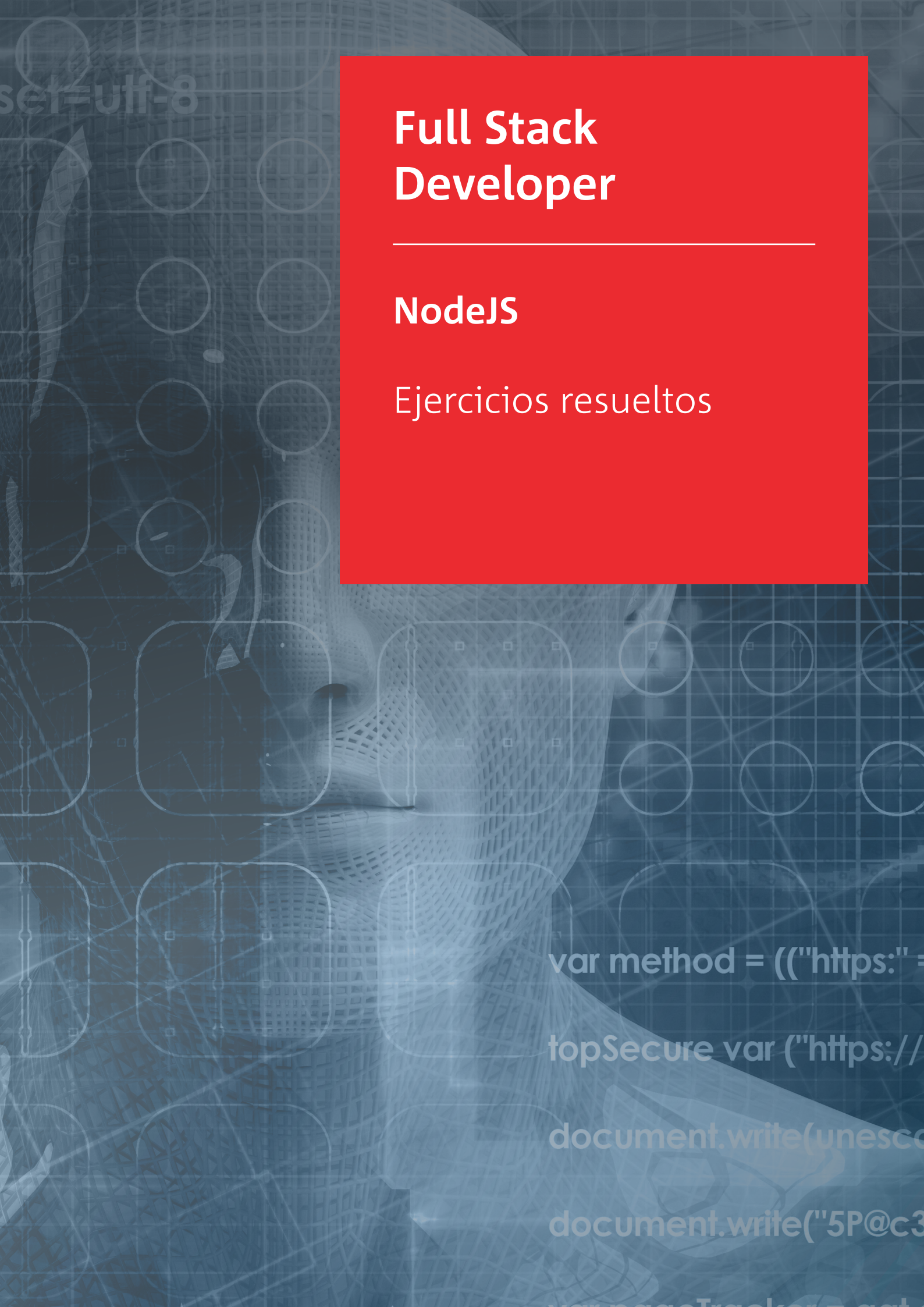

Full Stack Developer

NodeJS

Ejercicios resueltos

NodeJS

Ejercicio 1

Para comprobar la generación de un proyecto NodeJS y el uso de la funcionalidad de módulos, crea un proyecto en un directorio denominado ejercicio1; usando Visual Studio Code, deberás crear la siguiente estructura de archivos y directorio:

- **Un archivo app.js.**
- **Un directorio denominado utils.**
- **Un archivo age.js dentro del directorio utils.**
- **Las dependencias de NodeJS generadas al inicializar un proyecto.**

A continuación, desarrolla los siguientes pasos:

- **Genera un proyecto NodeJS con la herramienta de línea de comandos.**
- **Instala como dependencia de desarrollo la herramienta nodemon.**
- **Crea un *script* en el archivo package.json para ejecutar el proyecto con nodemon.**
- **Crea un archivo .gitignore para que el directorio node_modules no se controle en las versiones.**
- **Crea en el archivo age.js una función que, a partir de una fecha de nacimiento, devuelva la edad de una persona; luego, exporta la citada función.**
- **Importa la función anterior en el archivo app.js e impleméntala dentro de una lógica de tu elección de manera que se emplee y devuelva un resultado por consola.**
- **Ejecuta el proyecto con el *script* y comprueba los resultados.**

Ejercicio 2

Para comprobar la generación de un proyecto NodeJS y Express, y el uso de la funcionalidad de envío de archivos como respuesta a peticiones HTTP, crea un proyecto en un directorio denominado ejercicio2; usando Visual Studio Code, deberás crear la siguiente estructura de archivos y directorio:

- **Un archivo app.js.**
- **Un directorio denominado files.**
- **Un archivo info.txt dentro del directorio files.**
- **Cualquier archivo PDF que haya que introducir dentro del directorio files.**
- **Las dependencias de NodeJS generadas al inicializar un proyecto.**

A continuación, desarrolla los siguientes pasos:

- **Genera un proyecto NodeJS con la herramienta de línea de comandos.**
- **Instala como dependencia de desarrollo la herramienta nodemon.**
- **Instala como dependencia la librería Express.**
- **Crea un *script* en el archivo package.json para ejecutar el proyecto con nodemon.**
- **Crea un archivo .gitignore para que el directorio node_modules no se controle en las versiones.**
- **Crea en el archivo info.txt unos textos que posteriormente puedan ser comprobados.**
- **Ubica cualquier archivo PDF en el directorio files.**
- **Implementa la sintaxis de servidor Express en el archivo app.js.**

- **Crea dos métodos “get” para dos rutas diferentes que devuelvan cada archivo del directorio files.**

- **Ejecuta el proyecto con el script y comprueba los resultados.**

Ejercicio 3

Para comprobar la implementación de una lógica de respuestas a peticiones HTTP en NodeJS y Express, crea un proyecto en un directorio denominado ejercicio3; usando Visual Studio Code, deberás crear la siguiente estructura de archivos y directorio:

- **Un archivo app.js.**
- **Las dependencias de NodeJS generadas al inicializar un proyecto.**

A continuación, desarrolla los siguientes pasos:

- **Genera un proyecto NodeJS con la herramienta de línea de comandos.**
- **Instala como dependencia de desarrollo la herramienta nodemon.**
- **Instala como dependencia la librería Express.**
- **Crea un script en el archivo package.json para ejecutar el proyecto con nodemon.**
- **Crea un archivo .gitignore para que el directorio node_modules no se controle en las versiones.**
- **Implementa la sintaxis de servidor Express en el archivo app.js.**
- **Crea un método "get" que reciba un parámetro y evalúe su valor.**
- **Genera al menos dos respuestas diferentes según el valor del parámetro.**
- **Ejecuta el proyecto con el *script* y comprueba los resultados.**