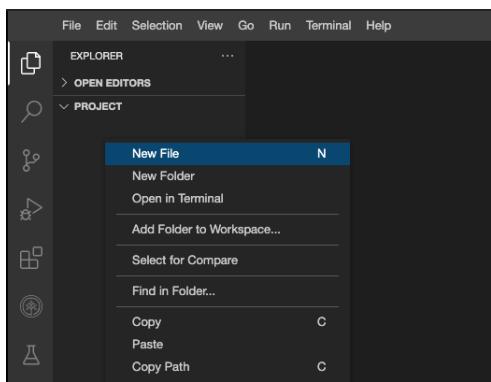


Aksi takdirde, boş bir proje şöyle görünür:



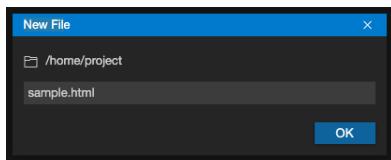
Yeni bir dosya oluşturma

Projenizde bir dosya oluşturmak için sağ tıklayıp Yeni Dosya seçeneğini seçebilirsiniz.

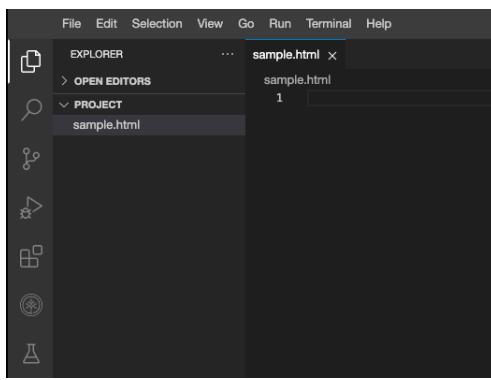


Aynı işlemi Dosya -> Yeni Dosya seçeneğiyle de yapabilirsiniz.

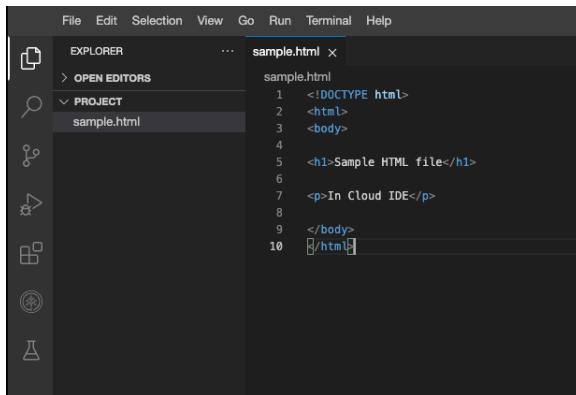
Sonrasında, bu yeni dosyanın adını girmeniz istenecek. Aşağıdaki örnekte, sample.html oluşturuyoruz.



Dizin yapısında sample.html dosya adına tıklamak, dosyayı sağ panelde açacaktır. Tüm farklı dosya türlerini oluşturabilirsiniz; örneğin JavaScript dosyası için FILE_NAME.js.

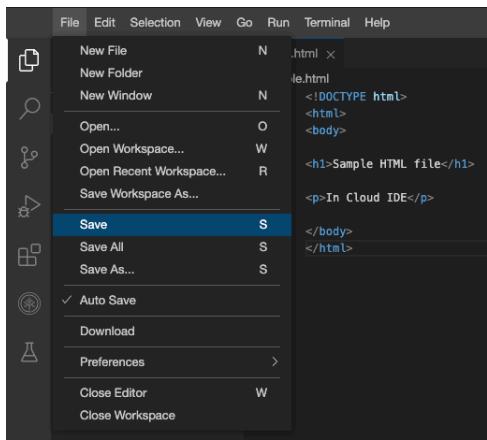


Örnekte, bazı temel html kodlarını yapıştırdık ve ardından dosyayı kaydettik.



