

# Node.js

## Lesson 02—Node.js Project



# Lesson Overview

In this lesson, you will understand what NPM is and will create a node.js project. You will also understand the importance of json package, and you'll learn to differentiate between local and global package manager.

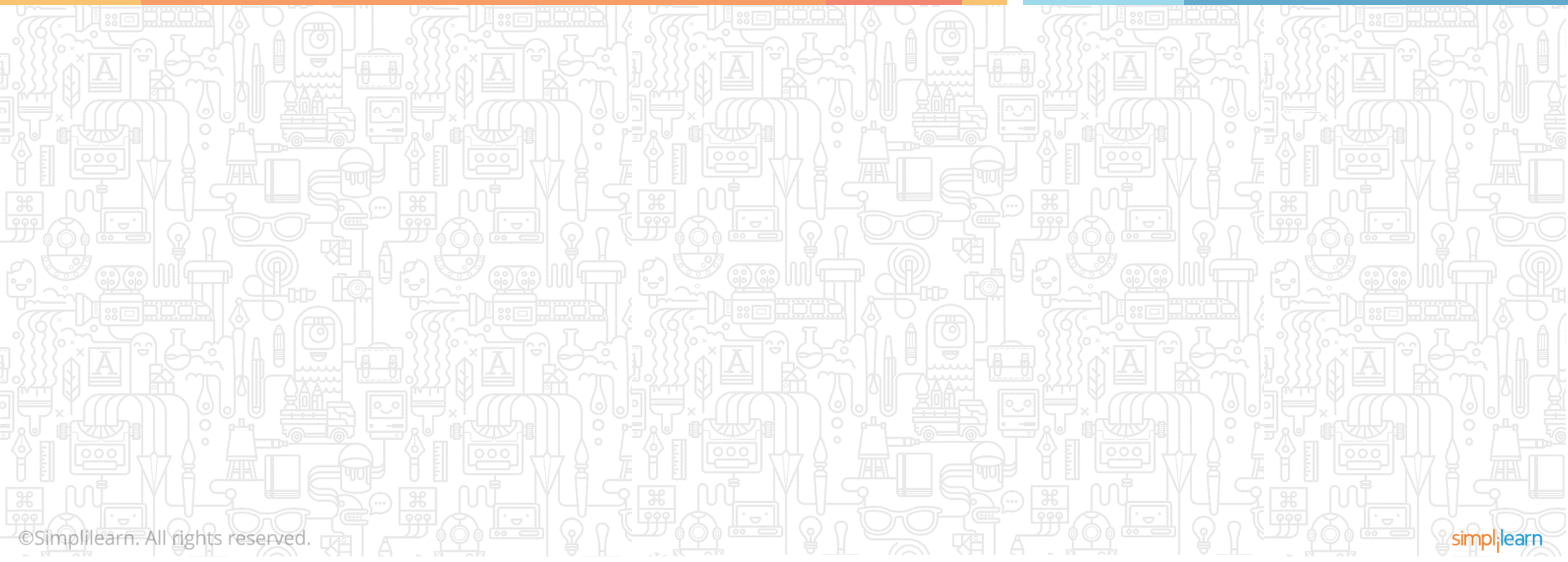
# Learning Objectives



- ✓ Define Node Package Manager (NPM)
- ✓ Create Node.js Project
- ✓ Learn about package.json
- ✓ Compare Global and Local Package Manager

