

## match と search

search は Perl と同じ。部分マッチ。グローバルオプションが意味を持つ。  
match は常に対象文字列の先頭からのマッチを行う。

## 後方参照

### インデックスの数え方

```
m = re.match(r'((a|b|c)ddd)(.*)', s)
```

最初のカッコが 1 個目。  
最初のカッコの中に含まれるサブカッコが 2 個目。  
それから最初のカッコの後ろにあるカッコが 3 個目。  
という風に数える。

### 後方参照の利用方法

正規表現内では、`\number` が利用できる。正規表現内なので、`\` はエスケープする必要がある。

```
m = re.match(r'(\d)\#\#\d1', s)
```

`re.sub()` に渡す置き換え文字列の中では、同じく `\number` と、`\g<number>` の 2 種類の後方参照表記が使える。`\g<number>` は単に、リテラルの数字文字列と紛れないようにするための記法。

## グループ

`m.group(0)`

正規表現にマッチした文字列全体。

`m.group(1)`

1 個目のマッチグループ文字列。

`m.group(1, 3)`

1 個目のマッチグループ文字列と 3 個目のマッチグループ文字列のタプル。

`m.group('name')`

名前付きグループ (`?P<name>...`) にマッチした文字列。

`m.groups()`

すべてのマッチグループ文字列のタプル。マッチしなかったグループに対しては `None` が返される。

`m.groups('N')`

すべてのマッチグループ文字列のタプル。マッチしなかったグループに対しては引数文字列が返される。

## グローバルマッチ

`re.findall`、`re.finditer` を使う。