



# INTRODUCCION A LA PROGRAMACION 2025-3

Retos 9 al 20

Luis Llumiquinga  
6 de diciembre de 2025

Ing. Santiago Mosquera  
MOVILIS

## Resumen

El presente informe recopila las soluciones y aprendizajes obtenidos durante el desarrollo de los retos 9 al 20 del curso “*Krakedev: Primeros Pasos*”. Los ejercicios abarcan actividades de corrección de errores, creación de funciones, manipulación del DOM, validaciones, cálculos matemáticos, conversión de unidades, aplicación de estilos y depuración de código en JavaScript. Además, se incluye evidencia conceptual del uso de HTML, CSS y JavaScript, así como reflexiones personales sobre el proceso de aprendizaje. Este conjunto de retos permitió fortalecer habilidades básicas de programación, promover buenas prácticas y mejorar la comprensión del ciclo de desarrollo web.

**Palabras clave:** JavaScript, HTML, CSS, funciones, depuración, programación web.

---

## Introducción

El curso “*Krakedev: Primeros Pasos*” está diseñado para introducir al estudiante en la programación web mediante ejercicios prácticos que integran conceptos básicos de HTML, CSS y JavaScript. Los retos 9 al 20 abarcan actividades progresivas destinadas a reforzar habilidades técnicas mediante la resolución de problemas reales de programación. Este informe presenta el desarrollo de cada reto, las soluciones implementadas y los aprendizajes adquiridos. Se sigue el formato APA séptima edición, estructurando los hallazgos en un documento claro y ordenado que refleja el avance alcanzado durante aproximadamente 200 minutos de trabajo.

---

## Desarrollo

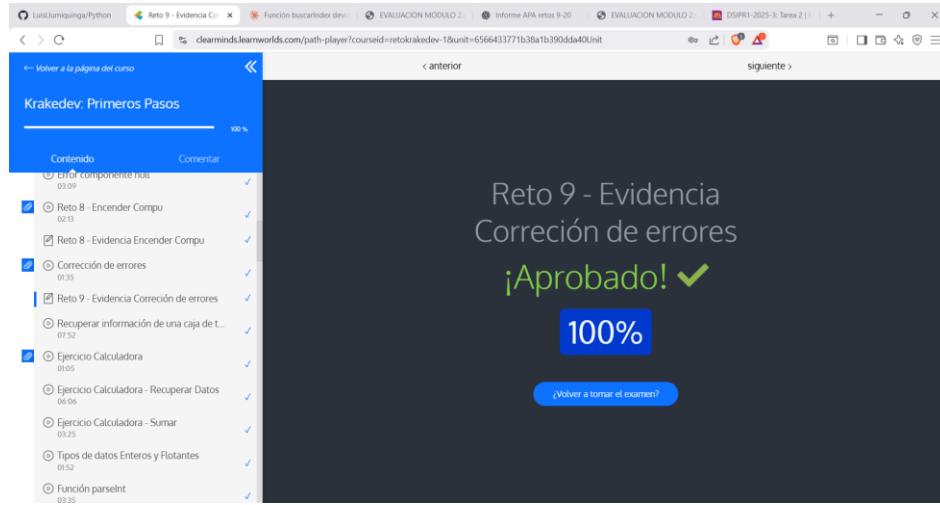
### Reto 9: Corrección de errores

Este reto consistió en identificar y corregir errores de sintaxis, rutas incorrectas, omisiones de etiquetas y fallos de lógica en un archivo de código. La actividad reforzó la capacidad de analizar mensajes de error y depurar paso a paso.

**Aprendizaje clave:** comprender la importancia de revisar la consola del navegador y validar la estructura del código para evitar fallos de ejecución.

