

**Laboratorio 6: Instalación y administración de las librerías**

**para las aplicaciones web**

**Objetivos:**

Al finalizar el laboratorio el estudiante será capaz de:

* Identificar las principales características de NPM
* Importación y uso de librerías principales de NPM
* Creación de módulos propios.
* Exportar sus propios módulos.

**Seguridad:**

* Ubicar maletines y/o mochilas en el gabinete del aula de Laboratorio.
* No ingresar con líquidos, ni comida al aula de Laboratorio.
* Al culminar la sesión de laboratorio apagar correctamente la computadora y la pantalla, y ordenar las sillas utilizadas.

**Equipos y Materiales:**

* Una computadora con:
* Windows 7 o superior
* Conexión a la red del laboratorio
* Instalador de node.js

**Procedimiento:**

**Identificación de npm**

1. Node.js viene con NPM instalado por defecto en todas sus versiones actuales. No se deberá hacer ninguna configuración adicional para hacerlo funcionar.
   1. Ingrese el comando **npm** en una terminal. Adjunte la captura con los argumento principales de la función npm y además mencione la versión con la que está trabajando.

**Creación de package.json**

1. Procedemos a iniciar nuestro proyecto creando una carpeta donde residirá nuestro código.
   1. Creamos una carpeta en C:\ con el nombre de proyecto



* 1. Nos situamos dentro de la misma para proceder a escribir el comando **npm init**



* 1. Con el comando anterior, habremos iniciado el asistente de creación de nuestro manifiesto de proyecto, o también conocido como **package.json**. Proceda a seguir las instrucciones del asistente, añadiendo aquí las capturas correspondientes a cada paso con su explicación. El nombre de proyecto será **tecsup-2017-XXX** donde **XXX** será sus dos apellidos seguidos, en minúsculas y sin acentos. Ingrese su nombre como autor del proyecto, versión 0.0.1
  2. Verifique la creación del archivo mencionado y adjunte una captura de su contenido.

**Instalación de paquetes mediante NPM**

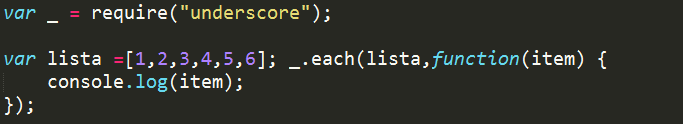
1. Procederemos a instalar un paquete mediante NPM para conocer su utilización en la vida real.
   1. Ingresamos el siguiente comando y esperamos se complete con éxito.



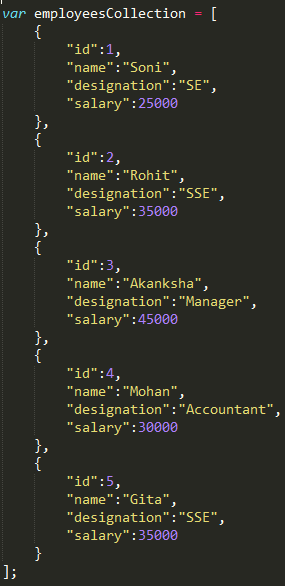
* 1. Verifiquemos si hay algún cambio en la carpeta **proyecto**. De ser así, adjunte una captura de pantalla de lo que ha sucedido.
  2. Abra el archivo package.json para verificar si ha habido algún cambio. De no ser así Ingrese el siguiente comando y ábralo nuevamente.



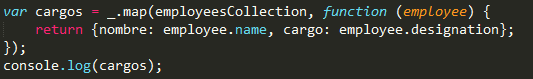
* 1. Adjunte una captura del archivo **package.json** y explique lo que ha cambiado.
  2. NPM permite instalar librerías para utilizarlas, independientemente de si estamos usando node.js o si queremos usarlas para nuestro proyecto web en el desarrollo del frontend. La diferencia de usar el argumento -**–save** es que se buscará un manifiesto de proyecto (**package.json**) donde agregar esta dependencia de nuestro proyecto.
  3. Cree el archivo **index.js** con el siguiente contenido.



* 1. Ejecutemos el resultado e interpretemos lo obtenido.
  2. Agreguemos la siguiente variable con un array de empleados de una compañía.



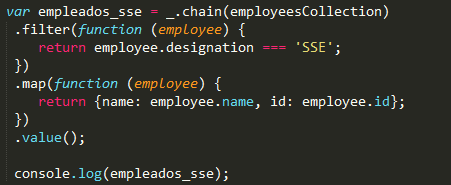
* 1. Le solicitan crear un reporte para obtener que personal está en que cargo de la compañía. Digite la siguiente función y ejecute el script.



* 1. A veces, solamente nos pedirán los nombres de los empleados.



