

## Preparando los proyectos SpringBoot para Heroku

\*El ejemplo mostrado a continuacion es de un proyecto llamado SpringEureka [ reemplazelo por el nombre de su proyecto ]

Abrir una terminal e ir a la carpeta del proyecto

```
C:\WINDOWS\system32\cmd.exe
```

```
E:\Proyectos\workspaces\sts-workspace\SpringEureka>
```

## Realizar un mvn compile

```
C:\WINDOWS\system32\cmd.exe
```

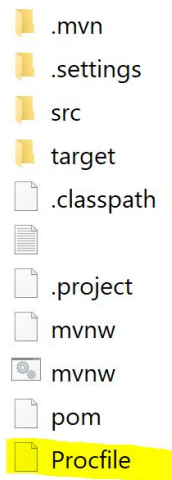
```
E:\Proyectos\workspaces\sts-workspace\SpringEureka>mvn compile
[INFO] Scanning for projects...
[INFO]
[INFO] -----
[INFO] Building SpringEureka 0.0.1-SNAPSHOT
[INFO] -----
[INFO] --- maven-resources-plugin:2.6:resources (default-resources) @ SpringEureka ---
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] Copying 1 resource
[INFO] Copying 0 resource
[INFO] --- maven-compiler-plugin:3.1:compile (default-compile) @ SpringEureka ---
[INFO] Changes detected - recompiling the module!
[INFO] Compiling 1 source file to E:\Proyectos\workspaces\sts-workspace\SpringEureka\target\classes
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 4.171 s
[INFO] Finished at: 2017-08-23T16:55:42-05:00
[INFO] Final Memory: 30M/320M
[INFO] -----
E:\Proyectos\workspaces\sts-workspace\SpringEureka>
```

## Realizar un mvn package -Dmaven.test.skip=true (Esperar que termine las descargas)

```
C:\WINDOWS\system32\cmd.exe
```

```
Downloaded: https://repo.maven.apache.org/maven2/asm/asm/3.3.1/asm-3.3.1.jar (44 kB at 21 kB/s)
Downloading: https://repo.maven.apache.org/maven2/org/jdom/jdom/1.1/jdom-1.1.jar
Downloaded: https://repo.maven.apache.org/maven2/org/apache/maven/wagon/wagon-provider-api/1.0-
-1.0-beta-6.jar (53 kB at 25 kB/s)
Downloading: https://repo.maven.apache.org/maven2/org/apache/maven/shared/maven-dependency-tree
e-2.1.jar
Downloaded: https://repo.maven.apache.org/maven2/asm/asm-tree/3.3.1/asm-tree-3.3.1.jar (22 kB a
Downloaded: https://repo.maven.apache.org/maven2/asm/asm-commons/3.3.1/asm-commons-3.3.1.jar (3
Downloading: https://repo.maven.apache.org/maven2/org/vafer/jdependency/0.7/jdependency-0.7.jar
Downloaded: https://repo.maven.apache.org/maven2/org/apache/maven/maven-compat/3.0/maven-compat
B/s)
Downloading: https://repo.maven.apache.org/maven2/commons-io/commons-io/1.3.2/commons-io-1.3.2.
Downloaded: https://repo.maven.apache.org/maven2/asm/asm-analysis/3.2/asm-analysis-3.2.jar
Downloaded: https://repo.maven.apache.org/maven2/org/vafer/jdependency/0.7/jdependency-0.7.jar
Downloading: https://repo.maven.apache.org/maven2/asm/asm-util/3.2/asm-util-3.2.jar
Downloaded: https://repo.maven.apache.org/maven2/asm/asm-analysis/3.2/asm-analysis-3.2.jar (18
Downloaded: https://repo.maven.apache.org/maven2/org/jdom/jdom/1.1/jdom-1.1.jar (153 kB at 65 k
Downloaded: https://repo.maven.apache.org/maven2/org/apache/maven/shared/maven-dependency-tree/
-2.1.jar (60 kB at 25 kB/s)
Downloaded: https://repo.maven.apache.org/maven2/commons-io/commons-io/1.3.2/commons-io-1.3.2.j
Downloaded: https://repo.maven.apache.org/maven2/asm/asm-util/3.2/asm-util-3.2.jar (37 kB at 14
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 21.722 s
[INFO] Finished at: 2017-08-23T16:56:52-05:00
[INFO] Final Memory: 29M/303M
[INFO] -----
E:\Proyectos\workspaces\sts-workspace\SpringEureka>
```