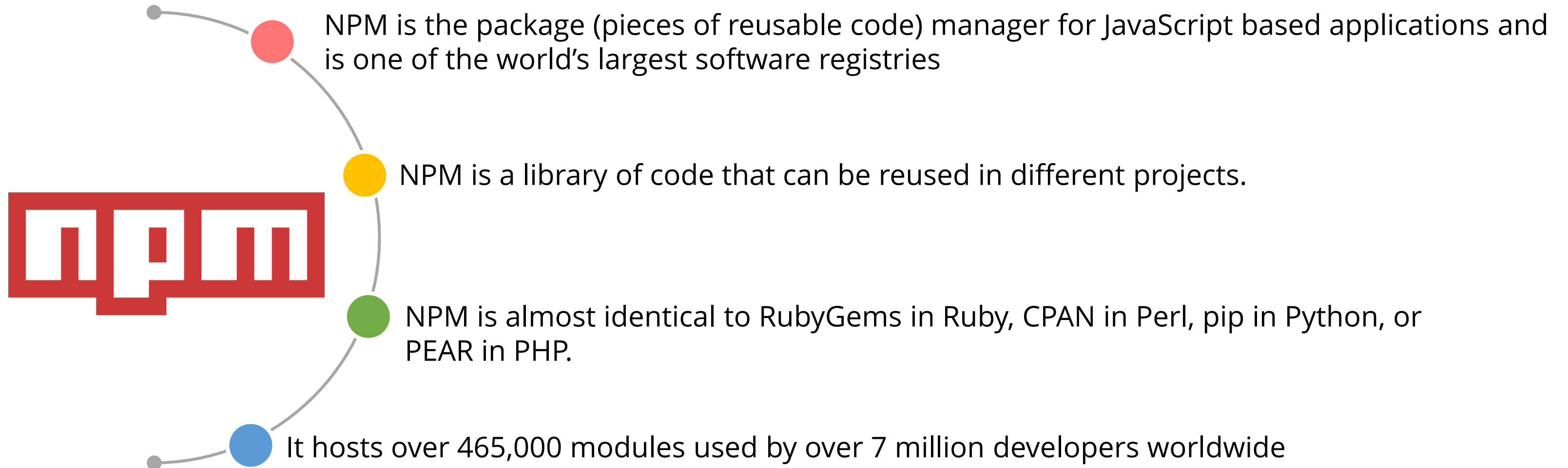
# Node.js Project

## Topic 1—Define Node Package Manager (NPM)



# Define Node Package Manager (NPM)

## INTRODUCTION

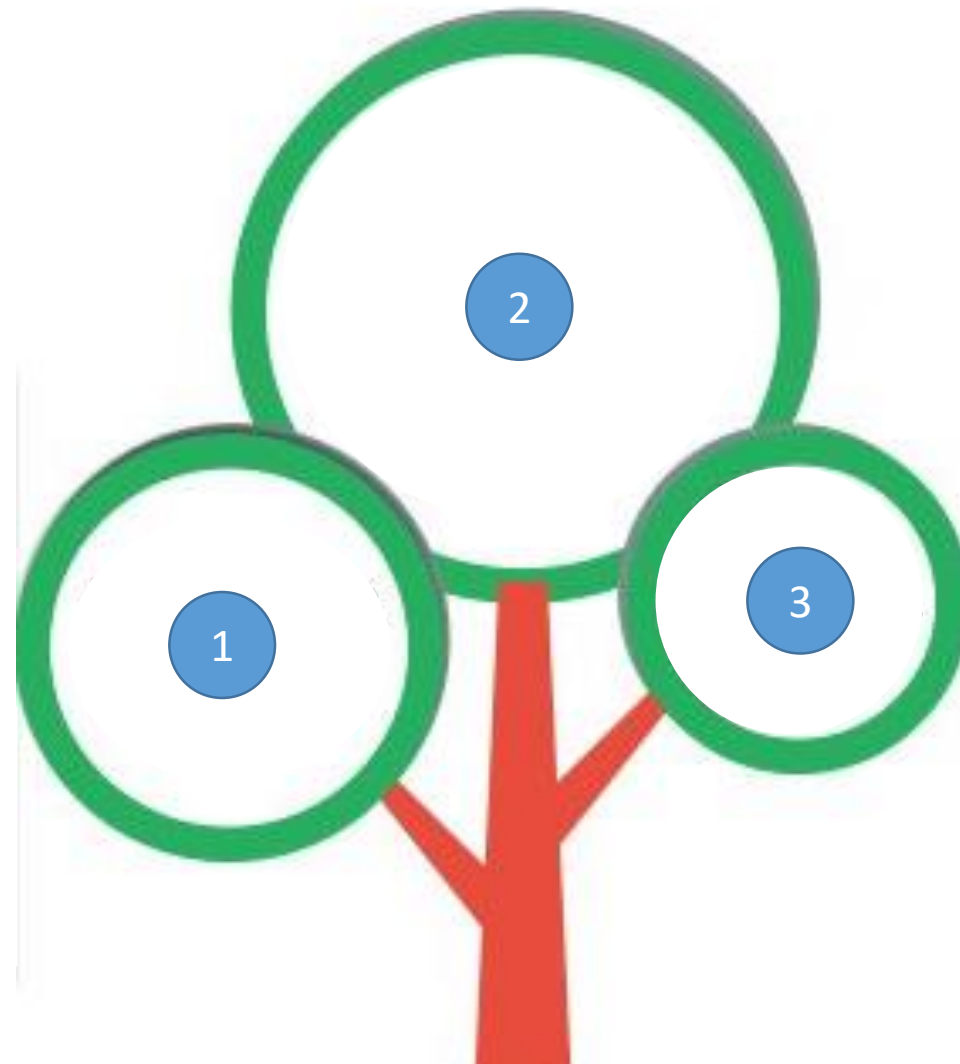


# Define Node Package Manager (NPM)

## EXAMPLES OF NPM

A standard library for validating user data (Form/Non-Form Based)

