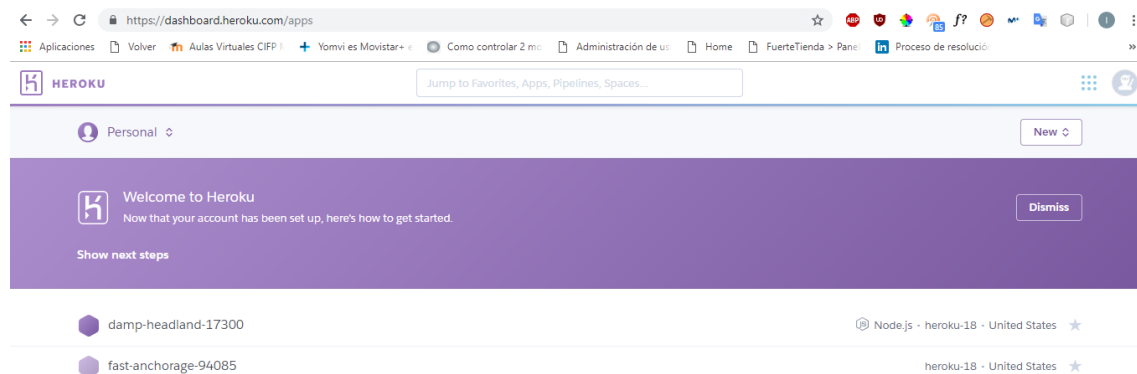
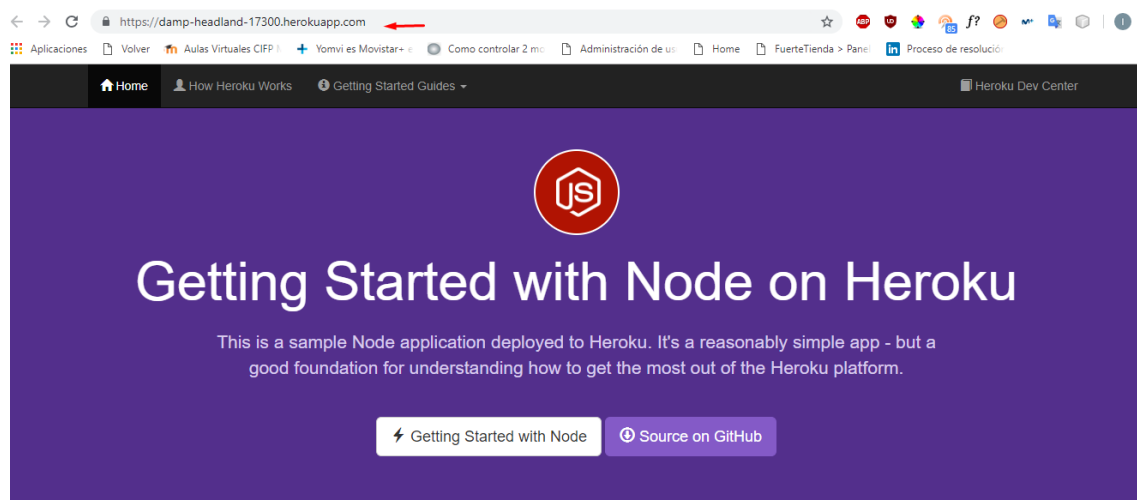