Ve bunu şu şekilde kaydettik:

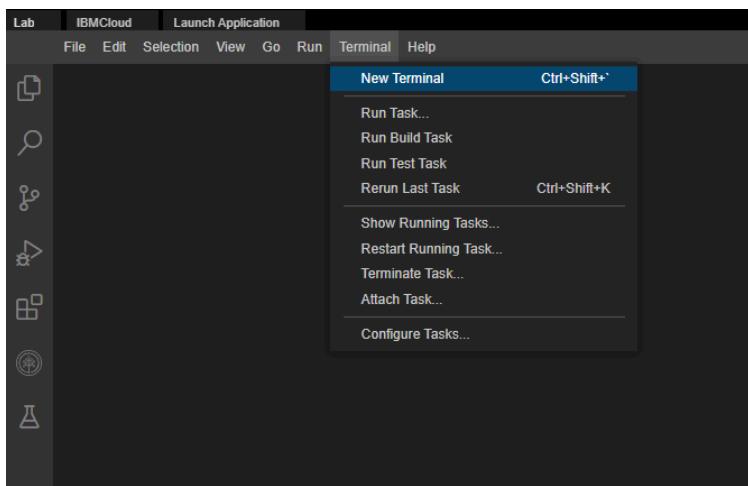
- Menüden giderek.
- Mac'te ⌘ + S veya Windows'ta CTRL + S tuşlarına basarak.
- Ya da otomatik olarak kaydedebilir.



Ortamı ve komut satırı araçlarını doğrulayın

1. Editördeki menütü kullanarak bir terminal penceresi açın: Terminal > Yeni Terminal.

Not: Eğer terminal zaten açıksa, lütfen bu adımı atlayın.



2. docker CLI'nin kurulu olduğunu doğrulayın.

```
docker --version
```

Aşağıdaki çıktıyı göremelisiniz, ancak sürüm farklı olabilir:

```
theia@theiadocker-:~/home/project$ docker --version
Docker version 20.10.7, build build 20.10.7-0ubuntu5~18.04.3
```

3. ibmcloud CLI'nin kurulu olduğunu doğrulayın.

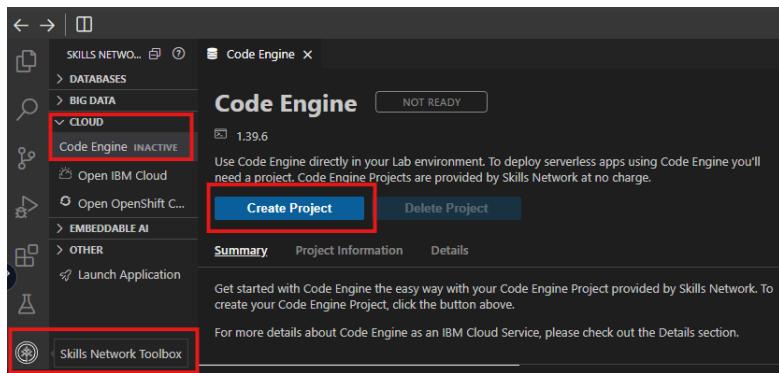
```
ibmcloud version
```

Aşağıdaki çıktıyı göremelisiniz, ancak sürüm farklı olabilir:

```
theia@theiadocker-:~/home/project$ ibmcloud version
ibmcloud version 2.1.1+19d7e02-2021-09-24T15:16:38+00:00
```

Kod Motorunu Başlat

1. Laboratuvar ortamınızda menüde, SN logosuna tıklayın, ardından Cloud açılır menüsüne tıklayın ve Code Engine seçeneğini seçin. Kod motoru kurulum paneli görünür. Başlamak için Create Project butonuna tıklayın.



2. Kod motoru ortamının hazırlanması biraz zaman alır. Kurulum panelinde ilerleme durumu gösterilecektir.

3. Kod motoru kurulumu tamamlandığında, aktif olduğunu göreceksiniz. Aşağıda gösterildiği gibi terminalde önceden yapılandırılmış CLI'yi başlatmak için Code Engine CLI butonuna tıklayın.