A standard library for  
interacting with databases  
(Nosql/Rdbms)



A standard library for  
parsing YAML files



For new developers, using Node.js modules provide a great way to get started as they can acquire the expertise of experienced developers. As developers, we must be familiar with NPM and the related libraries that it can offer you.

# Define Node Package Manager (NPM)

## UPDATING THE NPM VERSION AND INSTALLING MODULE

NodeJS comes preloaded with npm, so you will have a standard version of npm. However, NPM updates more frequently than Node does, so make sure you have the latest version.



**Version**

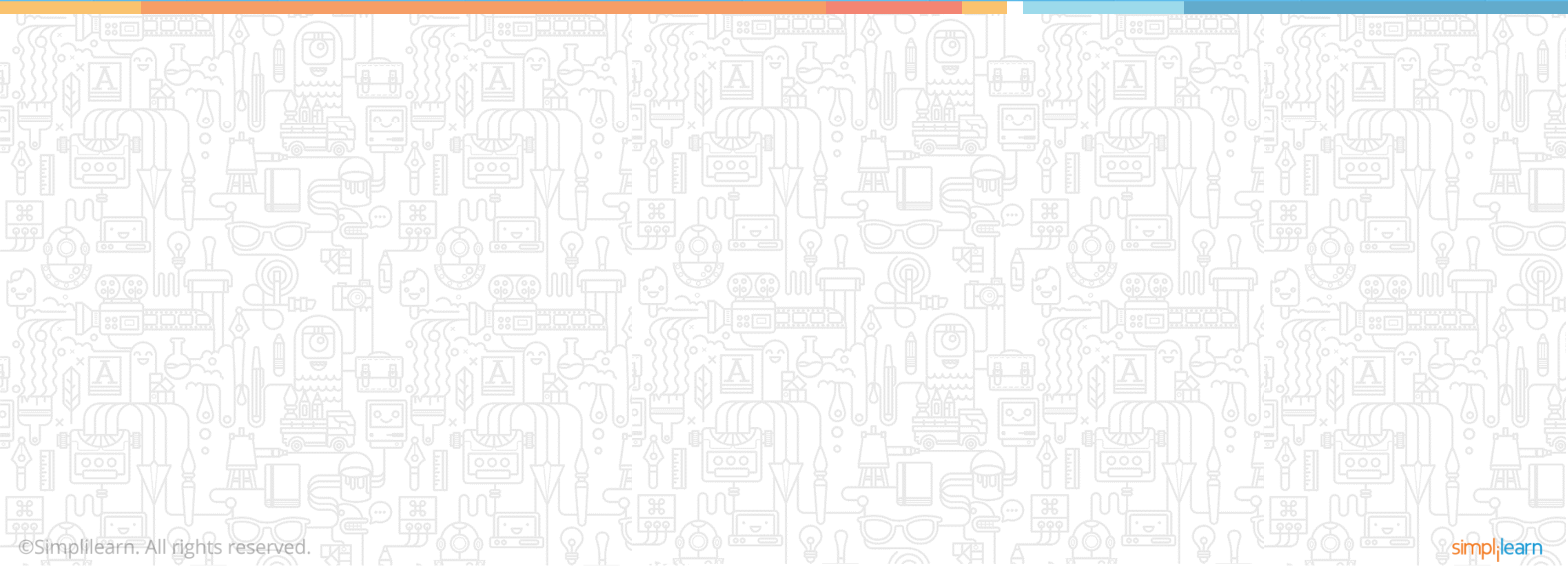
Command to update NPM to the latest version (the current version is 5.x) is

```
Npm install -g npm
```



# Node.js Project

## Topic 2—Node.js Project





# Node.js Project

---

We will write a very simple *Hello World* application that returns the message "welcome to node.js" to any incoming HTTP request. This is a basic application that will help you to setup node.js. This can later be replaced with a new application.

Type *notepad hello.js* in  
command prompt



# Node.js Project

---

We will write a very simple *Hello World* application that returns the message “welcome to node.js” to any incoming HTTP request. This is a basic application that will help you to setup node.js. This can later be replaced with a new application.