**Figura 1**

Evidencia corrección de errores



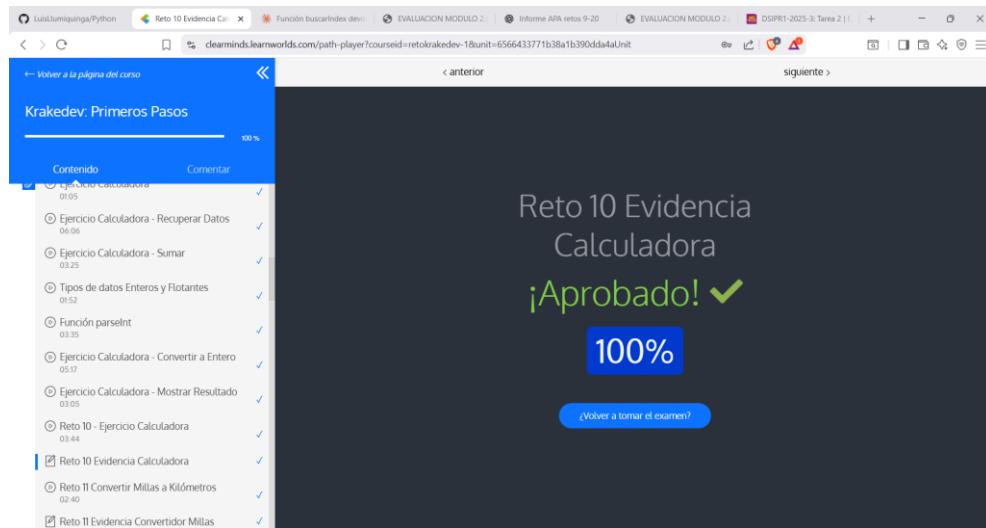
## Reto 10: Ejercicio calculadora

Se desarrolló una calculadora básica capaz de realizar operaciones aritméticas como suma, resta, multiplicación y división. Se utilizaron entradas de texto, botones y funciones en JavaScript para realizar los cálculos.

**Aprendizaje clave:** uso de funciones y eventos para procesar datos numéricos.

**Figura 2**

Evidencia calculadora



## Reto 11: Convertir millas a kilómetros

Se programó una función que recibe un valor en millas, lo convierte a kilómetros multiplicando por 1.60934 y muestra el resultado en pantalla.

**Aprendizaje clave:** aplicar conversiones matemáticas simples y trabajar con inputs numéricos.

**Figura 3**

Evidencia convertidor millas

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "Reto 11 Evidencia Convertidor Millas". The main content area displays the message "¡Aprobado! ✓" and "100%". To the left, there is a sidebar with a list of items, each with a checkmark indicating completion. The items listed are:

- ① Función parseInt 03:35
- ② Ejercicio Calculadora - Convertir a Entero 05:17
- ③ Ejercicio Calculadora - Mostrar Resultado 03:05
- ④ Reto 10 - Ejercicio Calculadora 03:44
- ⑤ Reto 10 Evidencia Calculadora
- ⑥ Reto 11 Convertir Millas a Kilómetros 02:40
- ⑦ Reto 11 Evidencia Convertidor Millas
- ⑧ Reto 12 Convertir Celsius a Fahrenheit 02:33
- ⑨ Reto 12 Evidencia Convertidor Celsius
- ⑩ Repaso 05:23
- ⑪ Autoevaluación 5-1

## Reto 12: Convertir Celsius a Fahrenheit

La tarea consistió en implementar una fórmula de conversión:

