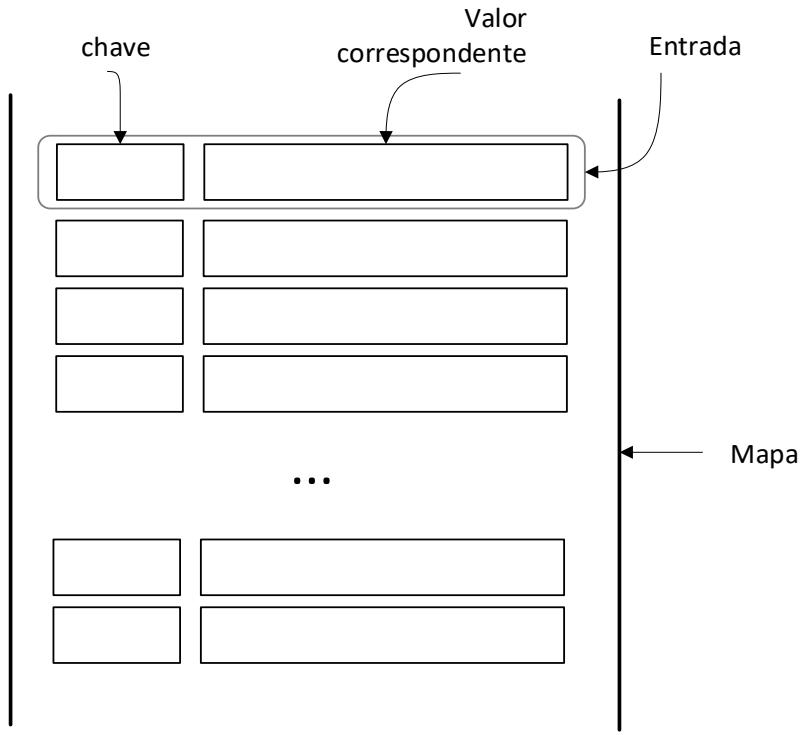
Mapa de dispersão

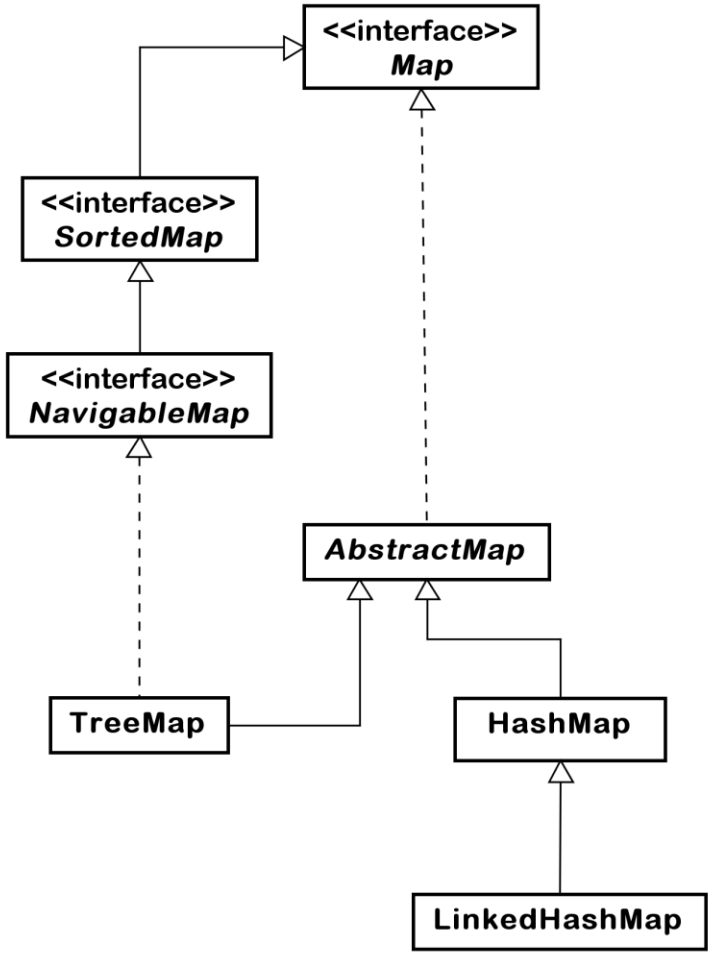
FURB
UNIVERSIDADE DE BLUMENAU

O que é um mapa de dispersão?

- Um mapa armazena uma coleção de pares *chave/valor.*
- As chaves funcionam como índices.
- Cada chave mapeia um valor.
- Permite executar busca, exclusão e atualização de dados (a partir da chave) de forma rápida
- Também conhecido como tabela hash (*haspmap*), mapa de espalhamento, etc.



