En un archivo de texto separado que debes crear, escribe explicaciones de los siguientes conceptos como si se lo estuvieras explicando a un niño de 12 años. Hacer esto te ayudará a descubrir rápidamente cualquier agujero en tu comprensión.

**-Variables**

Una variable es una forma de guardar información para usar en otro momento, como si fueran una caja. A estas cajas se les pueden escribir un nombre, entonces ya sabés que adentro de la caja que dice "Conejo" hay un conejo, en la que dice "Instrucciones" hay una lista de pasos para desarmar tu computadora. Entonces cada vez que necesitas usar esa información (que puede ser mucha, por ejemplo: ¡100 ingredientes y 50 pasos para cocinar!) no necesitas buscarlo uno por uno, sino que agarras la caja directamente.

**-Strings**

La traducción de este nombre es "Hilos", eso significa que un "string" es un hilo de caracteres, es decir, uno atrás de otro (incluyendo letras, números, símbolos, espacios, etc.)

**-Funciones (argumentos, return)**

Usando de vuelta la ilustración de las cajas, las funciones sirven para establecer qué queremos hacer con las cajas y en que momento. Ahí escribimos paso a paso lo que va a suceder con nuestras variables, puede ser: sumarlas, dividirlas, cambiarlas, agregarles información. Estas funciones recuerdan todos esos pasos y pueden hacerlos automaticamente cuando decimos su nombre (más tecnicamente, cuando las invocamos), por eso es muy importante ponerle un nombre que refleje lo que hace.

Los argumentos son variables, es decir, cosas que pueden cambiar. Vamos a un ejemplo: Depende de cuantos juguetes tiene o no la caja, decido si la guardo en el garage o la dono. Ahí nuestra variable sería: la cantidad de juguetes que entren en la caja. Es información crucial que necesitamos saber para decidir que decisión tomar. Como nuestras funciones siguen pasos automáticamente podemos especificarles al utilizar una variante que dependiendo lo que suceda con ese asunto es la decisión que tendrá que tomar.

La sentencia return tiene una doble función. Por un lado hace que la función detenga su funcionamiento y, por el otro lado, permite que recibamos alguna información acerca de lo que sucedió o como señal de que la función ha cumplido con su tarea. En este caso pensemos en la función como una persona, hizo todo su trabajo (apiló 3 cajas del mismo color) y tranquilamente podría irse a su casa (porque ya terminó de hacer lo que debía). En cambio, el jefe puede decirle: "Cuando termines avisame y decime: 'Pude apilar 3 cajas azules'". Esa es la segunda función del "return": hacer que la función nos de información cuando termina su deber.

**-Declaraciones if.**

Esta herramienta se entiende facilmente si nos ponemos en el papel del intérprete de JS. Si pasa "x" cosa, se realiza el codigo. Por ejemplo: Si la persona se llama "Lucas", imprimo en pantalla "Ese nombre en inglés es Luke". En el caso de que la persona no se llame "Lucas", no se ejecuta la orden.

Para determinar, siguiendo el ejemplo, qué va a suceder si la persona no se llama "Lucas", utilizaríamos "else" (traducido como "si no") donde estableceriamos que orden debe ejecutarse en caso de que la persona no se llame "Lucas".

**-Valores booleanos (true, false)**

Los valores booleanos son "verdadero" y "falso". Si, así de simple. Pero se pueden realizar muchísimas cosas con ellos y puede ser realmente difícil, solo decirte que hay toda un sistema matemático para aprender a usarlos. Los booleanos son los que componen el sistema binario (ceros y unos, activar y desactivar).