# Las expresiones regulares (regex): parte 1

# ¿Qué son y para qué se utilizan?

- **¿Qué son las regex?**
  - Un lenguaje formal para describir patrones en un texto
  - Hemos visto los uso de “sed” para encontrar y reemplazar patrones simples. Regex nos permite buscar patrones mucho mas complejos (y por cierto se pueden usar con “sed” ;))
  - p. ej:
    - En nahuatl, los verbos toman los sufijos “-s” (sg) y “-squeh” (pl) en tiempo futuro. Se pueden escribir “-z”, “-zqueh”, “-skej”, “-squej”, “-squeh”, “-skeh”, etc. ¿Cómo buscar todas estas posibilidades?

# ¿Qué son y para qué se utilizan?

**[sz]((qu|k)e[jh])?\b**

# ¿Qué son y para qué se utilizan?

- Hay “variantes” en la sintaxis de las expresiones regulares (aunque se parecen mucho)
- Vamos a usar las regex de Python (lo cuál viene de la sintaxis regex del lenguaje Perl)
- Para jugar y probar con las expresiones regulares, vamos a usar la página https://pythex.org/

# Sintaxis básica

# Elementos principales

- **Literales:** caracteres que se representan sólos (no representan otra cosa)
  - Regex: “abc” coincide con una subsecuencia de la cadena “twe e rfwe i**abc**kl ñwef”
- **Clases de caracteres:** representan mas de un sólo carácter.
  - Normalmente en [] (representa un conjunto de los carácteres que van a dentro
  - Las vocales: “[aeiouáéíóúAEIOUÁÉÍÓÚ]”
  - Last letras de a-z: “[a-z]” A-Z: “[A-Z]”

# Elementos principales

- Metacaracteres: carácteres especiales qué modifican a otros caracteres o patrones en la búsqueda
  - ., +, *, ?, ^, $, \w, \W, \s, \S, \d, \D, etc

- Para buscar la forma literal de uno de estas caracteres como + o *, usa \ (\+)

# Metacarácteres: La repetición

- ***** (kleene star): coincide con 0 o más del carácter/patrón que lo precede
  - “ab*” encuentra las cadenas “a”, “ab”, “abb”, “abbbbbbbbb”
- **+** (kleene plus): 1 o más del carácter/patrón que lo precede
  - “ab+” encuentra las cadenas “ab”, “abb”, “abbbbbb”, pero **no** “a”
- **?** : el carácter/patron que lo precede es opcional (puede aparecer 0 o 1 vez)

# Metacaracteres: límites de una secuencia

- **^:** significa dos cosas distintas. Si está al principio de una clase de carácteres, quiere decir “ninguno de estos caracteres. Si no, quiere decir “el inicio de una línea”
- $: fin de línea
- \b: límite de una palabra

# Metacaracteres: otros símbolos especiales

- . cualquier carácter (p.ej. “alumn.s” coincide con alumnos/alumnas/alumnxs/alumnes)
- \w : carácter alfanumérico
- \W: no-alfanumérico
- \d: dígito
- \D: no dígito
- Etc… (más detalles en el cheatsheet o en pythex.org)

# Las agrupaciones

- Usando parenteses, puedes tratar una secuencia de carácteres como una sóla unidad:
  - “Hello(, world)?” coincide con la cadena “Hello, world” y “Hello”

- Para alternar (buscar mínimo una entre varios patrones), interpone “|” entre los patrones
  - “hello|bonjour” coincide con cualquier de los dos saludos.