## Getting Started with Node.js on HIntroduction

This tutorial will have you deploying a Node.js app in minutes.

Hang on for a few more minutes to learn how it all works, so you can make the most out of Heroku.

The tutorial assumes that you have a free [Heroku account](https://signup.heroku.com/signup/dc), and that you have [Node.js](http://nodejs.org/) and[npm](https://github.com/npm/npm#synopsis) installed.

ErSet up

In this step you will install the Heroku Toolbelt. This provides you access to the Heroku Command Line utility, as well as git and Foreman, tools you’ll use in later steps.

Download Heroku Toolbelt for Windows

Once installed, you can use the heroku command from your command shell.

On Windows, start the Command Prompt (cmd.exe) or Powershell to access the command shell.

Log in using the email address and password you used when creating your Heroku account:

$ heroku login

Enter your Heroku credentials.

Email: zeke@example.com

Password:

Authentication successful.

Authenticating is required to allow both the heroku and git commands to operate.

Note that if you’re behind a firewall that requires use of a proxy to connect with external HTTP/HTTPS services, [you can set the HTTP\_PROXY or HTTPS\_PROXY environment variables](https://devcenter.heroku.com/articles/using-the-cli#using-an-http-proxy)in your local development environment before running the heroku command.

[Log in to report a problem](https://devcenter.heroku.com/login?back_to=%2Farticles%2Fgetting-started-with-nodejs%23set-up) [I have installed the Toolbelt](https://devcenter.heroku.com/articles/set-up)

([Log in](https://devcenter.heroku.com/login?back_to=%2Farticles%2Fgetting-started-with-nodejs%23set-up) to save and track your progress)

Prepare the app

In this step, you will prepare a simple application that can be deployed.

Execute the following commands to clone the sample application:

$ git clone https://github.com/heroku/node-js-getting-started.git

$ cd node-js-getting-started

You now have a functioning git repository that contains a simple application as well as apackage.json file, which is used by Node’s dependency manager.

## Deploy the app

In this step you will deploy the app to Heroku.

Create an app on Heroku, which prepares Heroku to receive your source code.

$ heroku create

Creating sharp-rain-871... done, stack is cedar-14

http://sharp-rain-871.herokuapp.com/ | https://git.heroku.com/sharp-rain-871.git

Git remote heroku added

When you create an app, a git remote (called heroku) is also created and associated with your local git repository.

Heroku generates a random name (in this case sharp-rain-871) for your app, or you can pass a parameter to specify your own app name.

Now deploy your code:

$ git push heroku master

Counting objects: 343, done.

Delta compression using up to 4 threads.

Compressing objects: 100% (224/224), done.

Writing objects: 100% (250/250), 238.01 KiB, done.

Total 250 (delta 63), reused 0 (delta 0)

-----> Node.js app detected

-----> Resolving engine versions

Using Node.js version: 0.12.2

Using npm version: 1.2.18

-----> Fetching Node.js binaries

-----> Vendoring node into slug

-----> Installing dependencies with npm

....

Dependencies installed

-----> Building runtime environment

-----> Discovering process types

Procfile declares types -> web

-----> Compiled slug size: 4.1MB

-----> Launching... done, v9

http://sharp-rain-871.herokuapp.com deployed to Heroku

To git@heroku.com:sharp-rain-871.git

\* [new branch] master -> master

The application is now deployed. Ensure that at least one instance of the app is running:

$ heroku ps:scale web=1

Now visit the app at the URL generated by its app name. As a handy shortcut, you can open the website as follows:

$ heroku open

[Log in to report a problem](https://devcenter.heroku.com/login?back_to=%2Farticles%2Fgetting-started-with-nodejs%23deploy-the-app) [I have deployed my app on Heroku](https://devcenter.heroku.com/articles/deploy-the-app)