**Estruturas, métodos e Map vs Objeto**

**Map:**

* uma coleção de arrays no **formato [chave, valor];**
* pode iterado por um loop for...of
* é declarado com o “new Map( )”
* const myMap = **new** *Map*()

**Métodos:**

* **adicionar, ler e deletar**
* adicionar – set
* const myMap = **new** *Map*()
* myMap.set('apple', 'fruit');
* //Map (1) {"apple" => "fruit"}   - adiciona um valor/ "apple" é minha chave e o valor é a "fruit"
* myMap.get("apple");
* //"fruit"  -  checar o valor de apple
* myMap.delete("apple");
* // true   -   se quiser deletar algum valor, usar "delete"
* myMap.get("apple")
* //undefined   -   se quiser retornar o valor de apple, vai dar undefined por não existir mais na coleção

**Map vs Objeto**

* **maps** podem ter chaves de qualquer tipo. Objeto usa a chave **string**
* maps possuem a propriedade length.
* maps são mais fáceis de iterar
* utilizado quando o valor das chaves é desconhecido. Objeto é utilizado quando já se conhece o valor das chaves
* os valores tem o mesmo tipo

**Set**

**Estrutura:**

* sets são estruturas que armazenam apenas **valores únicos**
* const myArray = [1, 1, 2, 2, 3, 3, 4, 4, 5, 2, 1]; //pode ter valores repetidos
* const mySet = **new** *Set*(myArray); //é uma coleção com valores únicos, que nunca irão repetir

**Métodos:**

* adicionar, consultar e deletar
* const mySet = **new** *Set*();
* mySet.add (1); //adicionar
* mySet.add (5);
* mySet.has(1); //consultar
* //true
* mySet.has(3); //consultar
* //false
* mySet.delete(5); //deletar

**Set vs Array**

* possui valores únicos. No Array os valores podem repetir
* em vez da propriedade length, consulta-se o número de registros pela propriedade size
* não possui os métodos map, filter, reduce, etc.
* o set é muito mais limitado que o array