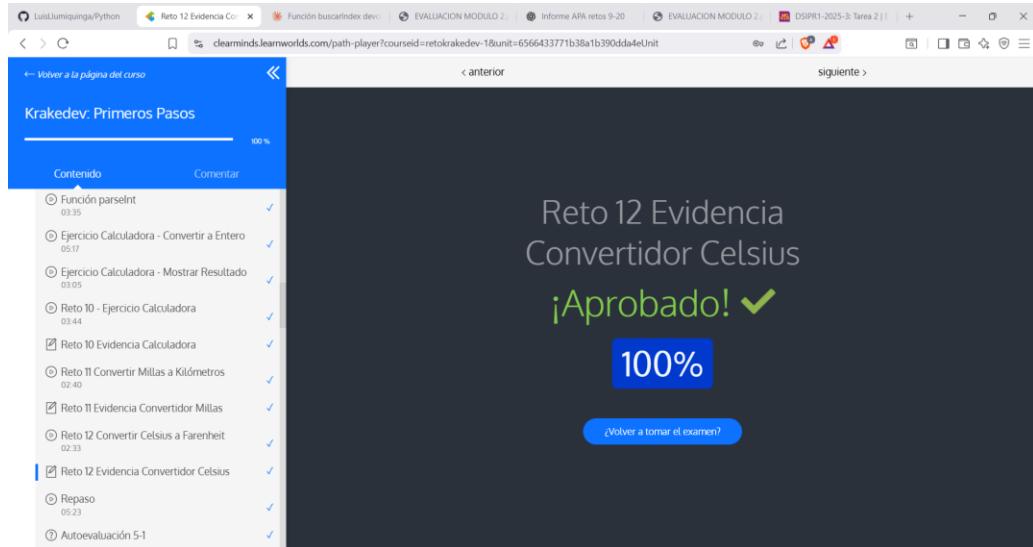
$$^{\circ}\text{F} = (^{\circ}\text{C} \times 9/5) + 32.$$

El resultado se mostró dinámicamente en la página.

**Aprendizaje clave:** trabajar con fórmulas matemáticas y manipulación del DOM.

**Figura 4**

Evidencia convertidor Celsius



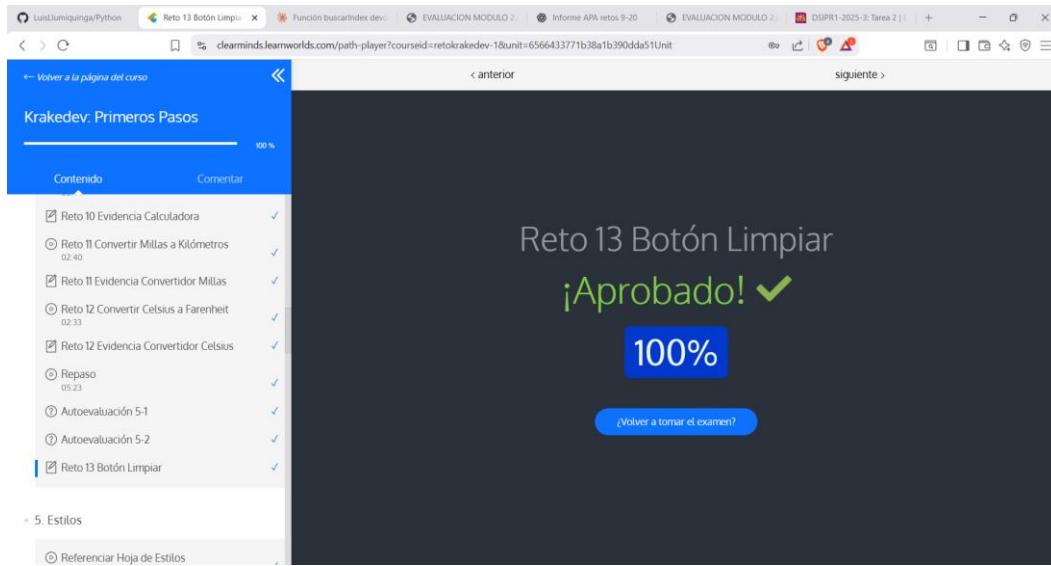
### Reto 13: Botón limpiar

Se añadió un botón capaz de limpiar los campos de entrada y reiniciar el resultado.

