

Aplicación DH Pizzas - Versión 1 - Paso I

- A. Lo primero que te vamos a pedir es que crees una carpeta llamada **dh-pizzas** en cualquier lugar de la computadora. Si bien tenés libertad de hacerlo en cualquier lugar, por buenas prácticas te recomendamos tener una alta prolijidad y la creés en un lugar donde generalmente guardés cosas de tu trabajo.
- B. Luego de que tengas creada la carpeta, vas a tener que ingresar a ella desde la terminal.

- C. Cuando estés dentro de la carpeta deberás ejecutar este comando **npm init -y**

- a. **¿Qué hace este comando?**

R/ **npm init -y**, inicializa un proyecto de Node en la carpeta. Todos los proyectos generados con Node debe comenzar con este comando, el cual ejecutarás una sola vez.

Al momento en que ejecutés este comando, te darás cuenta que se creó un archivo llamado **package.json**. ¿Te animás a consultar en google para qué sirve este archivo?

- D. **Puesta en común grupal:** Debatamos entre compañeras/os y docentes la importancia del archivo package.json.

- E. Continuando con la ejercitación, ahora te pediremos que ejecutes en la terminal el siguiente comando: **npm install inquirer --save**. ¿Después de ejecutarlo podés decirnos qué pasó?

- F. **Puesta en común de a pares:** Debatí con la persona de al lado qué pensás que sucedió después de ejecutar este comando. Apresúrense para que sean la primer dupla en comentarle al docente su hipótesis.

- G. Aclaradas las cosas vamos a continuar. Para este punto vas a tener que crear un archivo llamado **cliente.js**. Crear dentro de dicho archivo un **console.log** que diga "**Soy el archivo cliente.js**" (esto lo harás como una forma de testear lo que vas haciendo). Ejecutá el archivo desde la terminal con Node y fijate qué sucede.

- H. Si todo salió bien vamos a continuar, si no, pedile ayuda a algún docente o decile a tu colega de al lado que te dé una mano. Una vez listo esto. Volveremos a trabajar sobre el archivo **cliente.js**. Esta vez te pediremos que elimines el *console.log* y que ahora escribas esta línea:

```
const inquirer = require("inquirer");
```

- a. **¿Qué hace esta línea de código?**

R/ te invitamos a que investigués en google *¿qué hace la función require de node?*. Una vez tengás la respuesta vamos a hacer una puesta en común.

- I. **Puesta en común grupal:** Debatamos entre compañeras/os y docentes para qué sirve la función require.

- J. Entendiendo ahora para qué sirve la función require, te invito a leer tu primer documentación.

Sabiendo que inquirer es un paquete / librería de Node, vas a ir a la [documentación oficial](#) de esta librería y por tu cuenta vas a tratar de definir para qué sirve. Tenés 15 minutos para sacar rápidamente tus conclusiones.

- K. **Puesta en común grupal:** Debatamos entre compañeras/os y docentes para qué sirve el paquete inquirer y en qué nos servirá dentro de esta ejercitación.

- L. **Live Coding Guiado:** Ahora que ya comprendés las bases de este paquete, vamos a comenzar a ponerlo en funcionamiento. Para ello te invito a que sigas a tu docente y hagás lo mismo que ella o él hace en pantalla.

- M. Ahora es tu turno de brillar por cuenta propia. O con tu colega de al lado ¿por qué no? Tu objetivo será crear un *array* con 3 *objetos literales*, cada uno representará a pregunta así:
- Ingresá tu nombre → type: "input"
 - Ingresá tu número de teléfono → type: "input"
 - Elegí el gusto de la pizza → type: "rawlist" (con al menos 4 opciones)

Después, entregue estas preguntas al método **prompt** de **inquirer** y hace un **console.log** de las respuestas dentro de la función que va dentro del método **then**. **Importante:** Si tenés preguntas de prueba que hiciste en el bloque anterior, eliminalas y dejá tu código lo más prolijo posible 😊

- N. Si todo lo del punto anterior funcionó, seguí adelante. Si no, solicitá apoyo a tu colega de al lado o a algún docente para destrabarte.

