

Atividade 4 – Lista de Conjunto

Luna Costa Vasconcelos

Responda às seguintes questões:

01) O que é um conjunto em Dart?

Um conjunto no Dart é implementado pela classe 'set' e é um agrupamento de componentes distintos que não são ordenados e que também não podem ser repetidos. Os métodos add() e remove() podem ser usados para adicionar e remover elementos de conjuntos, respectivamente. As propriedades contains() e length podem ser usadas para determinar o tamanho do conjunto e para determinar se um elemento está presente.

02) Como criar um conjunto vazio em Dart?

Algumas formas de criar um conjunto vazio em Dart são as seguintes:

- Usando o construtor padrão da classe Set:
`Set<int> conjunto = Set<int>();`
- Usando a sintaxe de inicialização de conjunto:
`Set<int> conjunto = { };`
- Usando o construtor nomeado of da classe Set:
`Set<int> conjunto = Set<int>.of([]);`

03) Como criar um conjunto com elementos em Dart?

Algumas formas de criar um conjunto com elementos em Dart são as seguintes:

- Usando a sintaxe de conjunto:
`Set<int> conjunto = { 1, 2, 3 };`
- Usando o construtor nomeado 'from' da classe Set:
`Set<int> conjunto = Set<int>.from([1, 2, 3]);`
- Usando o operador spread (...) com a sintaxe de conjunto:
`Set<int> conjunto = {...[1, 2, 3]};`

04) Qual a diferença entre um conjunto e uma lista em Dart?

Um conjunto (Set) e uma lista (List) são estruturas de dados no Dart que podem armazenar um grupo de elementos. A principal distinção entre os dois é de que uma lista preserva a ordem dos elementos e permite elementos duplicados, enquanto um conjunto não. Ou seja, deve ser usado uma lista se precisar manter os componentes em uma ordem precisa e não se importar se houver duplicatas. Por outro lado, você deve usar um conjunto se precisar garantir que não haja duplicatas e caso a ordem não seja importante. Cada estrutura de dados também possui um conjunto diferente de métodos, alguns dos quais são exclusivos para listas ou conjuntos.

05) Como adicionar um elemento a um conjunto em Dart?

Pode-se utilizar o método add() para adicionar somente um elemento ao conjunto ou o método addAll() para adicionar vários elementos de uma vez. Como nesses exemplos:

```
Set<int> conjunto = { 1, 2, 3 };
conjunto.add(4);
Set<int> conjunto = { 1, 2, 3 };
conjunto.addAll([4, 5, 6]);
```

06) Como remover um elemento de um conjunto em Dart?

Pode-se utilizar o método remove() para remover somente um elemento do conjunto ou o método removeAll() para remover vários elementos de uma vez. Como nesses exemplos:

```
Set<int> conjunto = { 1, 2, 3 };
conjunto.remove(2);
Set<int> conjunto = { 1, 2, 3, 4, 5, 6 };
conjunto.removeAll([2, 4, 6]);
```

07) Como verificar se um conjunto contém um determinado elemento em Dart?

Para verificar se um conjunto contém um determinado elemento em Dart, pode-se utilizar o método `contains()`. Como nesse exemplo:

```
Set<int> conjunto = { 1, 2, 3 };
print(conjunto.contains(2));
```

O resultado desse exemplo deve ser `True`, pois o conjunto em questão possui o elemento 2.

08) Como verificar se um conjunto é vazio em Dart?

Para verificar se um conjunto é vazio em Dart, pode-se utilizar os métodos `isEmpty()` ou `isNotEmpty()`. Segue um exemplo utilizando `isEmpty()`:

```
Set<int> conjunto = { };
print(conjunto.isEmpty());
```

O resultado desse exemplo deve ser `True`, pois o conjunto em questão está vazio.

09) Como unir dois conjuntos em Dart?

Para unir dois conjuntos em Dart, podemos usar o método `union()`, assim como no seguinte exemplo:

```
Set<int> conjunto1 = { 1, 2, 3 };
Set<int> conjunto2 = { 3, 4, 5 };
Set<int> conjuntoUnido = conjunto1.union(conjunto2);
print(conjuntoUnido);
```

O resultado do `print` será o conjunto: `{ 1, 2, 3, 4, 5 }`, ou seja, a união dos dois conjuntos sem repetição de elementos (visto que o 3 aparece nos dois conjuntos, mas aparece só uma vez, característica do tipo `Set` no dart).

10) Como obter a interseção de dois conjuntos em Dart?

Para obter a interseção de dois conjuntos em Dart, você pode utilizar o método `intersection()`, assim como no seguinte exemplo:

```
Set<int> conjunto1 = { 1, 2, 3 };
Set<int> conjunto2 = { 3, 4, 5 };
Set<int> intersecao = conjunto1.intersection(conjunto2);
print(intersecao);
```

O resultado do `print` será o conjunto: `{ 3 }`, ou seja, a interseção entre os dois conjuntos.

11) Como obter a diferença entre dois conjuntos em Dart?

Para obter a diferença entre dois conjuntos em Dart, você pode utilizar o método `difference()`, assim como no seguinte exemplo:

```
Set<int> conjunto1 = { 1, 2, 3 };
Set<int> conjunto2 = { 3, 4, 5 };
Set<int> diferenca = conjunto1.difference(conjunto2);
print(diferenca);
```

Os dois conjuntos de números inteiros neste exemplo são `conjunto1` com os números 1, 2 e 3, e `conjunto2` com os elementos 3, 4 e 5. A diferença dos dois conjuntos é então obtida em um terceiro conjunto chamado `diferenca` usando o método `difference()`. Como resultado, é criado um conjunto que inclui apenas elementos do `conjunto1` e não do `conjunto2`.