4. Önceden yapılandırılmış CLI'nin başlatıldığı ve ana dizinin mevcut dizin olarak ayarlandığını gözlemleyecesiniz. Ön yapılandırma kapsamında proje ayarlandı ve Kubeconfig ayarlandı. Ayrintılar terminalde aşağıdaki gibi gösterilmektedir.

The screenshot shows the Skills Network Toolbox interface with the 'Code Engine' project selected. The main panel displays the 'Code Engine' project details, including its version (1.39.6) and a note about deploying serverless apps. A red box highlights the 'Code Engine CLI' button. Below it, a terminal window shows the command 'ibmcloud ce project current' being run, followed by the project context information and a configuration file path.

SKILLS NETWORK TOOLBOX ?

> DATABASES

> BIG DATA

< CLOUD

Code Engine **ACTIVE**

Open IBM Cloud

> EMBEDDABLE AI

> OTHER

Launch Application

Code Engine x

Code Engine READY TO USE

1.39.6

Use Code Engine directly in your Lab environment. To deploy serverless app
Code Engine Projects are provided by Skills Network at no charge.

Delete Project

Summary Project Information Details

Your Skills Network Code Engine Project is now ready to use. You can now create Applications.

For important information about your project view the Project Information section or an IBM Cloud Service, please check out the Details section.

In order to interact with Code Engine please click the following button:

Code Engine CLI

Problems theia@theiadocker-captainfedor1: /home/project theia@th

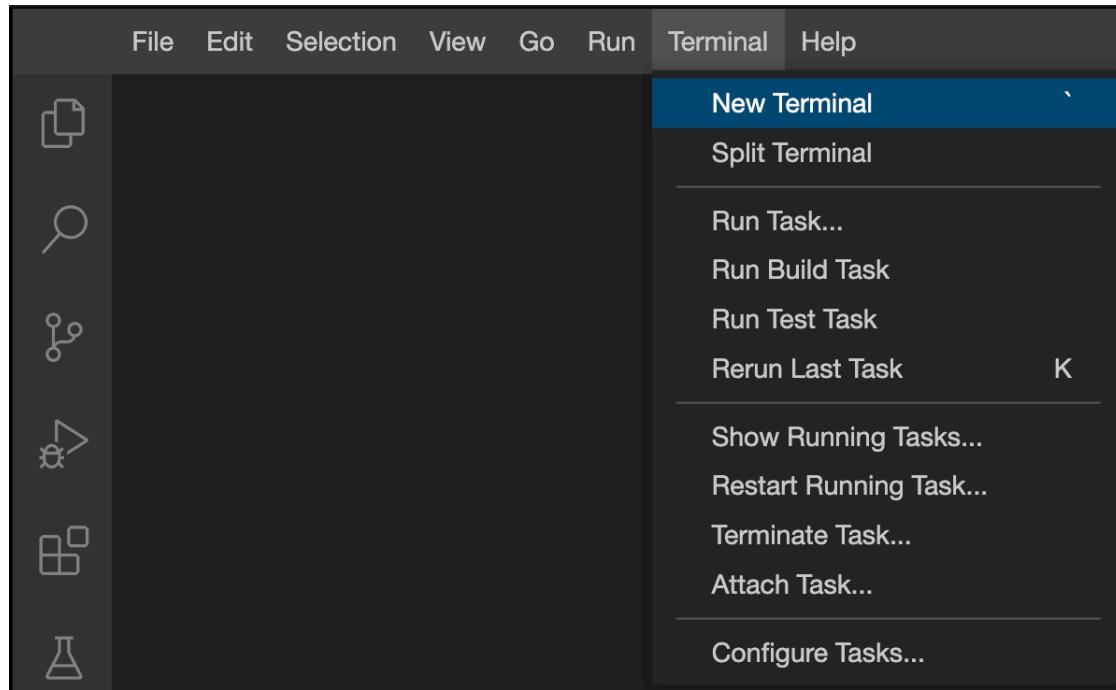
```
ibmcloud ce project current
theia@theiadocker-captainfedor1:/home/project$ ibmcloud ce project current
Getting the current project context...
OK

Name:      Code Engine - sn-labs-captainfedor1
ID:       9c079722-5f80-4056-bed6-798cdb0acf04
Subdomain: ywj8nhvp9f9
Domain:   us-south.codeengine.appdomain.cloud
Region:   us-south

Kubernetes Config:
Context:      ywj8nhvp9f9
Environment Variable: export KUBECONFIG="/home/theia/.bluemix/plugins/container-tools/n-labs-captainfedor1-9c079722-5f80-4056-bed6-798cdb0acf04.yaml"
theia@theiadocker-captainfedor1:/home/project$
```

