

# Analizador

Es acá en donde el archivo json es procesado a través de un analizador lexico, encargado de verificar cada token perteneciente al lenguaje y almacenando aquellos que no lo son, como un conjunto de errores.

## Funciones

- tokenize\_input
- tokenize\_number
- tokenize\_string

Para poder analizar dicho contenido era necesario una serie de validaciones tales como verificar si era un número, un espacio en blanco, un símbolo en específico o una cadena de texto. Por ello se crean estas funciones que ayudaran a verificarlo cuando todo se complemente en.

```
def tokenize_input(input_str):
```

Cada token que no sea perteneciente a lo anterior será denominado como un error y se trasladará a un diccionario anidado que almacenara los errores con sus respectivas especificaciones.

```
numero = lenerrores) + 1
    error = { #diccionario para los errores
        "numero": str(numero),
        "descripcion": { #diccionario anidado
            "lexema": char,
            "tipo": "error lexico",
            "columna": col,
            "fila": line,
        },
    }
    errores.append(error)
```

La función encargada de tomar las instrucciones, es decir las operaciones a realizar.

```
def get_instruccion():
```

A través del método pop(0) es posible eliminar el : y así poder tomar el siguiente y atribuirlo a la variable necesaria siendo esta operación, valor 1 o valor 2 sino mostrara el error correspondiente.

Luego de esto se crearan instancias provenientes de la clase expresión según la opción, si son operaciones aritméticas o trigonométricas

```
def create_instructions():
```

A traves de esta funcion se invocan las funciones necesarias para la creacion del grafico, dando asi las especificaciones para su diseño.

```
def analizar(entrada):
```

Gracias a esta funcion es posible crear el grafico ya con las respecticas configuraciones necesarias, tomando en cuenta su contenido y su diseño.

```
def mostrar_elementos():
```

Por medio de esta funcion se crea un listado con cada token, el cual contiene la informacion necesari como su columna, fila y su contenido.

```
def mostrar_errores():
```

Esta funcion es encargada de retornar el diccionario de los errores creado en la funcion analizador en la cual se almacena cada simbolo o caracter no perteneciente al lenguaje a analizar.

## Navegacion

- [Ir a la Sección de expresiones](#)
- [Ir a la Sección de la interfaz](#)
- [Ir a la Sección de graficas](#)