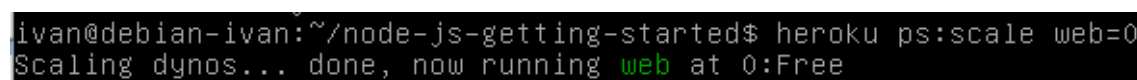
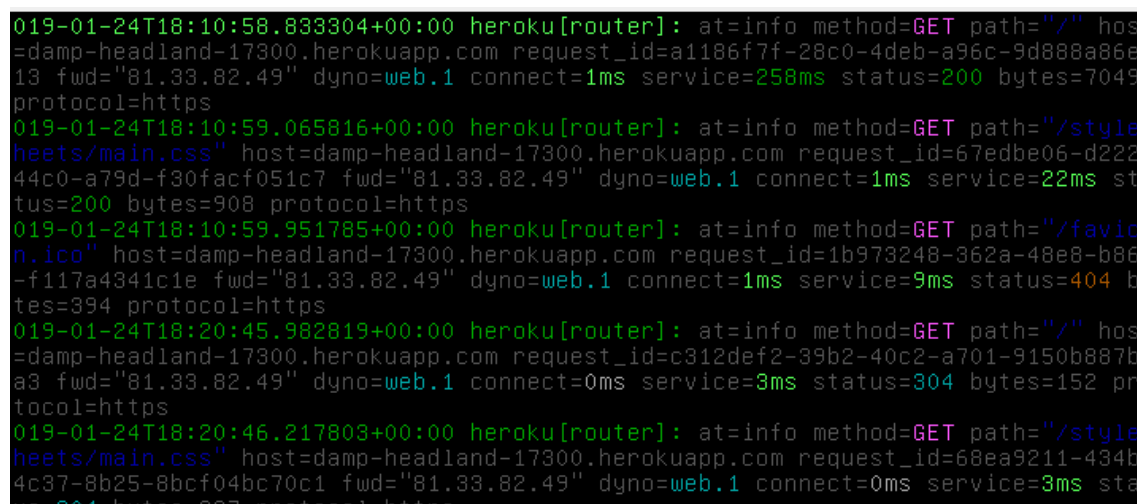
## Hemos realizado nuestro Deploy



