

## Proyecto Práctico de TypeScript: Calculadora de Gastos Personales

### Objetivo:

Desarrollar una calculadora de gastos personales utilizando TypeScript. Aplicar los conceptos básicos de TypeScript mientras crean una herramienta útil para el seguimiento y gestión de gastos personales.

### Instrucciones:

#### Funcionalidades requeridas:

- Registro de Gastos: Permitir al usuario registrar nuevos gastos ingresando el nombre del gasto y su respectivo monto.
- Categorización de Gastos: Permitir al usuario categorizar los gastos en diferentes categorías (por ejemplo: alimentos, transporte, entretenimiento, etc.).
- Visualización de Gastos: Mostrar al usuario una lista de los gastos registrados, junto con su categoría y monto.
- Cálculo de Total: Calcular y mostrar al usuario el total de gastos registrados.
- Gráficos Estadísticos: Mostrar gráficos estadísticos que representen la distribución de los gastos por categoría.
- Filtrado de Gastos: Permitir al usuario filtrar los gastos por categoría o rango de fechas.
- Guardar en LocalStorage: Almacenar los gastos registrados en el almacenamiento local del navegador para persistencia de datos.

#### Interfaz de Usuario:

- Diseñar una interfaz de usuario intuitiva y atractiva para las funcionalidades mencionadas.
- Utilizar HTML y CSS para crear la estructura y estilos de la interfaz.

#### Implementación en TypeScript:

- Utilizar TypeScript para manejar la lógica de cálculo y actualización de la interfaz de usuario.
- Definir tipos de datos adecuados para los gastos, categorías y otros elementos necesarios.

- Organizar el código de forma modular y legible.

#### Requerimientos Técnicos:

- Utilizar TypeScript para todo el código del proyecto.
- Utilizar el almacenamiento local del navegador (LocalStorage) para persistir los datos de los gastos.
- No se permite copiar y pegar código de soluciones existentes en Internet, como tutoriales de YouTube o proyectos de otros desarrolladores.
- Se permite buscar y aprender de recursos en línea, pero el proyecto final debe ser original y reflejar el entendimiento y habilidades del estudiante.

#### Entrega y Evaluación:

- El proyecto debe ser entregado como un repositorio de GitHub público.
- Se evaluará la calidad del código, la implementación de las funcionalidades requeridas, la usabilidad de la interfaz de usuario y la efectividad en el manejo del almacenamiento de datos.
- La entrega debe incluir un archivo README.md con instrucciones claras para ejecutar la aplicación localmente.

**Fecha de Entrega: 04/04/2024**

**Entrega vía campus.**