

Expressions régulières

Principe

La classe `NSRegularExpression` est utilisée pour représenter des expressions régulières

Création d'une expression régulière

- ❖ `init(pattern:options:)`

Recherche d'une chaîne de caractères correspondant

- ❖ `numberOfMatches(in:options:range:)`
- ❖ `matches(in:options:range:)`

26

Expressions régulières

Les motifs de recherche

^	Indique le début de la chaîne exemple : ^chat reconnaît une ligne qui commence par chat
\$	Indique la fin de la chaîne exemple : chat\$ reconnaît une ligne qui finit par chat
.	Le point indique n'importe quel caractère

*	Indique 0, 1 ou plusieurs occurrences du caractère exemple : a* reconnaît « pas de a », a, aa, aaa...
+	Indique une ou plusieurs occurrences du caractère exemple : a+ reconnaît a, aa, aaa...
?	Indique 0 ou une occurrence du caractère exemple : a? reconnaît « pas de a » et a

27

Expressions régulières

Les motifs de recherche

[...]	tous les caractères énumérés entre les crochets exemple : [æeiouy] reconnaît toutes les voyelles
[^...]	Tous les caractères sauf ceux énumérés exemple : [^æeiouy] reconnaît tous les caractères sauf les voyelles
	alternative - ou reconnaît l'un ou l'autre exemple : aa ab reconnaît les chaînes aa ou ab
[...-...]	intervalle de recherche exemple : [0-9] reconnaît tous les chiffres de 0 à 9

28

Expressions régulières

Les motifs de recherche

a{3}	correspond exactement à aaa
a{2,}	correspond à un minimum de deux a consécutifs soit aa, aaa, aaaa....
a{2,4}	correspond uniquement à aa, aaa, aaaa
(...)	Limite la portée d'un masque exemple : a(b){2} reconnaît la chaîne abb

Certains caractères sont des caractères spéciaux, ce sont les métacaractères:

* ? + [] () { } ^ \$ | \ . /

Pour les rechercher dans une chaîne de caractères il faut les échapper avec \\

Exemple : \\[.{3}\\] reconnaît une suite de trois caractères entre crochets

29

Expressions régulières

Utilisation de la classe **NSRegularExpression**

ViewController.swift

```
override func viewDidLoad() {
    super.viewDidLoad()

    let s1 = "123abc"
    let s2 = "abc123"
    let s3 = "abc"

    for s in [s1, s2, s3] {
        do {
            let regex = try NSRegularExpression(pattern:"^\\d{3}", options: .caseInsensitive)
            let match = regex.numberOfMatches(in: s,
                                              options: .reportProgress, range: NSRange(location:0,
                                                                 length:s.characters.count))
            if (match > 0) {
                print("\((s)) correspond à l'expression régulière")
            }
        } catch {
            print("Erreur")
        }
    }
}
```