

Trabalhando com Sets

Outra estrutura de Lista, porém com umas particularidades, que são:

- Não aceita elementos repetidos.
- Não trabalha com indexação, ou seja, não garante ordem dos elementos inseridos!

Dito isso, é uma listagem de elementos, onde não terá elementos repetidos sem ordenação dos elementos inseridos!

Criando Sets

E para trabalhar com listas sets, basta criarmos um objeto do tipo **Set**, e veja que essa listagem é identificada com o uso de {}.

```
# Criando Sets
lista1 = set([10, 20, 20, 30, 40])
print(lista1)
```

{40, 10, 20, 30}

Veja que além dos elementos não se repetirem, está sem ordenação por não trabalhar com indexes!

Além disso, com os **Sets** podemos utilizar alguns operadores matemáticos, como o **Union()**, **Intersecção(&)** e **Diferença(-)**.

- **Union(|)**: Cria uma listagem **Set**, onde é realizado um “merge” de 2 listas, removendo todos os itens repetidos.

```
2 lista1 = set([10, 20, 20, 30, 40])
3 lista2 = set([20, 60, 70])
4
5 print(lista1 | lista2)
```

{20, 70, 40, 10, 60, 30}

- **Intersecção(&)**: Cria uma listagem **Set**, onde os itens serão o que estiverem em ambas as listas!

```
lista1 = set([10, 20, 20, 30, 40])
lista2 = set([20, 60, 70])

print(lista1 & lista2)
```

{20}

- **Diferença(-):** Cria uma listagem **Set**, onde removemos todos os itens que são iguais de 2 listas, permanecendo apenas os itens que são diferentes entre ambas as listas!

```
lista1 = set([10, 20, 20, 30, 40])  
lista2 = set([20, 60, 70])  
  
print(lista1 - lista2)
```

```
{40, 10, 30}
```