Kurulum : Uygulama oluştur

1. Editördeki menütü kullanarak bir terminal penceresi açın: **Terminal > Yeni Terminal**.



2. Eğer şu anda proje klasöründe değilseniz, proje klasörünüze geçmek için aşağıdaki kodu kopyalayıp yapıştırın.

```
cd /home/project
```

3. Bu proje için gerekli başlangıç kodunu içeren Git deposunu klonlamak için aşağıdaki komutu çalıştırın, eğer Git deposu henüz mevcut değilse.

```
[ ! -d 'fyidw-guess-the-capital' ] && git clone https://github.com/ibm-developer-skills-network/fyidw-guess-the-capital.git
```

4. Laboratuvara başlamak için **fyidw-guess-the-capital** dizinine geçin.

```
cd fyidw-guess-the-capital
```

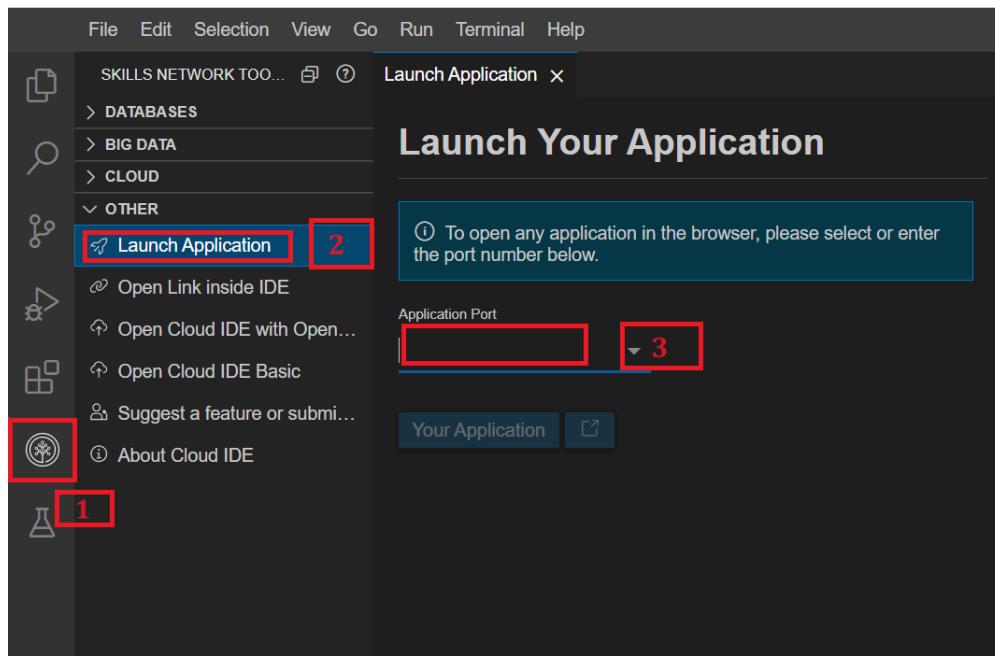
5. Bu dizinin içeriğini listeleyin, böylece bu laboratuvar için artefaktları görebilirsiniz.

```
ls
```