**Aprendizaje clave:** restablecer estados del formulario mediante JavaScript.

**Figura 5**

Evidencia botón limpiar



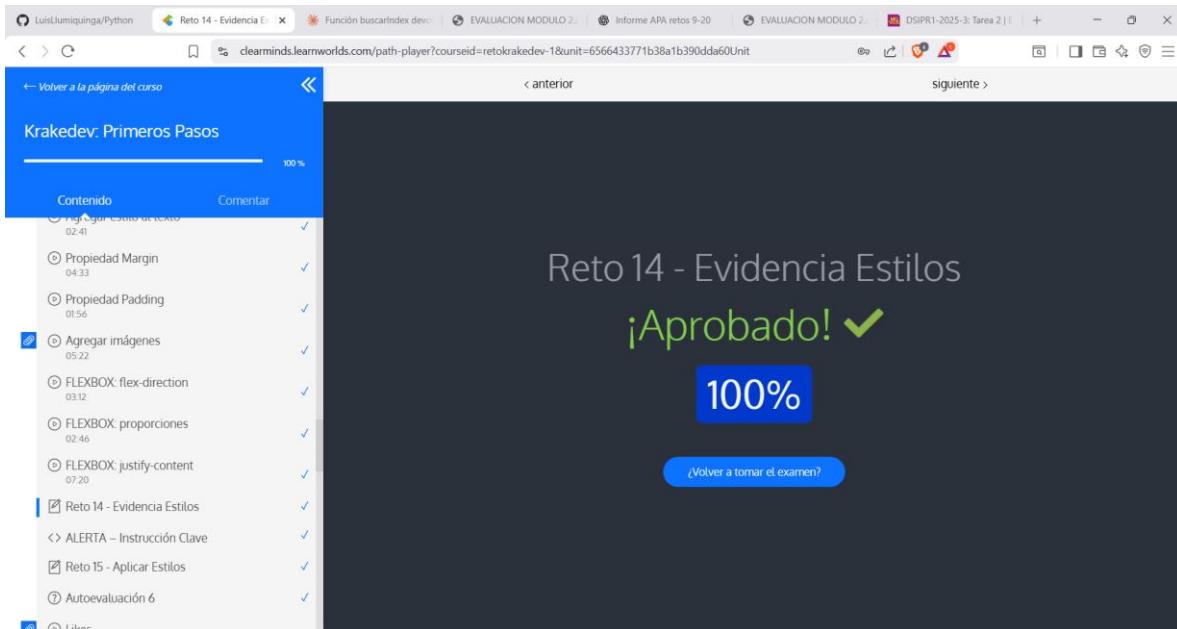
## Reto 14: Evidencia de cambios en estilos.html

Se realizó la modificación de estilos en la página *estilos.html*, aplicando un fondo personalizado en el elemento *body* y cargando evidencia visual mediante capturas.

**Aprendizaje clave:** uso de propiedades CSS y personalización de interfaces web.

**Figura 6**

Evidencia estilos



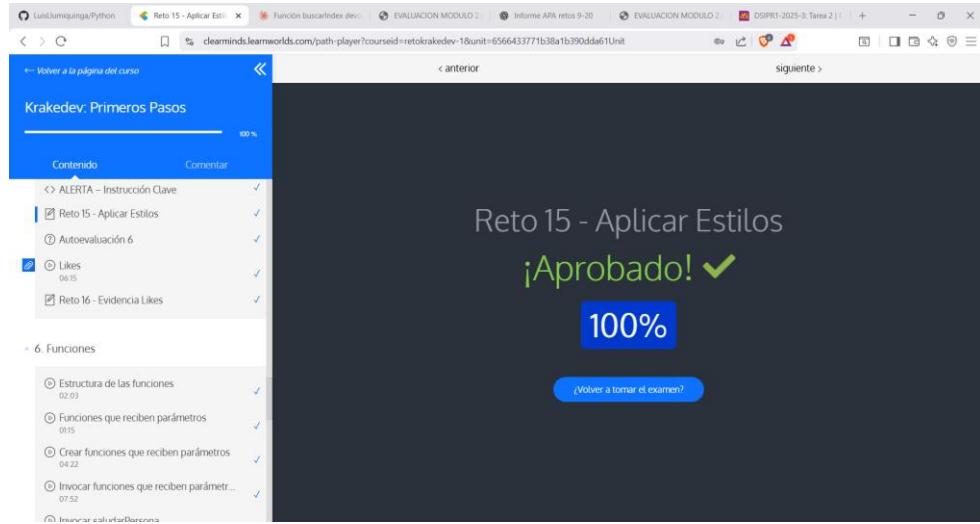
## Reto 15: Estilos para el convertidor de millas a kilómetros