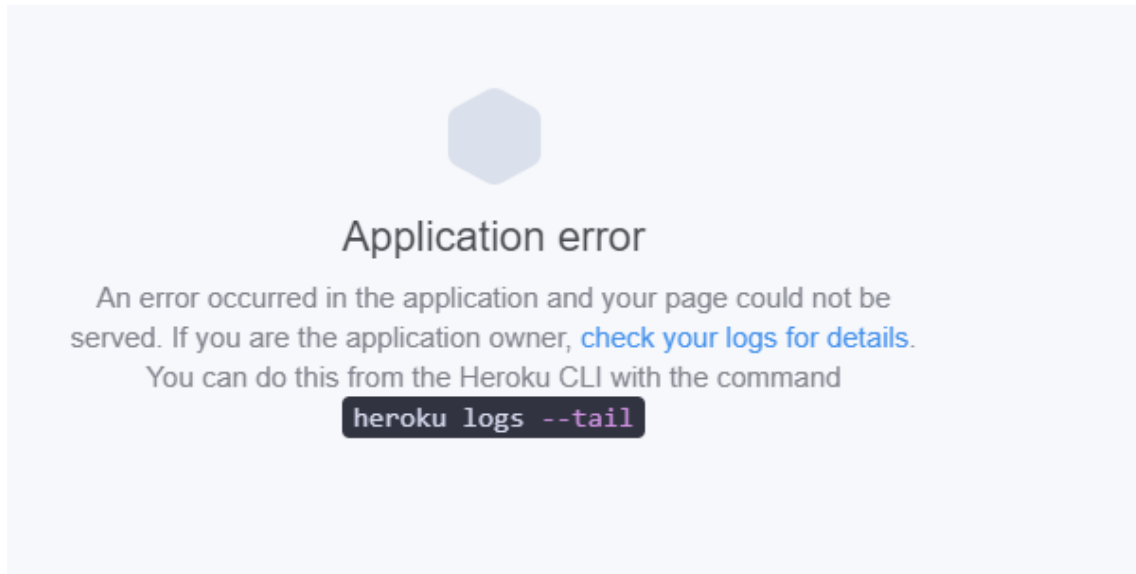
## Entramos en la web



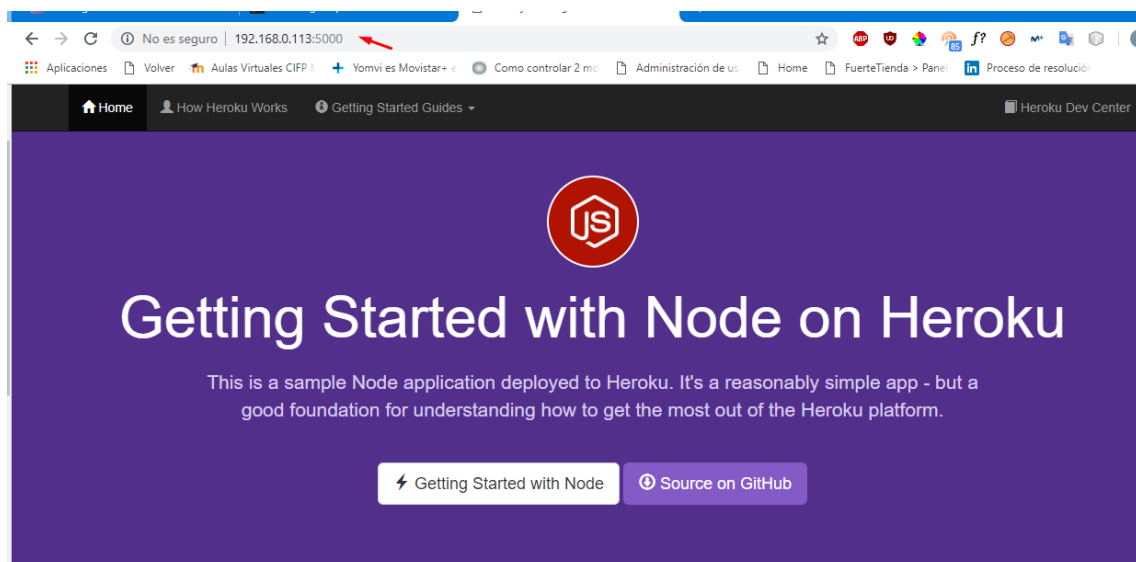
## Vemos el log



Creamos 0 instancias de la aplicación, nos conectamos en el navegador



Seguimos el tutorial y vamos a probar la aplicación en local

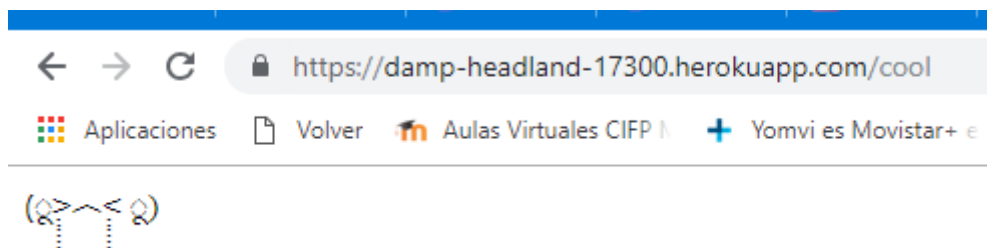


Realizamos el apartado Push local changes , modificamos los archivos descritos.

```
ivan@debian-ivan:~/node-js-getting-started$ cat index.js
const cool = require('cool-ascii-faces');
const express = require('express')
const path = require('path')
const PORT = process.env.PORT || 5000

express()
  .use(express.static(path.join(__dirname, 'public')))
  .set('views', path.join(__dirname, 'views'))
  .set('view engine', 'ejs')
  .get('/', (req, res) => res.render('pages/index'))
  .get('/cool', (req, res) => res.send(cool()))
  .listen(PORT, () => console.log(`Listening on ${ PORT }`))
ivan@debian-ivan:~/node-js-getting-started$
```

Actualizamos el repositorio, volvemos al navegador y vemos como funciona.



Probamos la consola de heroku “Start a console”

```
ivan@debian-ivan:~/node-js-getting-started$ heroku run bash
Running bash on damp-headland-17300... up, run.7882 (Free)
~ $ ls
Procfile  app.json  node_modules  package.json  test.js
README.md index.js  package-lock.json  public        views
~ $ exit
exit
ivan@debian-ivan:~/node-js-getting-started$
```