Crear el archivo Procfile (un archivo sin extensión) con el siguiente contenido



web: java -Dserver.port=\$PORT -jar target/SpringEureka-0.0.1-SNAPSHOT.jar

Logearse en heroku en la terminal

```
E:\Proyectos\workspaces\sts-workspace\SpringEureka>heroku login
Enter your Heroku credentials:
Email:
```

Crear un repo en git y seguir los siguientes pasos (tener en cuenta el contenido de .gitignore)

(Por única vez estos 2 pasos)

git config --global user.email "[you@example.com](mailto:you@example.com)"

git config --global user.name "Your Name"

git init

git add .

git commit -m "Inicial"

heroku apps:create nombre\_que\_desea

git push heroku master

Si todo lo anterior fue correcto puede ejecutar el siguiente comando

heroku open

[Probar en la url sus servicios sin data]

-----

\*Si este servicio maneja base de datos, ejecutar lo siguiente

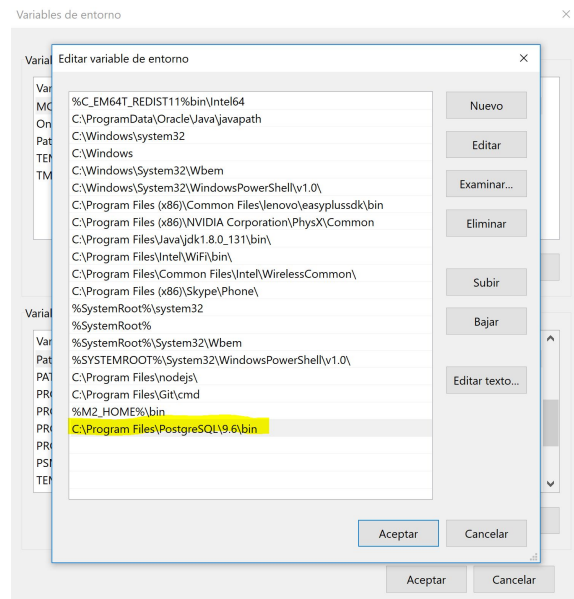
heroku pg:psql

```
E:\Proyectos\workspaces\sts-workspace\SpringMicro>heroku pg:psql
```

Si le sale la siguiente información (usualmente en windows) es porque no tiene asociado psql como variable de entorno del sistema operativo

```
--> Connecting to postgresql-flat-72773
! The local psql command could not be located. For help installing psql, see
! https://devcenter.heroku.com/articles/heroku-postgresql#local-setup
```

Deberá agregar la ruta de la carpeta de instalación de postgre al path de windows (proceso similar a la instalación de maven)



Reiniciar y luego cuando intente ejecutar el comando heroku pg:psql obtendrá algo similar a esto

```
--> Connecting to postgresql-flat-72773
psql (9.6.3, servidor 9.6.4)
ADVERTENCIA: El código de página de la consola (437) difiere del código
de página de Windows (1252).
Los caracteres de 8 bits pueden funcionar incorrectamente.
Vea la página de referencia de psql «Notes for Windows users»
para obtener más detalles.
conexión SSL (protocolo: TLSv1.2, cifrado: ECDHE-RSA-AES256-GCM-SHA384, bits: 256, compresión: desactivado)
Digite «help» para obtener ayuda.

mito-micro:DATABASE=>
```

[Se abrirá una consola de postgre, ahí ejecutar las sentencias sql, cada sentencia deberá terminar en “;” para ejecutarse]

Ejecutar su código sql

Para salir del modo de base de datos, presionar CTRL + C, opción “S”

Luego heroku open y probar sus servicios.