6. Web sayfanızı barındırmak için terminalde aşağıdaki komutu çalıştırın.

```
python3 -m http.server
```

7. Uygulamanızı tarayıcınızda test etmek için önce uygulamayı çalıştırın. Uygulamanızı görüntülemek için sol paneldeki Skills Network simgesine tıklayın (numara 1'e bakın). Bu işlem **SKILLS NETWORK TOOLBOX**'u açacaktır. Ardından, **Launch Application**'a tıklayın (numara 2'ye bakın). **Application Port** bölümüne **8000** port numarasını girin (numara 3'e bakın) ve butonuna tıklayın. Ayrıca uygulamanızı başlatmak için aşağıdaki butona da tıklayabilirsiniz.



Launch Application

8. Böyle görünecek:

What is the capital of Lebanon?

- Beirut
- Panama City
- Djibouti
- Moroni

Next

9. Terminalinizde, web sunucusunu durdurmak için **CTRL + C** tuşlarına basın.

Görev 1: Uygulamayı Konteynerleştirme

Uygulamamızı modernleştirmeye başlayalım. Bunun ilk adımı, Docker kullanarak konteynerleştirmektir.

Dockerfile Oluştur

Görevlerin:

- Aşağıdaki içeriği yapıştır

[Open Dockerfile in IDE](#)

Aşağıdakini Dockerfile içeriği olarak kullan.

```
# Use the official Nginx base image from Docker Hub
FROM nginx
# Copy the favicon to the default Nginx HTML directory
COPY favicon.ico /usr/share/nginx/html/favicon.ico
# Copy the main HTML file to serve as the landing page
COPY index.html /usr/share/nginx/html/index.html
# Copy the JavaScript file to enable client-side functionality
COPY script.js /usr/share/nginx/html/script.js
# Copy the CSS file to define the visual presentation and styling
COPY style.css /usr/share/nginx/html/style.css
# Copy the JSON data file to serve structured content to the frontend
COPY data.json /usr/share/nginx/html/data.json
```

Aşağıdaki gibi görünmelidir:

Dockerfile M X

```
fyidw-guess-the-capital > ⚡ Dockerfile
1 # Use the official Nginx base image from Docker Hub
2 FROM nginx
3
4 # Copy the favicon to the default Nginx HTML directory
5 COPY favicon.ico /usr/share/nginx/html/favicon.ico
6
7 # Copy the main HTML file to serve as the landing page
8 COPY index.html /usr/share/nginx/html/index.html
9
10 # Copy the JavaScript file to enable client-side functionality
11 COPY script.js /usr/share/nginx/html/script.js
12
13 # Copy the CSS file to define the visual presentation and styling
14 COPY style.css /usr/share/nginx/html/style.css
15
16 # Copy the JSON data file to serve structured content to the frontend
17 COPY data.json /usr/share/nginx/html/data.json
18 |
```

