# Primer TP de Procesamiento de Lenguaje Natural

Autor: Martín Ezequiel Langberg

El siguiente trabajo tiene como objetivo entender el funcionamiento de un tokenizador y los problemas que presenta la segmentación. El mismo fue desarrollado en el lenguaje Python (ver `src/tokenizer.py`).

## Modo de Uso:

```
Python python tokenizer.py [Nombre del archivo] [Tipo de salida]
```

Si el tipo de salida es "A", el código crea un archivo con una línea por cada **token** y si es "P" sale por pantalla una línea por cada **token**.

## Implementación

Para implementar el tokenizador se decidió separar el procesos en diferentes etapas: Primero separar los tokens que contenían comillas, luego los guiones o barras y por último los espacios en blanco. Todo esto se realizó usando expresiones regulares.

La dificultad consistió en que a medida que se iban ampliando los criterios de división de tokens, muchos tokens que eran correctos pasaron a ser divididos por este nuevo criterio, por lo tanto fue necesario limitar dichos criterios en los anteriores mencionados dado que dieron los resultados más aceptables.

## Tests y Resultados

Durante el trabajo se usaron dos textos, uno de muestra para ir creando el tokenizador, y otro de test que fue utilizado finalmente para ver los resultado obtenidos y evitar confeccionar un tokenizador a la medida del texto muestra. Los resultados pueden verse en `\texts\text_muestra_tokenized` y `\texts\text_control_tokenized`.