Mapas de dispersão



Declaração de uma mapa de dispersão

- Sintaxe para declarar um mapa de dispersão:

```
HashMap<Classe1, Classe2> identificador;
```

Classe da chave

Classe do objeto a ser armazenado

- Exemplo:

```
HashMap<Integer, Aluno> alunos;
```

Criação de um mapa de dispersão

- Sintaxe:

```
identificador = new HashMap<>()
```

- Exemplo:

```
alunos = new HashMap<>()
```

Incluir objetos no mapa de dispersão

- Usar o método put() para incluir um objeto no mapa
- Requer informar a chave do objeto
- Exemplo:

```
alunos.put(287812, new Aluno(287812, "José da Silva"));
alunos.put(128444, new Aluno(128444, "Leandro Souza"));
alunos.put(166886, new Aluno(166886, "Marina Albuquerque"));


Aluno aluno = new Aluno(190443, "Marta Antunes");
alunos.put(190443, aluno);
```

Localizar um objeto no mapa

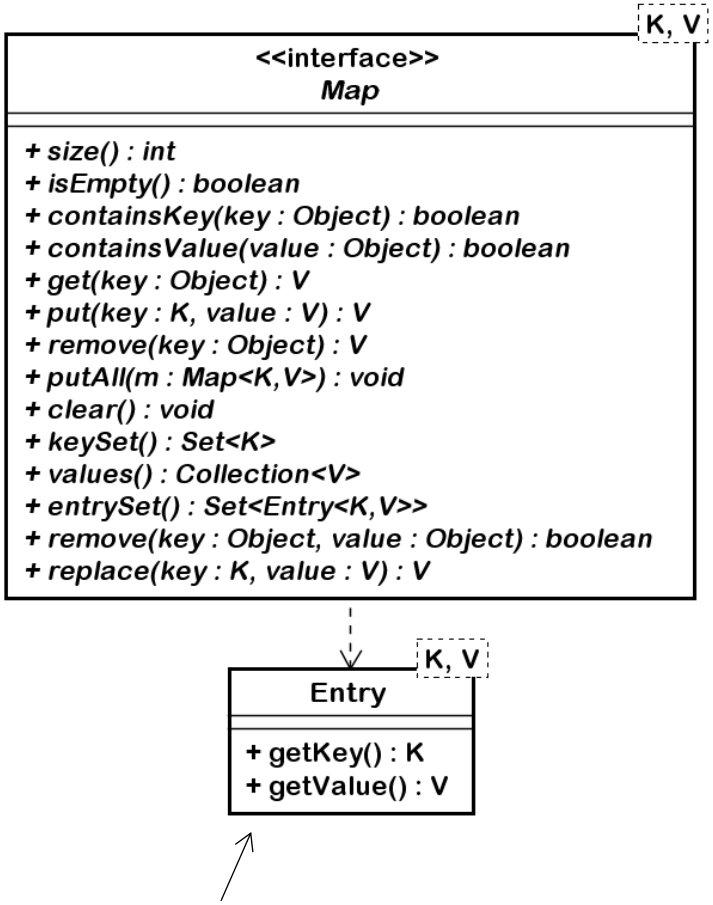
- A localização deve ser feita pela chave do objeto.
- Utilizar o método `get()` para localizar o objeto
- Exemplo:

```
Aluno alunoPesquisado = alunos.get(166886);
```


Procura um aluno cuja matrícula seja 166886

- Se a chave não tiver sido inserida previamente, o método `get()` retornará `null`.

A interface Map



Classe de objetos que armazenam o par chave/valor

Método	Descrição
put(K, V)	Armazena no mapa o par com chave K e valor V. Se a chave já foi inserida, retorna o valor já associado e o altera no mapa. Se a chave ainda não tinha sido inserida, armazena o par no mapa.
get(Object)	Retorna o valor associado à chave. Retorna null, se não existir (ou se o valor associado for nulo).
remove(Object)	Remove o objeto a partir da chave informada. Devolve o valor associado à chave removida (ou null, se a chave não existe no mapa ou está associada ao valor null)
containsKey()	Retorna true se há um par no mapa com a chave informada
putAll(Map<K,V)	Copia os dados de um mapa.
keySet(): Set<K>	Retorna uma coleção do tipo contendo as chaves armazenadas no mapa
values()	Retorna uma coleção do tipo Collection contendo os valores armazenados no mapa.
entrySet()	Devolve uma coleção com os pares (chave/valor) armazenados no mapa.

O método hashcode()

- É responsabilidade da classe da chave sobrescrever o método hashCode()
- O hashcode é um valor que pertence ao objeto. Trata-se de um número de 32 bits, que permite que o objeto seja utilizado numa estrutura de dados *mapa de dispersão*
  - A partir deste número é possível calcular qual a posição em que se encontra o par
- Dois objetos iguais devem retornar o mesmo hashcode.
  - Se dois objetos (*a* e *b*) são iguais, isto é, o a.equals(b) == true, então o método hashCode() deve retornar o mesmo valor para os dois objetos.
  - Sempre que o método equals() é implementado, deve-se sobrescrever hashCode() também, e vice versa.
- Dois objetos distintos podem retornar o mesmo hashcode, mas se possível, deve-se evitar.

Diagrama de classes