2. Bir Dockerfile'dan bir imaj oluşturun

```
docker build -t guess-the-capital .
```

```
theia@theiadocker-:~/home/project/fyidw-guess-the-capital$ docker build -t guess-the-capital .
[+] Building 12.2s (12/12) FINISHED
=> [internal] load build definition from Dockerfile 0.1s
=> => transferring dockerfile: 291B 0.0s
=> [internal] load .dockerignore 0.0s
=> => transferring context: 2B 0.0s
=> [internal] load metadata for docker.io/library/nginx:latest 0.3s
=> [auth] library/nginx:pull token for registry-1.docker.io 0.0s
=> [1/6] FROM docker.io/library/nginx@sha256:67f9a4f10d147a6e04629340e6493c970 10.3s
=> => resolve docker.io/library/nginx@sha256:67f9a4f10d147a6e04629340e6493c9703 0.0s
=> => sha256:262696647b70a57f5f7dbf97a91091e7b51c1d2537dff72a 41.46MB / 41.46MB 3.9s
=> => sha256:67f9a4f10d147a6e04629340e6493c9703300ca23a2f7f3aa5 1.86kB / 1.86kB 0.0s
=> => sha256:73e957703f1266530db0aeac1fd6a3f87c1e59943f4c13eb34 1.78kB / 1.78kB 0.0s
=> => sha256:648e0aadf75ac2ef63c5390adc6dc14fde37a5ad88c2870e 29.12MB / 29.12MB 2.0s
=> => sha256:89da1fb6dc964dd35c3f41b7b93ffc35eaf20bc61f2e1335f 8.15kB / 8.15kB 0.0s
=> => sha256:e66d0270d23f3038e0e8c94ee9244950fbfdb582476f61736b3c28 625B / 625B 0.1s
=> => sha256:55ac49bd649c325395133ae4f3640a07e28d9a25c4a56eb8ac3df9 957B / 957B 0.2s
=> => sha256:cbf42f5a00d268edb1684b8eb9039543669fc5f5d0aa801a01d346 366B / 366B 0.2s
=> => sha256:8015f365966bfa259003c319a44df5bb9290d279ca775b4f24 1.21kB / 1.21kB 0.3s
=> => sha256:4cadff8bc2aa83b23dd9e02a590174a84691f954eff4346888 1.40kB / 1.40kB 0.4s
=> => extracting sha256:648e0aadf75ac2ef63c5390adc6dc14fde37a5ad88c2870ea604df0 2.7s
=> => extracting sha256:262696647b70a57f5f7dbf97a91091e7b51c1d2537dff72a2bdf908 1.5s
=> => extracting sha256:e66d0270d23f3038e0e8c94ee9244950fbfdb582476f61736b3c28a 0.0s
=> => extracting sha256:55ac49bd649c325395133ae4f3640a07e28d9a25c4a56eb8ac3df91 0.0s
=> => extracting sha256:cbf42f5a00d268edb1684b8eb9039543669fc5f5d0aa801a01d3460 0.0s
=> => extracting sha256:8015f365966bfa259003c319a44df5bb9290d279ca775b4f249f291 0.0s
=> => extracting sha256:4cadff8bc2aa83b23dd9e02a590174a84691f954eff4346888724c0 0.0s
=> [internal] load build context 0.0s
=> => transferring context: 33.34kB 0.0s
=> [2/6] COPY favicon.ico /usr/share/nginx/html/favicon.ico 0.0s
=> [3/6] COPY index.html /usr/share/nginx/html/index.html 0.0s
=> [4/6] COPY script.js /usr/share/nginx/html/script.js 0.0s
=> [5/6] COPY style.css /usr/share/nginx/html/style.css 0.0s
=> [6/6] COPY data.json /usr/share/nginx/html/data.json 0.0s
=> exporting to image 1.2s
=> => exporting layers 1.2s
=> => writing image sha256:9f46c2925ff29c582eef7c32e63bc879fe3162cb49b484d2c60f 0.0s
=> => naming to docker.io/library/guess-the-capital 0.0s
```

3. Oluşturulan görüntülerini listeleyin

```
docker images
```

```
theia@theiadocker-[REDACTED]:~/home/project$ docker images
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE
guess-the-capital  latest   9adbca9e97  4 minutes ago  187MB
nginx           latest   eb4a57159180  7 days ago   187MB
```

4. Görüntüyü çalıştırın

```
docker run -it -d -p 8080:80 guess-the-capital
```

5. Tarayıcıda Doğrula

[Uygulamayı Başlat](#)

Görev 2: IBM Cloud'da Dağıtım

Code Engine CLI'yi başlatarak başlayalım.