Una vez con el camino limpio. Te pedimos que agregues otras tres preguntas al **array** del paso anterior. De la misma manera que lo hiciste. Cada una de ellas representará una pregunta así:

- Elegí el tamaño de la pizza → type: "list" (Personal, Mediana Grande)
- Elegí el gusto de la bebida → type: "list" (con al menos 4 opciones)
- ¿Qué gusto de empanadas querés? → type: "checkbox" (con al menos 6 opciones)

Con esto hecho, probá ejecutando con Node el archivo **cliente.js** desde la terminal y observa el resultado obtenido.

¿Tenés algún error de ejecución? No te preocupes, pedile a algún docente que venga y te dé una mano para que todo ande "10 puntos".

O. Puesta en común grupal: Debatamos entre compañeras/os y docentes qué hemos hecho hasta el momento y cuales son los fallos más frecuentes que observaste se pueden dar.

P. Live Coding Guiado: vamos a comenzar a ponerle más "chiches" a la aplicación. Para ello, te invito a que sigas a tu docente y hagás lo mismo que ella o él hace en pantalla.

- Q. Ahora con muchas más herramientas para implementar en la aplicación, se deberá sumar al array de preguntas, las siguientes (tené cuidado de organizarlas de una manera lógica):
- ¿La pizza es para llevar? → type: "confirm", default: false
 - Ingresá tu dirección → type: "input". Se dispara **cuando** la pregunta anterior se marca como verdadera. **Validar** que el campo no quede vacío. Si está vacío, mostrar un mensaje que diga "Dejanos saber tu

dirección para llevarte la pizza." La aplicación no debe seguir su ejecución si este campo está vacío.

- c. ¿Querés agregar una bebida? → type: "confirm", default: false
- d. ¿Sos cliente habitual? → type: "confirm", default: false

Con esto hecho, probá ejecutando con Node el archivo **cliente.js** desde la terminal y observa el resultado obtenido.

¿Tenés algún error de ejecución? No te preocupes, pedile a algún docente que venga y te dé una mano o quizás preguntale a tu colega de al lado y traten de encontrar la solución 🍋.

R. Ya la aplicación tiene las preguntas necesarias. Hasta el momento las mismas serán:

- a. ¿La pizza es para llevar?
- b. Ingresá tu dirección
- c. Ingresá tu nombre
- d. Ingresá tu número de teléfono
- e. Elegí el gusto de la pizza
- f. Elegí el tamaño de la pizza
- g. ¿Querés agregar una bebida?
- h. Elegí el gusto de la bebida
- i. ¿Sos cliente habitual?
- j. ¿Qué gusto de empanadas querés?

Si tenés todo bien, te invitó a continuar, de lo contrario hace los ajustes necesarios para llegar a dicho punto.

S. Para terminar, a las preguntas anteriores les vamos a agregar métodos de **validación** y a su vez métodos que definan **cuando** se dispara cada pregunta. Por lo tanto la lógica de cada pregunta deberá implementarse así:

- a. ¿La pizza es para llevar? *(confirmación, Y o N)*
- b. Ingresá tu dirección *(se dispara si la anterior es verdadera - No puede quedar vacía)*
- c. Ingresá tu nombre *(no puede quedar vacía)*
- d. Ingresá tu número de teléfono *(no puede quedar vacía - debe admitir solo números)*
- e. Elegí el gusto de la pizza *(por defecto debe ser Muzzarella)*
- f. Elegí el tamaño de la pizza *(por defecto debe ser Grande)*
- g. ¿Querés agregar una bebida? *(confirmación, Y o N)*

- h. Elegí el gusto de la bebida *(se dispara si la anterior es verdadera - por defecto debe ser Coca-cola)*
 - i. ¿Sos cliente habitual? *(confirmación, Y o N)*
 - j. ¿Qué gusto de empanadas querés? *(se dispara si la anterior es verdadera - se debe poder elegir solo 3 gustos)*
- T. Con todo esto realizado. Probá ejecutando la aplicación desde la terminal y observa los resultados en la misma.

U. **Puesta en común grupal:** Debatamos entre compañeras/os y docentes cuál debería ser el resultado final de la aplicación hasta el momento.

👉 Fin de la práctica 👈