Se añadió una hoja de estilos CSS para mejorar la presentación del convertidor creado en el reto 11. La evidencia incluyó el código HTML, CSS y la página visual desplegada.

**Aprendizaje clave:** separación entre estructura (HTML), estilo (CSS) y lógica (JS).

**Figura 7**

Evidencia aplicar estilos



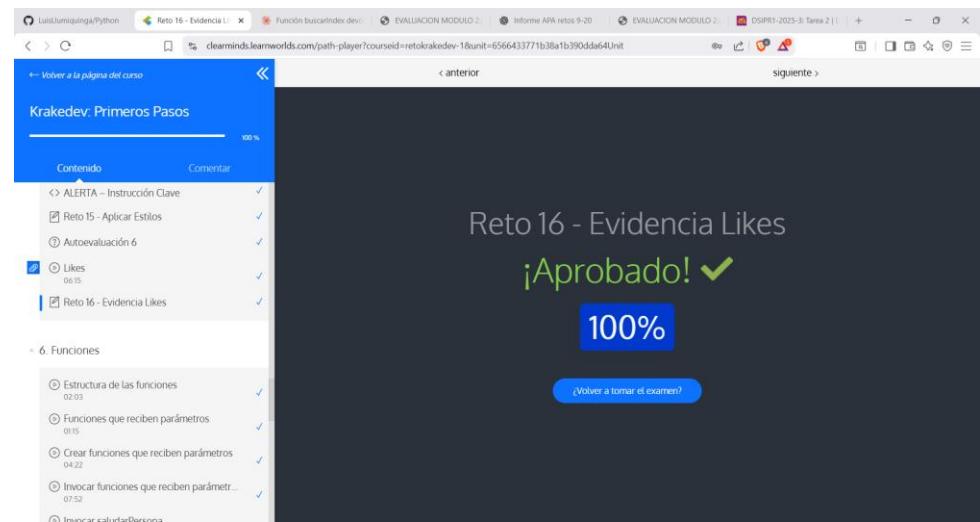
### Reto 16: Evidencia de likes

Se implementó un contador de “likes”, aumentando el valor al hacer clic en un botón. Se presentó evidencia visual del funcionamiento.

**Aprendizaje clave:** manejo de eventos y modificación dinámica del contenido.

**Figura 8**

Evidencia Likes



## Reto 17: Funciones en JavaScript

Este reto incluyó varias actividades:

1. Crear una función saludarAmigo(nombre, apellido, apodo) que mostrara en consola los parámetros concatenados.
2. Agregar una caja de texto adicional en HTML para el apellido.
3. Crear la función testSaludarAmigo() que recuperara los valores de las cajas de texto y llamarla a la función principal.
4. Agregar un botón para activar la función y mostrar evidencia del debug.

**Aprendizaje clave:** depuración de funciones, paso de parámetros y conexión entre HTML y JS para entradas del usuario.

**Figura 9**

Evidencia crear funciones

The screenshot shows a browser window with multiple tabs open at the top. The main content area displays a challenge titled "Reto 17 Evidencia Crear Funciones". On the left, there is a sidebar with a list of tasks, each with a checkmark indicating completion. The tasks include creating functions with parameters, invoking them, and completing specific challenges like "ALERTA - Instrucción Clave" and "Retardo 17 Evidencia Crear Funciones". To the right of the sidebar, the main content area shows the challenge title, a large green message "¡Aprobado!" with a checkmark, and a blue button labeled "100%". At the bottom right of the content area, there is a small link "¿Volver a tomar el examen?".