[Code Engine Projesi Oluştur in IDE](#)

```
cd ~/home/project/fyidw-guess-the-capital
docker build . -t us.icr.io/${SN_ICR_NAMESPACE}/guess-the-capital
```

```
theia@theiadocker-[REDACTED]:~/home/project/fyidw-guess-the-capital$ docker build . -t us.icr.io/${SN_ICR_NAMESPACE}/guess-the-capital
[+] Building 0.3s (11/11) FINISHED
=> [internal] load build definition from Dockerfile          0.0s
=> => transferring dockerfile: 32B                          0.0s
=> [internal] load .dockerignore                            0.0s
=> => transferring context: 2B                           0.0s
=> [internal] load metadata for docker.io/library/nginx:latest 0.1s
=> [1/6] FROM docker.io/library/nginx@sha256:67f9a4f10d147a6e04629340e6493c9703300ca2 0.0s
=> [internal] load build context                          0.0s
=> => transferring context: 150B                         0.0s
=> CACHED [2/6] COPY favicon.ico /usr/share/nginx/html/favicon.ico 0.0s
=> CACHED [3/6] COPY index.html /usr/share/nginx/html/index.html 0.0s
=> CACHED [4/6] COPY script.js /usr/share/nginx/html/script.js 0.0s
=> CACHED [5/6] COPY style.css /usr/share/nginx/html/style.css 0.0s
=> CACHED [6/6] COPY data.json /usr/share/nginx/html/data.json 0.0s
=> exporting to image                                    0.0s
=> => exporting layers                                 0.0s
=> => writing image sha256:9f46c2925ff29c582eef7c32e63bc879fe3162cb49b484d2c60f2bcae6 0.0s
=> => naming to us.icr.io/sn-labs-[REDACTED]/guess-the-capital 0.0s
```

Görüntüyü IBM Cloud'a yükleyin

```
docker push us.icr.io/${SN_ICR_NAMESPACE}/guess-the-capital
```

```
theia@theiadocker-[REDACTED]:~/home/project/fyidw-guess-the-capital$ docker push us.icr.io/${SN_ICR_NAMESPACE}/guess-the-capital
Using default tag: latest
The push refers to repository [us.icr.io/sn-labs-[REDACTED]/guess-the-capital]
2312f964fb3: Pushed
88d643ad324f: Pushed
5af561e009ff: Pushed
d9e09fe5565a: Pushed
263b485e3d75: Pushed
9e96226c58e7: Pushed
12a568acc014: Pushed
7757099e19d2: Pushed
b78627b2f13: Pushed
4ca101c49e14: Pushed
a83110139647: Pushed
ac4d164fe90: Pushed
latest: digest: sha256:5529ece02a96a33195669ca90063d7a8d77dd0b04898ac3567b778b03533dd05 size: 2817
```

Görüntüyü IBM CE'ye dağıtm.

```
ibmcloud ce application create --name guess-the-capital --image us.icr.io/${SN_ICR_NAMESPACE}/guess-the-capital --registry-secret icr-secret --port 80
```

```
theia@theidocker:~/home/project/fyidw-guess-the-capital$ ibmcloud ce application create --name guess-the-capital --image us.icr.io/${SN_ICR_NAMESPACE}/guess-the-capital --registry-secret icr-secret --port 80
Creating application 'guess-the-capital'...
The Route is still working to reflect the latest desired specification.
Configuration 'guess-the-capital' is waiting for a Revision to become ready.
Ingress has not yet been reconciled.
Waiting for load balancer to be ready.
Run 'ibmcloud ce application get -n guess-the-capital' to check the application status.
OK
https://guess-the-capital.13y9j7uqjreh.us-south.codeengine.appdomain.cloud
```

Çıktıdan Cloud URL'sini alın; bu, aşağıdaki gibi görülmektedir: <https://guess-the-capital.somerandomalphphanumeric.us-south.codeengine.appdomain.cloud> ve tarayıcınızda açın.

İsteğe bağlı olarak durumu kontrol edin.

```
ibmcloud ce application get --name guess-the-capital
```

Tebrikler

Bu son laboratuvarı tamamladınız; bu laboratuvar, standart bir JavaScript uygulamasını Docker'da ve IBM Cloud'da nasıl dağıtıp barındıracağınızı gösterdi.

Yazar(lar)

[Muhammad Yahya](#)

© IBM Corporation. Tüm hakları saklıdır.