

# Expressions régulières (en javascript)

# L'objet RegExp

- Une expression régulière est un *objet standard javascript* (comme String ou Array)
- Instancier une expression régulière
  - var re = **new RegExp**(motif)
  - ou
  - var re = /motif/
- Exemple
  - var re = **new RegExp**('a') ou
  - Var re = /a/

# Motif d'une expression régulière

- `var re = new RegExp(motif)`
  - un symbole
    - `var re=new RegExp('a')`
  - un ensemble de symboles
    - `var re=new RegExp('[abc]')`
  - le complément à un ensemble de symboles
    - `var re=new RegExp('[^abc]')`
  - Un intervalle
    - `var re=new RegExp('[3-6]')`
    - `var re=new RegExp('[0-9]')`, ou  
`var re=new RegExp('\\d')`

# Motif d'une expression régulière

- `var re = new RegExp(motif)`
  - un intervalle (pour les lettres)
    - `var re=new RegExp('[a-d]')`
  - Une alternative
    - `var re=new RegExp('[a|c]')`

# Quantificateurs

- Fermeture de Kleene:
  - 0 à n occurrences
    - `var re=new RegExp('a*')`
  - 1 à n occurrences
    - `var re= new RegExp('a+')`
  - Un nombre fixe d'occurrences
    - `var re= new RegExp('a{3}')`

# Début et fin de séquence

- Début de séquence
  - **var** re= new RegExp('^a')
- Fin de séquence
  - **var** re=new RegExp('a\$')
- Début et fin de séquence
  - **var** re=new RegExp('^a\$')

# Modificateurs

- `var r=new RegExp(motif,modificateur)`
- Recherche globale
  - `var r= new RegExp('a','g')`  
ou `var r=/a/g`
- Recherche à la fois minuscule/majuscule
  - `var r= new RegExp('a','i')`