

Vecteurs et HashMap

Vecteurs

Les vecteurs sont des listes à taille variable

```
fn main() {  
    // Je peux créer un vecteur mutable (qui peut être modifié) comme ceci :  
    let mut ma_liste = vec![1,2,3,4,5,6,7,8,9];  
    println!("Ma liste est {:?}", ma_liste);  
  
    // Je peux ensuite ajouter des éléments  
    ma_liste.push(10);  
  
    // Récupérer la longueur ou n'importe quel élément  
    println!("Ma liste a une longueur de {} et son premier élément est {}", ma_liste.len(), ma_liste[0]);  
  
    // Supprimer le dernier élément  
    ma_liste.pop();  
  
    // Ou supprimer un élément d'index précis (dans ce cas, le premier)  
    ma_liste.remove(0);  
  
    println!("A présent ma liste est {:?}", ma_liste);  
}
```

Les HashMap

Un vecteur stocke les données par index (0, 1, 2, 3, 4, 5, etc) tandis qu'une hashmap stocke les données par clé ("foo", "hello")

```
use std::collections::HashMap;

fn main() {
    let mut dictionnaire = HashMap::new();

    // On peut ensuite insérer des données
    dictionnaire.insert("foo", "bar");
    dictionnaire.insert("hello", "world");
    dictionnaire.insert("life", "42");

    // On peut récupérer la valeur correspondante à une clé avec .get
    println!("Foo: {}", dictionnaire.get("foo").unwrap());

    // Pour supprimer un élément on utilise .remove
    dictionnaire.remove("life");

    // Mais on peut aussi itérer une hashmap
    for (key, val) in dictionnaire.iter() {
        println!("{}", key, val);
    }
}
```

En savoir plus

- [Rust By Example - Vectors](#)
- [Rust By Example - HashMap](#)
- [docs.rs - Vec](#)
- [docs.rs - HashMap](#)

Revision #1

Created 27 April 2023 06:10:31 by SnowCode

Updated 27 April 2023 06:12:49 by SnowCode