---

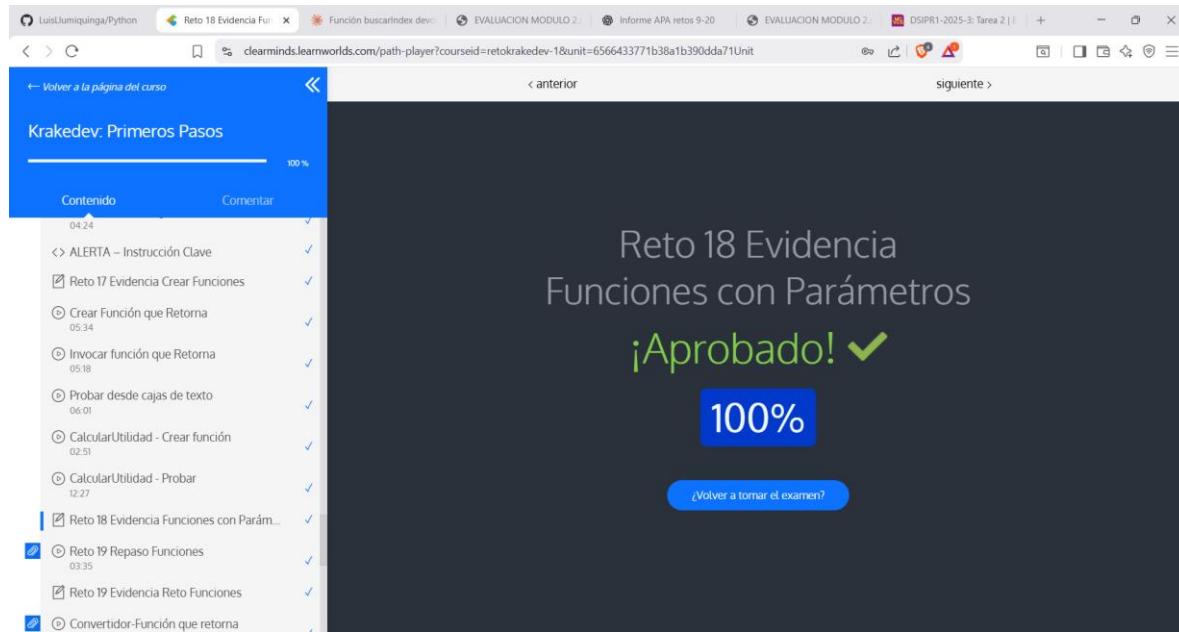
## Reto 18: Funciones con parámetros

Se desarrollaron ejercicios donde funciones recibían datos y devolvían resultados específicos. Se fortaleció el uso de argumentos, retornos y estructura modular.

**Aprendizaje clave:** reutilización de código mediante funciones bien definidas.

**Figura 10**

Evidencia funciones con parámetros



### Reto 19: Reto funciones

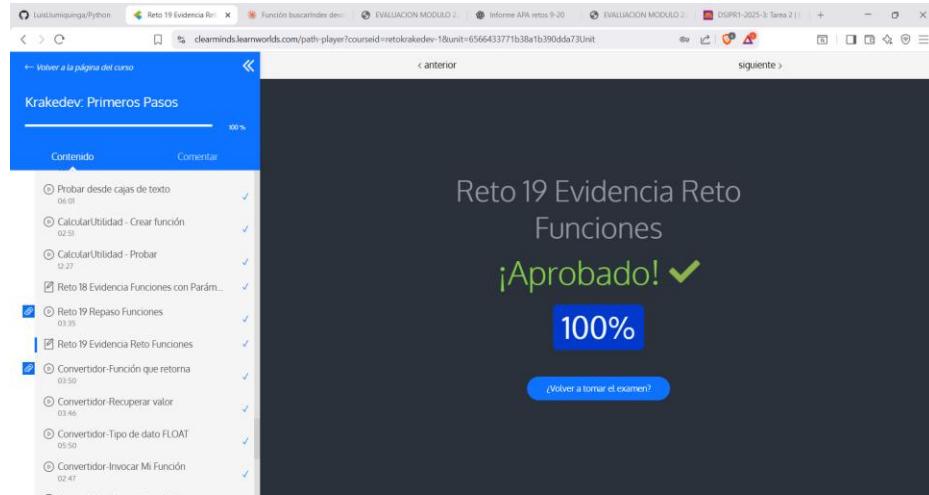
Se resolvieron problemas adicionales relacionados con funciones, reforzando conceptos como:

