Clase 2

Control de versiones con git

Permite a los desarrolladores rastrear y gestionar los cambios en el código fuete de manera eficiente.

Conceptos básicos de Git:

Repositorios: almacena los archivos y su historial de cambios.

Commit: registra los cambios realizados en el repositorio en un momento especifico.

Branch: crea ramas para trabajar en diferentes líneas de desarrollo.

Merge: combina los cambios de diferentes ramas en una sola.

Que es Nodo js?

Es un entorno de tiempo de ejecución de JavaScript de código abierto y multiplataforma.

Permite ejecutar código JavaScript fuera del navegador, en cualquier dispositivo que tenga instalado nodo.js

Características principales:

-Basado en motor v8 de Google chorme.

-utiliza un modelo de operaciones entrada/salida sin bloqueo y orientado a eventos, lo que lo hace altamente escalable y adecuado para aplicaciones en tiempo real y de alto rendimiento.

-Posee un ecosistema de paquetes y módulos npm (Nodo Package Manager) que permite reutilizar código y compartirlo.

Nodojs -NPM

Npm(node Package Manager) es el gestor de paquetes predeterminado para Nodo js

Es una herramienta que nos permite descargar, instalar y administrar fácilmente paquetes de código reutilizable para nuestros proyectos.

Características principales:

--Repositorio de paquetes: cuenta con un repositorio en línea con miles de paquetes de código abierto.

--instalación sencilla: permite instalar paquetes específicos o dependencias de proyectos mediante comandos simples.

--Manejo de dependencias: administra automáticamente las dependencias de un proyecto y asegura que las versiones correctas estén instalados.

--Scripts personalizados: permite ejecutar scripts personalizados definidos en el archivo “package.json” de un proyecto.

--Actualizaciones regulares: se actualiza periódicamente para agregar nuevas funciones y mejoras.

Para que usamos npm?

Se utiliza en línea de comandos como:

Para iniciar el uso de npm en un proyecto***: >npm init -y*** de esta manera se generar un archivo “package.json” que almacena información sobre el proyecto y sus dependencias.

Para instalar dependencias necesarias: ***>npm install es igual a npm i***

Para buscar paquetes en el repositorio de npm y utilizar ***>npm install <nombreDelPaquete>*** para descargar e instalar en tu proyecto.

Paquete chance

Chance es una biblioteca, que genera datos aleatorios de manera realista y controlada. Puede ser utili para simular datos en las aplicaciones y para demostrar como trabajas con datos aleatorios.

Para importar bibliotecas:

Se hace atraves de **const Chance = require(‘chance’); require es importar**

Creo una variable del tipo Chance **const chance = new Chance();**

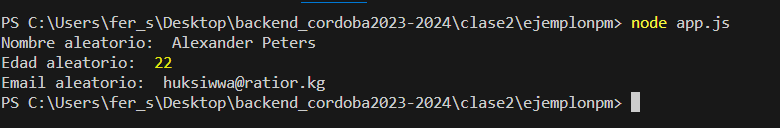
Utilizo las funciones que están en el paquete chance

const randomNombre = chance.name();

const randomEdad = chance.age();

const randomEmail = chance.email();

para ejecutar tu código hay que escribir **node nombeCodigo.js**



Aa

Introducción a los Módulos de Node.js 20

● Los módulos de Node.js son una forma de organizar y reutilizar el código en aplicaciones de Node.js.

● Los módulos de Node.js se pueden cargar mediante la función require() y se exportan utilizando module.exports o exports.

Archivo cabecera o donde definición la funciones a importar se hace con

**Modulo.exports = {función};**

Texto

Descripción generada automáticamente

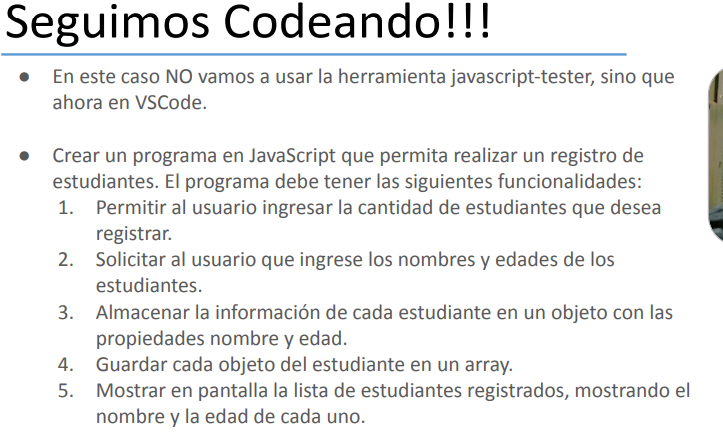
Archivo donde usanos el código del otro modulo : **const{función} =requiere(‘ruta./nombarchivo’);**

d

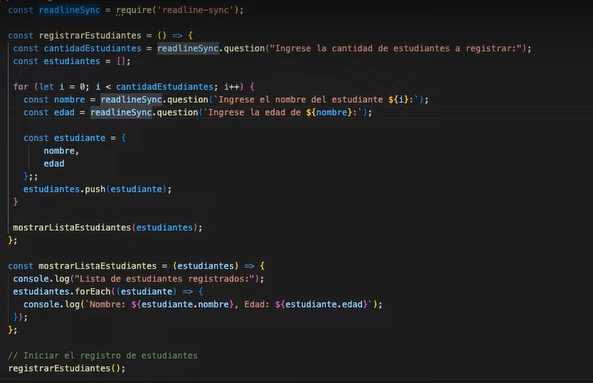
Texto

Descripción generada automáticamente

A



Instalamos el pquete readline-sync tiene una función que hace lo mismo que promt



PS C:\Users\fer\_s\Desktop\backend\_cordoba2023-2024\clase2\un\_ejemplo\_exportar> node calculadora.js

node:internal/modules/cjs/loader:1147

throw err;

^

Error: Cannot find module 'C:\Users\fer\_s\Desktop\backend\_cordoba2023-2024\clase2\un\_ejemplo\_exportar\calculadora.js'

at Module.\_resolveFilename (node:internal/modules/cjs/loader:1144:15)

at Module.\_load (node:internal/modules/cjs/loader:985:27)

at Function.executeUserEntryPoint [as runMain] (node:internal/modules/run\_main:135:12)

at node:internal/main/run\_main\_module:28:49 {

code: 'MODULE\_NOT\_FOUND',

requireStack: []

}

Node.js v20.10.0