12) Como verificar se um conjunto é subconjunto de outro conjunto em Dart?

Para verificar se um conjunto é subconjunto de outro conjunto em Dart, podemos utilizar o método `containsAll()`, segue um exemplo:

```
Set<int> conjunto1 = {1, 2, 3, 4, 5};  
Set<int> conjunto2 = {1, 2, 6};  
bool isSubconjunto = conjunto1.containsAll(conjunto2);  
print(isSubconjunto);
```

O resultado será Falso nesse exemplo, pois o conjunto2 não é subconjunto do conjunto1

13) Como verificar se dois conjuntos são iguais em Dart?

Para verificar se dois conjuntos são iguais em Dart, podemos aliar o método `containsAll()` e o operador `length` para comparar os elementos e o tamanhos dos conjuntos respectivamente. Segue um exemplo:

```
Set<int> conjunto1 = {1, 2, 3};  
Set<int> conjunto2 = {3, 2, 1};  
bool saoIguais = conjunto1.length == conjunto2.length &&  
    conjunto1.containsAll(conjunto2);
```

14) Como criar um conjunto a partir de uma lista em Dart?

Para criar um conjunto a partir de uma lista em Dart, podemos passar a lista como argumento para o construtor do conjunto `Set`. Por exemplo:

```
List<int> lista = [1, 2, 3, 4, 5];  
Set<int> conjunto = Set<int>.from(lista);
```

Ou podemos usar também usar o método `toSet()`:

```
List<int> lista = [1, 2, 3, 4, 5];  
Set<int> conjunto = lista.toSet();
```

15) Como criar uma lista a partir de um conjunto em Dart?

Podemos usar a mesma lógica da questão 14 para criar uma lista a partir de um conjunto. Por exemplo:

```
Set<int> conjunto = {1, 2, 3, 4, 5};  
List<int> lista = List<int>.from(conjunto);
```

Ou podemos usar também usar o método `toList()`:

```
Set<int> conjunto = {1, 2, 3, 4, 5};  
List<int> lista = conjunto.toList();
```

16) Como transformar um conjunto em uma lista de strings em Dart?

Podemos usar o método `toList()` da classe `Set` para obter uma lista de elementos do conjunto e, em seguida, usar o método `toString()` para converter cada elemento em uma string. Aqui está um exemplo:

```
Set<int> conjunto = {1, 2, 3, 4, 5};  
List<String> listaDeStrings = conjunto.map((e) => e.toString()).toList();
```

17) Como calcular a união de vários conjuntos em Dart?

Isso pode ser feito da mesma forma que foi mostrado para somente um conjunto. Segue um exemplo:

```
Set<int> conjunto1 = {1, 2, 3};  
Set<int> conjunto2 = {2, 3, 4};  
Set<int> conjunto3 = {3, 4, 6};
```

```
Set<int> uniao = conjunto1.union(conjunto2).union(conjunto3);
print(uniao);
//Saída: {1, 2, 3, 4, 6}
```

18) Como calcular a interseção de vários conjuntos em Dart?

Isso pode ser feito da mesma forma que foi mostrado para somente um conjunto. Segue um exemplo:

```
Set<int> conjunto1 = {1, 2, 3, 4, 5};
Set<int> conjunto2 = {3, 4, 5, 6, 7};
Set<int> conjunto3 = {5, 6, 7, 8, 9};
```

```
Set<int> intersecao = conjunto1.intersection(conjunto2).intersection(conjunto3);

print(intersecao);
// Saída: {5}
```

19) Como verificar se dois conjuntos são disjuntos em Dart?

Podemos verificar se dois conjuntos são disjuntos utilizando novamente o método `intersection()` para obter a interseção entre os conjuntos e em seguida verificar se o resultado é um conjunto vazio. Caso seja um conjunto vazio, isso significa que eles são conjuntos disjuntos pois não possuem elementos em comum, caso o conjunto resultante não seja vazio, isso significa que eles não são conjuntos disjuntos pois possuem elementos em comum. Por exemplo:

```
Set<int> conjunto1 = {1, 2, 3};
Set<int> conjunto2 = {4, 9, 6};
bool saoDisjuntos = conjunto1.intersection(conjunto2).isEmpty;

if (saoDisjuntos == true){
  print("Os conjuntos conjunto1 e conjunto2 são disjuntos");
} else {
  print("Os conjuntos conjunto1 e conjunto2 não são disjuntos");
}
// Saída: Os conjuntos conjunto1 e conjunto2 são disjuntos
```

20) Como remover todos os elementos duplicados de uma lista usando um conjunto em Dart?

Podemos criar um conjunto a partir da lista e, em seguida, converter o conjunto de volta para a lista. Isso garantirá que todos os elementos duplicados sejam removidos, visto que em conjuntos não são permitidos elementos duplicados. Por exemplo:

```
List<int> lista = [1, 1, 2, 2, 3, 2, 4, 3, 5, 1, 6, 5];
Set<int> conjunto = Set<int>.from(lista);
List<int> listaSemDuplicatas = conjunto.toList();
print(listaSemDuplicatas);
// Saída: [1, 2, 3, 4, 5, 6]
```