- uso de funciones puras
- validación de entradas
- lógica condicional
- organización del código

**Aprendizaje clave:** la importancia de estructurar el código en funciones para mejorar la legibilidad y mantenimiento.

**Figura 11**

Evidencia reto funciones



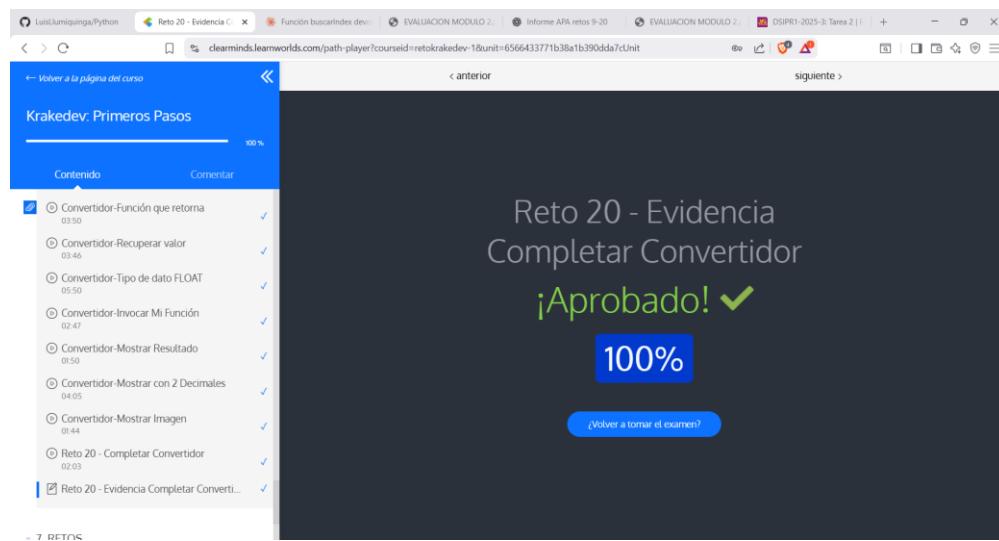
## Reto 20: Convertidor a euros y pesos colombianos

Se programó un convertidor que toma un valor en dólares y calcula su equivalente en euros y pesos colombianos usando tasas predefinidas.

**Aprendizaje clave:** manejo de múltiples resultados, formateo de números y comprensión de conversiones internacionales.

**Figura 12**

Evidencia completar convertidor



## Conclusiones

El desarrollo de los retos 9 al 20 permitió fortalecer habilidades fundamentales en programación web, especialmente en JavaScript. Se aprendió a depurar errores, aplicar fórmulas matemáticas, manipular elementos del DOM, diseñar interfaces con CSS y estructurar funciones reutilizables. Además, se reforzaron buenas prácticas como modularizarían, separación de responsabilidades y verificación mediante depuración. Estos retos contribuyeron significativamente al dominio de la lógica básica de programación y al entendimiento del ecosistema web, sentando bases sólidas para continuar con módulos más avanzados.

## Referencias

Mosquera, S. (01 de 2023). *Clearminds*. Obtenido de Clearminds KrakeDev:  
<https://clearminds.learnworlds.com/path-player?courseid=retokrakedev-1&unit=6566433771b38a1b390dda7cUnit>