# Node installation

* Node.js runtime
  + Performance counters
  + Event tracing (ETW)
* npm package manager
* Online documentation shortcuts
* Add to PATH
  + Node.js and npm
  + npm modules

# Node Lite-Server

## Install Lite-Server

- Launch the Bash command line

- Install lite-server:

**$ npm install -g lite-server**

## New Project

* Create a directory
* Create a Node project:

**$ npm init –y**

package.json:

{

"name": "NodeJS",  
"version": "1.0.0",  
"description": "",  
"main": "index.js",  
"scripts":

{  
"test": "echo \"Error: no test specified\" && exit 1"  
},

"keywords": [],  
"author": "",  
"license": "ISC"

}

* Create an index.html file with the text "Hola Mundo" in the project
* Modify the package.json so in the "start" the lite-server will be launched:

"scripts":  
{  
"start": "lite-server"  
},

## Start and close the server

* Launch the Project:

**$ npm start**

The server is launched and listenning at <http://localhost:3000/>

All plugins introduced at the dependencies JSON of the package.JSON file, are taken from de Node Package Manager (npm). [www.npmjs.com](http://www.npmjs.com)

**$ npm install** 🡪 executes package.JSON and install all plugins writen in the sub-JSON dependencies

[**https://docs.npmjs.com/getting-started/semantic-versioning**](https://docs.npmjs.com/getting-started/semantic-versioning)

<https://blog.risingstack.com/your-first-node-js-http-server/>

<https://jesuslc.com/>

<https://geekytheory.com/node-js-primeros-pasos-y-hola-mundo>

YEOMAN 🡪 scaffolding

Comando: **yo**

Comprar un dominio, configurar las DNS para que apunten al router de la máquina servidor y permitir la entrada por el puerto 80, además de establecer un IP fija.

# NodeProjectExample

Creamos la carpeta del nuevo proyecto, entramos en ella y ejecutamos el primer comando:

1. **$ mkdir newProject**
2. **$ cd newProject/**
3. **$ npm init**
4. **$ npm install bootstrap**
5. **$ npm install jquery**
6. **$ npm install Popper.js**

The package.json file has a key called: “dependencies” with the required modules for the project. We can copy the package.json file in a new Project and then execute the command line **$ npm install** 🡪 automatically, the resources will be updated to the last version according to the semantic-versioning description of the dependencies sub-json.

<https://es.wikipedia.org/wiki/Anexo:C%C3%B3digos_de_estado_HTTP>

<https://www.joezimjs.com/javascript/3-ways-to-parse-a-query-string-in-a-url/>