Type notepad hello.js in  
command prompt



```
var http = require('http');  
http.createServer(function (req, res) {  
  res.writeHead(200, {'Content-Type': 'text/plain'});  
  res.end('welcome to nodejs\n');  
}).listen(8080, 'localhost');  
console.log('Server is running at port:8080');
```

# Node.js Project

We will write a very simple *Hello World* application that returns the message "welcome to node.js" to any incoming HTTP request. This is a basic application that will help you to setup node.js. This can later be replaced with a new application.

Type notepad hello.js in  
command prompt

Save the file

```
var http = require('http');
http.createServer(function (req, res) {
  res.writeHead(200, {'Content-Type': 'text/plain'});
  res.end('welcome to nodejs\n');
}).listen(8080, 'localhost');
console.log('Server is running at port:8080');
```

# Node.js Project

---

We will write a very simple Hello World application that returns the message "welcome to node.js" to any incoming HTTP request. This is a basic application that will help you to setup node.js. This can later be replaced with a new application.

Type notepad hello.js in  
command prompt

Save the file

```
var http = require('http');
http.createServer(function (req, res) {
  res.writeHead(200, {'Content-Type': 'text/plain'});
  res.end('welcome to nodejs\n');
}).listen(8080, 'localhost');
console.log('Server is running at port:8080');
```