Definimos las “configs vars”

```
ivan@debian-ivan:~/node-js-getting-started$ cat index.js
const cool = require('cool-ascii-faces');
const express = require('express')
const path = require('path')
const PORT = process.env.PORT || 5000

express()
  .use(express.static(path.join(__dirname, 'public')))
  .set('views', path.join(__dirname, 'views'))
  .set('view engine', 'ejs')
  .get('/', (req, res) => res.render('pages/index'))
  .get('/cool', (req, res) => res.send(cool()))
  .get('/times', (req, res) => res.send(showTimes()))
  .listen(PORT, () => console.log(`Listening on ${ PORT }`))

showTimes = () => {
  let result = ''
  const times = process.env.TIMES || 5
  for(i = 0; i < times; i++){
    result += i + ' '
  }
  return result;
}
ivan@debian-ivan:~/node-js-getting-started$
```

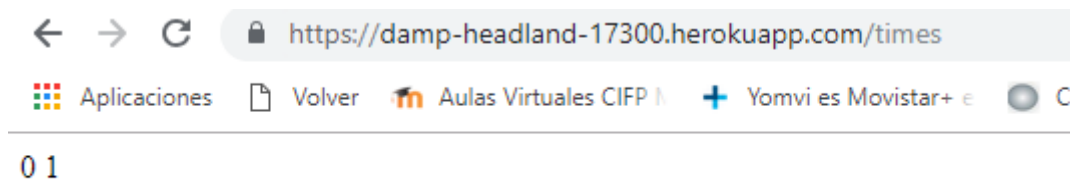
Configuramos TIMES

```
ivan@debian-ivan:~/node-js-getting-started$ heroku config:set TIMES=2
Setting TIMES and restarting ⚡ damp-headland-17300... done, v5
TIMES: 2
ivan@debian-ivan:~/node-js-getting-started$
```

### Realizamos un heroku config

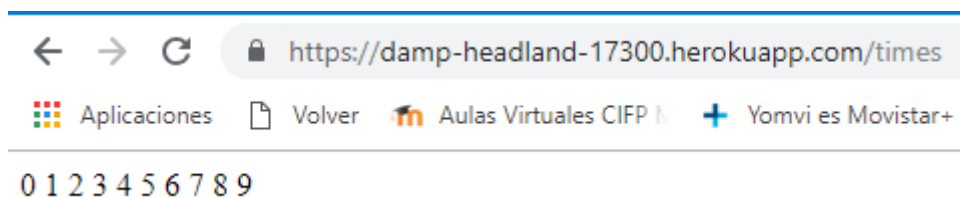
```
ivan@debian-ivan:~/node-js-getting-started$ heroku config
=== damp-headland-17300 Config Vars
TIMES: 2
ivan@debian-ivan:~/node-js-getting-started$
```

### Probamos en el navegador y vemos como da 2 vueltas en el for



A screenshot of a web browser window. The address bar shows the URL <https://damp-headland-17300.herokuapp.com/times>. Below the address bar, there are several icons and text: 'Aplicaciones', 'Volver', 'Aulas Virtuales CIEP', 'Yomvi es Movistar+', and a search icon. The main content area of the browser displays the text '0 1'.

### Cambiamos el valor de time a 10, y observamos que da 10 vueltas



A screenshot of a web browser window. The address bar shows the URL <https://damp-headland-17300.herokuapp.com/times>. Below the address bar, there are several icons and text: 'Aplicaciones', 'Volver', 'Aulas Virtuales CIEP', 'Yomvi es Movistar+', and a search icon. The main content area of the browser displays the text '0 1 2 3 4 5 6 7 8 9'.

## Preguntas

Una vez hecho el deploy comprueba el contenido de tu Procfile y explica el significado de su contenido.

```
index.js package.json Procfile README.md view
ivan@debian-ivan:~/node-js-getting-started$ cat Procfile
web: node index.js
```

El Procfile, básicamente es el comando con el que se ejecutara la aplicación en la web.

Explica brevemente para qué se usa dyno en Heroku.

Los dynos son piezas fundamentales del modelo de arquitectura de Heroku, son las unidades que proveen capacidad de cómputo dentro de la plataforma, están basados en contenedores Linux.

Cada dyno esta aislado del resto, por lo que cualquier acción que hagamos, no perjudicara a los otros.