

1. Desarrollar una expresión regular para encontrar todas las líneas que comienzan con la palabra "Start"

.
->^Start

2. Realizar una expresión regular para encontrar todas las líneas que terminan con la palabra "end".

->(end\$)

3. Realizar una expresión regular para encontrar la palabra " whole" solo cuando es una palabra completa.

->\bwhole\b

4. Realizar una expresión regular para encontrar la palabra "foo" solo si es seguida por "bar"

->foo(=bar)

5. Realizar una expresión regular para encontrar la palabra "foo" solo si no es seguida por "bar"

->.foo(!bar)

6. Realizar una expresión regular para encontrar la palabra "bar" solo si es precedida por "foo"

->(?=foo)bar

7. Realizar una expresión regular para encontrar la palabra "bar" solo si no es precedida por "foo".

->(?!foo)bar

8. Realizar una expresión regular para encontrar la palabra "Hello" solo si está al inicio del texto completo.

->(^hello)

9. Realizar una expresión regular para encontrar la palabra "goodbye" solo si está al final del texto completo

->(goodbye\$)(?!.)

10. Realizar una expresión regular para encontrar la secuencia "mid" solo si no está en un borde de palabra

->(?!\\b)mid(?!\\b)

1. Realizar una expresión regular para validar números de tarjeta de crédito que cumplan con las siguientes condiciones: ▪ Los números deben estar agrupados en bloques de 4 dígitos. ▪ Cada bloque debe estar separado por un espacio o un guión. ▪ Debe haber un total de 4 bloques (16 dígitos en total).

→

->^(\\d{4}[-\\s]){3}\\d{4}\$

2. Realizar una expresión regular para encontrar y extraer texto que esté entre comillas dobles, pero que no esté precedido por una barra invertida (\\)

->(?!\\)"(^[^\\]*(?:\\.\\.[^\\]*)*)"

3. Realizar una expresión regular para encontrar fechas en formato DD/MM/YYYY donde el día esté entre 01 y 31, el mes entre 01 y 12 y el año entre 1900 y 2099.

→^(0[1-9]|[12][0-9]|3[01])/(0[1-9]|1[0-2])/((19|20)\\d{2})\$