

# Les chaînes de caractères

Plusieurs traitements peuvent nécessiter la manipulation de chaînes de caractères :

- Trouver la longueur d'une chaîne de caractères;
- Extraire des caractères dans une chaîne;
- Remplacer un caractère par un autre dans une chaîne;
- Vérifier la présence d'un caractère dans une chaîne;
- Convertir une chaîne en majuscules, en minuscules;
- Etc.

Voyons certaines méthodes de manipulation de chaînes de caractères offertes par la classe **String** :

```
{  
    // Déclaration des variables qui serviront aux tests de manipulation  
    // de chaînes de caractères  
  
    string chaineUn = "";  
    string chaineDeux = "";  
    bool réponse;  
  
    // Méthode ToLower()  
    // =====  
    // Retourne une chaîne où tous les caractères sont en minuscules  
    chaineUn = "Judith";  
    chaineDeux = chaineUn.ToLower();  
    Console.WriteLine(chaineDeux);      // judith  
    Console.ReadKey();  
  
    // Méthode ToUpper()  
    // =====  
    // Retourne une chaîne où tous les caractères sont en majuscules  
    chaineUn = "Judith";  
    chaineDeux = chaineUn.ToUpper();  
    Console.WriteLine(chaineDeux);      // JUDITH
```

```

// Méthode Contains
// =====
// Retourne True si chaineDeux est incluse dans la chaîne, False sinon
chaineUn = "Judith";
chaineDeux = "dth";
réponse = chaineUn.Contains(chaineDeux);
Console.WriteLine(réponse);          // False
Console.ReadKey();

```

```

// Méthode Équals
// =====
// Retourne True si chaineDeux est égale à chaineUn, False sinon
chaineUn = "Judith";
chaineDeux = "Judith";
réponse = chaineUn.Equals(chaineDeux);
Console.WriteLine(réponse);          // True
réponse = chaineUn.Equals("Judithe");
Console.WriteLine(réponse);          // False

```

```

// Propriété Length
// =====
// Retourne le nombre de caractères dans la chaîne
int longueur = 0;
chaineUn = "Judith";
longueur = chaineUn.Length;
Console.WriteLine(longueur);          // 6

```

```

// Méthode Substring (int départ)
// =====
// Retourne le contenu de la chaîne à partir de la position départ
chaineUn = "Judith";
chaineDeux = chaineUn.Substring(0);
Console.WriteLine(chaineDeux);        // Judith
chaineDeux = chaineUn.Substring(3);
Console.WriteLine(chaineDeux);        // ith

```

```

// Méthode Substring (int départ, int nombreDeCaractèresAExtraire)
// =====
// Retourne la sous-chaîne de longueur nombreDeCaractèresAExtraire
// débutant à la position départ
chaîneUn = "Judith";
chaîneDeux = chaîneUn.Substring(0,3);
Console.WriteLine(chaîneDeux);    // Jud
chaîneDeux = chaîneUn.Substring(4,2);
Console.WriteLine(chaîneDeux);    // th

// Méthode Remove (int ÀPartirDe)
// =====
// Retourne la chaîne obtenue en supprimant tous les caractères
// à partir de la position ÀPartirDe
chaîneUn = "Judith";
chaîneDeux = chaîneUn.Remove(3);
Console.WriteLine(chaîneDeux);    // Jud

// Méthode Remove (int ÀPartirDe, int nombreDeCaractèresASupprimer)
// =====
// Retourne la chaîne obtenue en supprimant la sous-chaîne de
// longueur nombreDeCaractèresASupprimer à partir de la position ÀPartirDe
chaîneUn = "Judith";
chaîneDeux = chaîneUn.Remove(2,2);
Console.WriteLine(chaîneDeux);    // Juth

// Méthode Insert (int ÀPartirDe, string autreChaîne)
// =====
// Insère autreChaîne à partir de la position ÀPartirDe dans la chaîne.
chaîneUn = "Judith";
chaîneDeux = "Bettez";
string chaîneFinale = chaîneUn.Insert(3, chaîneDeux);
Console.WriteLine(chaîneUn);      // Judith
Console.WriteLine(chaîneDeux);    // Bettez
Console.WriteLine(chaîneFinale);  // JudBettezith

```

```
// Méthode Replace (string carARemplacer, string carDeRemplacement)
// =====
// Retourne une chaîne où tous les caractères à remplacer sont
// remplacés par le caractère de remplacement.
chaîneUn = "P00JudithBettezaimelaP00";
chaîneDeux = chaîneUn.Replace("P00", "Programmation orientée objet");
Console.WriteLine(chaîneDeux);
    // Programmation orientée objetJudithBettezaimelaProgrammation orientée objet
```

```
// Méthode IndexOf (string chaîneAChercher)
// =====
// Retourne la position où chaîneAChercher débute dans la chaîne.
// Retourne -1 si non trouvée.
chaîneUn = "Judith";
chaîneDeux = "dit";
int position = chaîneUn.IndexOf(chaîneDeux);
Console.WriteLine(position);    // 2
```

```
// Méthode StartsWith (string chaîneParLaquelleDoitDébuter)
// =====
// Retourne true si la chaîne commence par chaîneParLaquelleDoitDébuter, False sinon
chaîneUn = "YES!!!Apprendre laP00";
réponse = chaîneUn.StartsWith("y");
Console.WriteLine(réponse);    // False
```