Run with node server.js

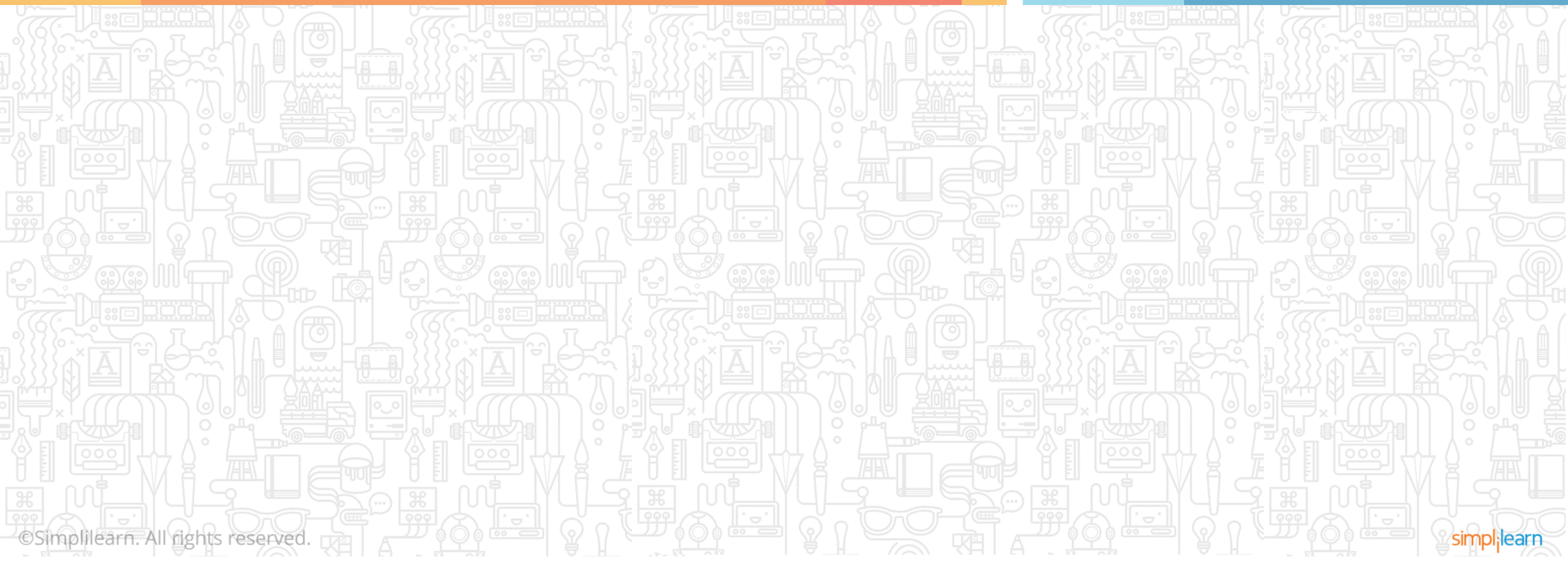
# Demo for creating node.js project

---



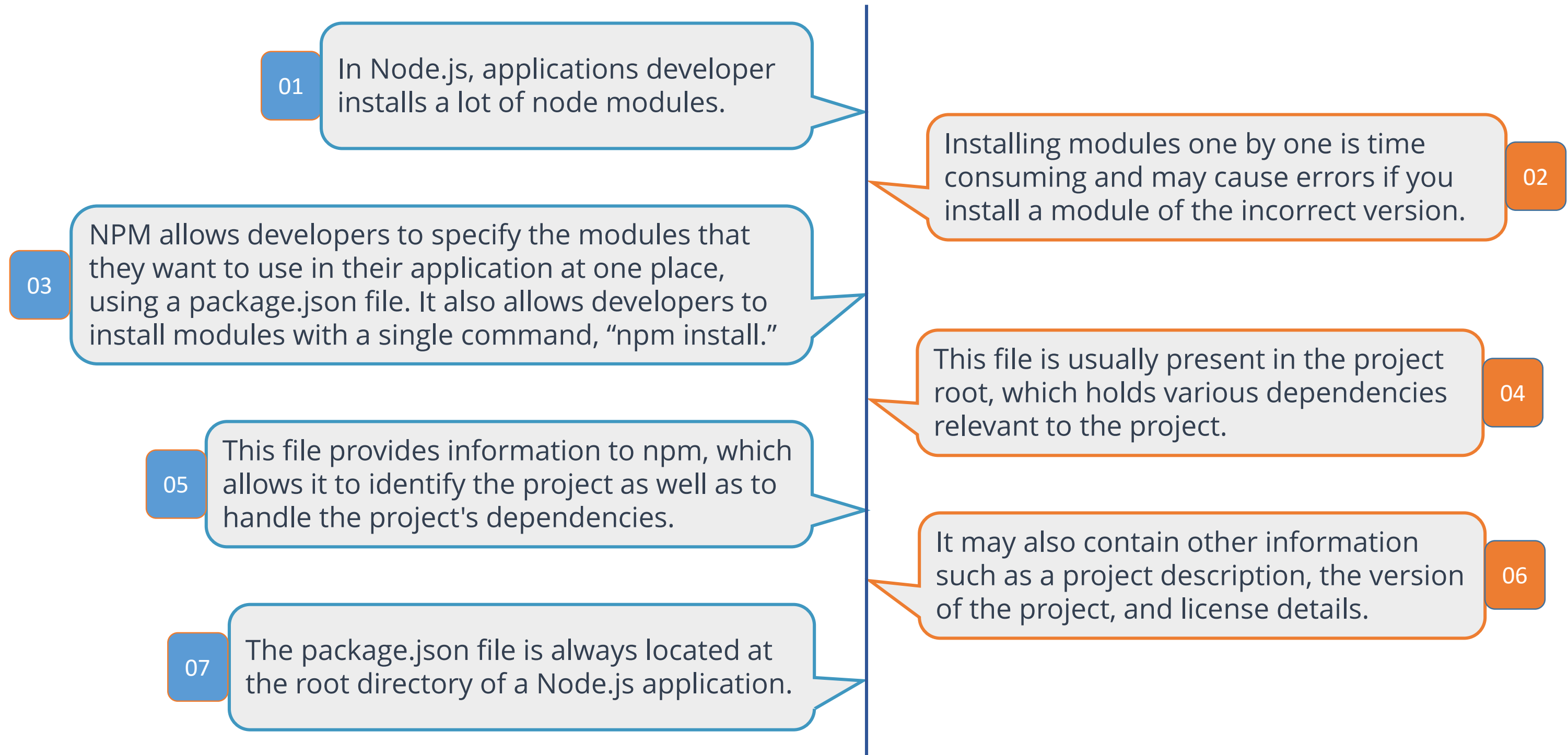
# Node.js Project

## Topic 3—Importance of Package.json





# Importance of Package.json



# Steps for Creating json package

---

Create a new file called package.json and add the following content:

```
{  
  "name": "myapplication",  
  "version": "2.5.1",  
  "dependencies":  
  {  
    "express": "4.15.1"  
  }  
}
```



# Steps for Creating json package

---

Create a new file called package.json and add the following content:

```
{  
  "name": "myapplication",  
  "version": "2.5.1",  
  "dependencies": {  
    "express": "4.15.1"  
  }  
}
```



Save your file as follows:  
package.json

# Steps for Creating json package

---

Create a new file called package.json and add the following content:

```
{  
  "name": "myapplication",  
  "version": "2.5.1",  
  "dependencies": {  
    "express": "4.15.1"  
  }  
}
```