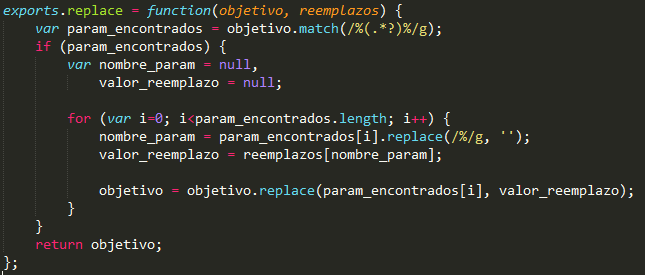
* 1. Underscore tiene funcione que realmente nos facilitarán la vida. Por ejemplo, la función **chain** permite “encadenar” distintas funciones de la misma librería para luego devolver un solo valor (de una forma parecida al Aggregation Framework de MongoDB)



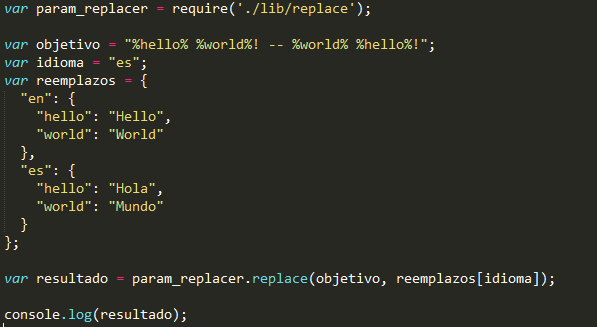
* 1. Explique el resultado obtenido de la última función.

**Creación de un paquete en node.js**

1. Uno de los ejemplos clásicos al momento de aprender a desarrollar tus propias librerías es desarrollar una que permita el reemplazo de cadenas de texto.
   1. Creamos una carpeta con el nombre **lib**.
   2. En dicha carpeta, crearemos el archivo **replace.js** con el siguiente código:



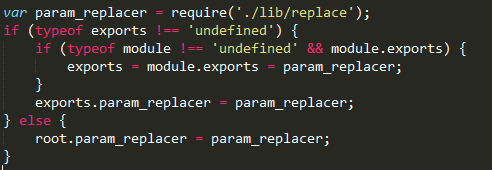
* 1. La función exports le indica a node que partes son reconocidas como integrantes del módulo creado.
  2. Para probar nuestro código que reemplaza cadenas de texto, modificaremos el contenido de **index.js** para que luzca de la siguiente manera.



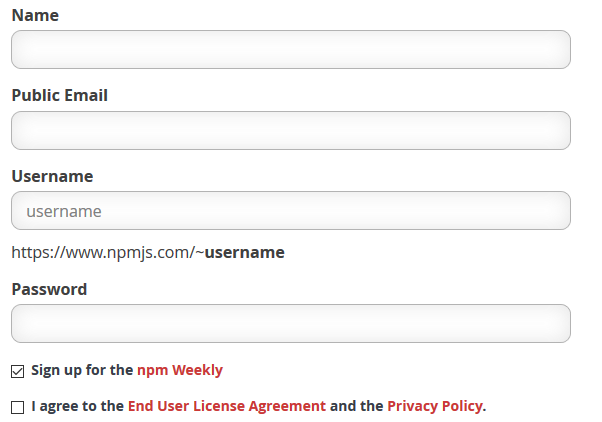
* 1. Adjunte una captura del resultado y agregue como podría mejorar el código o hacerlo más funcional.

**Publicación en NPM**

1. Lo importante de haber desarrollado un módulo es poder reutilizarlo en algún nuevo proyecto o compartirlo con la comunidad. Procederemos entonces a ver cómo se publica un módulo en NPM
   1. Primero deberemos modificar nuestro index.js para que luzca de la siguiente manera.



* 1. Ingrese a la página web de npm: <https://www.npmjs.com/>
  2. Proceda a crear una cuenta en npm. Ingrese los datos solicitados en el formulario.



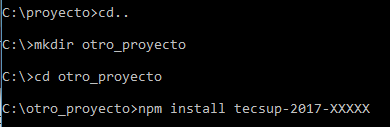
* 1. Recuerde su usuario y contraseña, ya que estos serán solicitados al momento de publicar nuestro módulo.
  2. Usamos el comando **npm adduser** para poder validar nuestras credenciales en el proyecto que tenemos creado. Procedemos a ingresar la data que nos solicite.



* 1. Para publicar un paquete en npm, basta con solicitarlo.



* 1. Procederemos a probar dicha publicación y reutilizar el código ya creado. Ingresemos la siguiente cadena de comandos para instalar el paquete replace. Coloque el nombre que le corresponde.



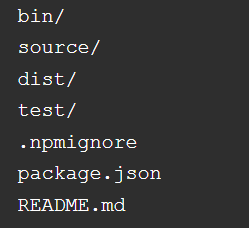
* 1. Deberíamos obtener un error, explique el por qué y cómo lo solucionará.
  2. Después de solucionarlo, utilice el código llamándolo directamente desde su nuevo proyecto. Agregue el código que utilizaría para hacer pruebas con su módulo replace. Puede basarse en la forma en la que usamos underscore.js

**Tarea:**

Desarrolle un módulo **tecsup-2017-tarea-XXX** donde **XXX** serán sus apellidos sin espacios, en minúsculas y sin acento, que contenga las siguientes funciones:

* Una función que determine la cantidad de días hasta navidad.
* Una función que devuelva la edad de una persona habiendo ingresado su fecha de nacimiento.
* Un validador de formularios: debe recibir todos los datos de un formulario, conocer de antemano los que son obligatorios y los opcionales y validar si son correctos (fechas son fechas, emails son emails, etc.)

Para este proyecto, utilice la siguiente estructura y adjunte según su investigación de internet que uso se les da a los recursos que se encuentran en este diseño.



**Conclusiones:**

Indicar las conclusiones que llegó después de los temas tratados de manera práctica en este laboratorio.

|  |
| --- |
|  |