To ensure that you are in the same directory as hello.js, run  
npm install

Save your file as follows:  
package.json

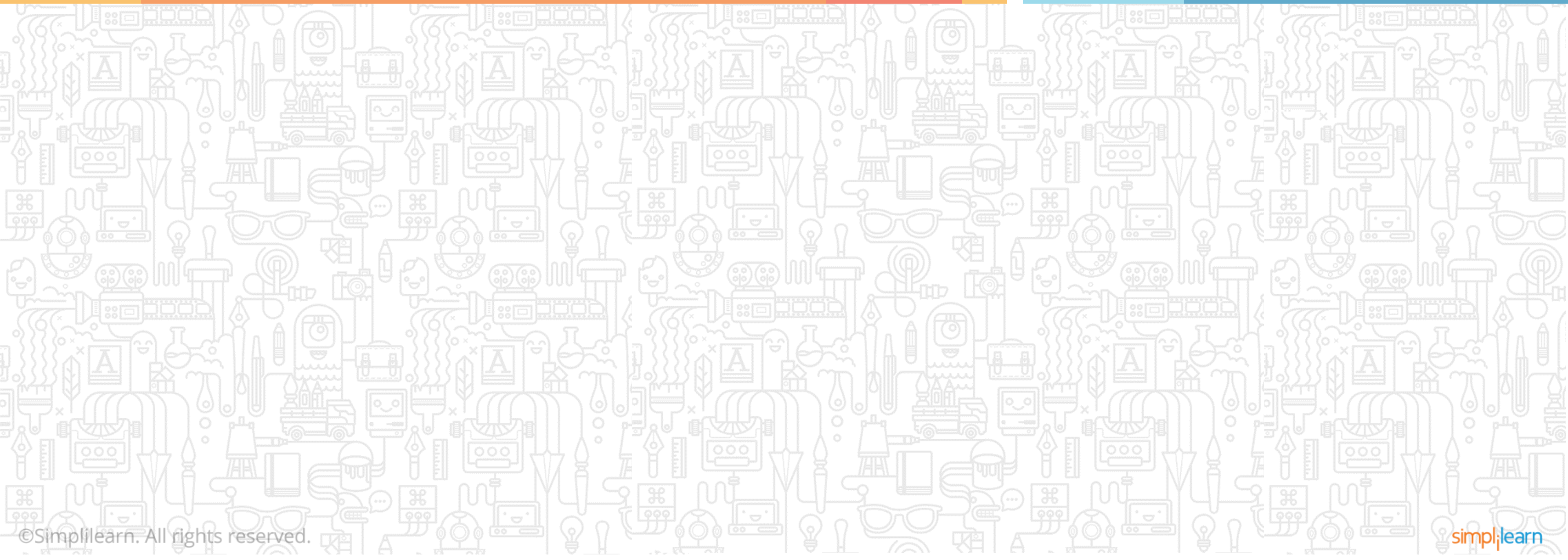
# Demo for creating json package

---



# Introduction to Node.js

## Topic 4—Global and Local Modules Installation





# Global and Local Modules Installation

---

Local Installation

Global Installation

There are two types of Installation.

# Global and Local Modules Installation

## Local Installation

## Global Installation

- In Local Installation, the module will get installed locally in a folder called `node_modules`. This is a default behavior and will happen whenever you run `npm install [module_name]`.
- If your Node.js application is called `DemoApp.js`, it will result in the following folder structure:
  - `DemoApp.js`
  - `node_modules/`
  - `module_name`

**Note:** This is a common and recommended way of installing Node.js modules.

# Global and Local Modules Installation

## Local Installation

## Global Installation

- Some node modules have executables that you may want to run wherever you are on your filesystem.
- An example of a module where you might want to use a global install is Nodemon.
- Nodemon is a utility module that will monitor for any changes in your source and will automatically restart your server
- To install modules globally, simply add the -g flag when you install it:

```
npm install -g nodemon
```

- Installing a module globally means that it can be run wherever you are on your filesystem.

# Demo for Global and Local Modules Installation

---



# Key Takeaways



- ✓ NPM is the package manager for JavaScript-based applications and is one of the world's largest software registries.
- ✓ As a node.js project, created a "Hello world" program that returns the message "welcome to node.js"
- ✓ NPM allows developers to specify the modules that they want to use in their application at one place, using a package.json file. It also allows the developers to install modules with a single command, "npm install."
- ✓ In local package, the files are installed locally in a folder called "node\_modules." In global package, the files are installed in the place where you may want to run your filesystem.



# Thank You