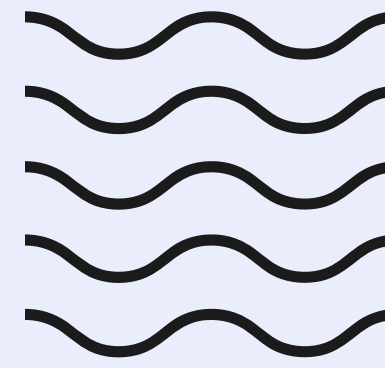


HashMap



ПОДГОТОВИЛ: АЛИШЕР ХАМИДОВ



Основные моменты:

План лекции

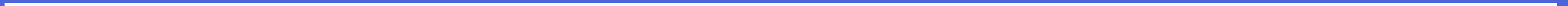
1. Как создать?
2. Методы
3. Важные замечания
4. Внутреннее устройство
5. Дополнительная информация





Что такое HashMap?

Ваши ассоциации

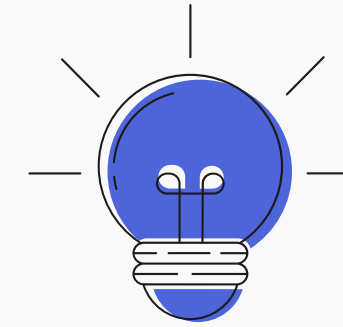




Когда мы работали с `ArrayList` мы могли получать доступ к элементам через индекс.

В `HashMap` мы тоже можем получать доступ к элементам, но не по индексу, а по специальному ключу.

Как создать?



```
import java.util.HashMap;
```

```
HashMap<String, String> capitalCities = new HashMap<String,  
String>();
```

```
<Тип_ключа, тип_значения>
```

Как добавить пару ключ-значение?

```
HashMap<String, String> capitalCities = new  
HashMap<String, String>();
```

```
capitalCities.put("Russia", "Moscow");  
capitalCities.put("Germany", "Berlin");  
capitalCities.put("Kasakhstan", "Astana");
```

```
System.out.println(capitalCities);
```



Как удалить пару ключ-значение по ключу?

```
HashMap<String, String> capitalCities = new  
HashMap<String, String>();
```

```
capitalCities.put("Russia", "Moscow");  
capitalCities.put("Germany", "Berlin");  
capitalCities.put("Kasakhstan", "Astana");
```

```
capitalCities.remove("Kasakhstan");  
System.out.println(capitalCities);  
//{Germany=Berlin, Russia=Moscow}
```

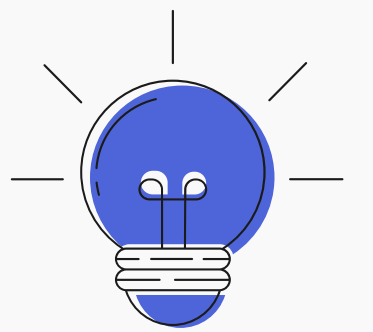


Как удалить все элементы?

```
HashMap<String, String> capitalCities = new  
HashMap<String, String>();
```

```
capitalCities.put("Russia", "Moscow");  
capitalCities.put("Germany", "Berlin");  
capitalCities.put("Kasakhstan", "Astana");
```

```
capitalCities.clear();  
System.out.println(capitalCities);  
//{}
```




```
capitalCities.put("Russia", "Moscow");  
capitalCities.put("Germany", "Berlin");  
capitalCities.put("Kasakhstan", "Astana");
```

```
System.out.println(capitalCities.keySet());
```

```
// [Kasakhstan, Germany, Russia]
```

```
System.out.println(capitalCities.values());
```

```
// [Astana, Berlin, Moscow]
```

```
System.out.println(capitalCities.entrySet());
```

```
// [Kasakhstan=Astana, Germany=Berlin, Russia=Moscow]
```

Как распечатать ключи?

```
for (String i : capitalCities.keySet()) {  
    System.out.println(i);  
}
```

Как распечатать значения?

```
for (String i : capitalCities.values()) {  
    System.out.println(i);  
}
```

Как распечатать ключи-значения?

```
for (String i : capitalCities.keySet()) {  
    System.out.println("key: " + i + " value: " + capitalCities.get(i));  
}
```

Как проверить, есть ли такой ключ?

```
productsByName.containsKey("Russia"); // true
```



Как проверить, есть ли такое
значение?

```
productsByName.containsValue("Moskow"); // true
```

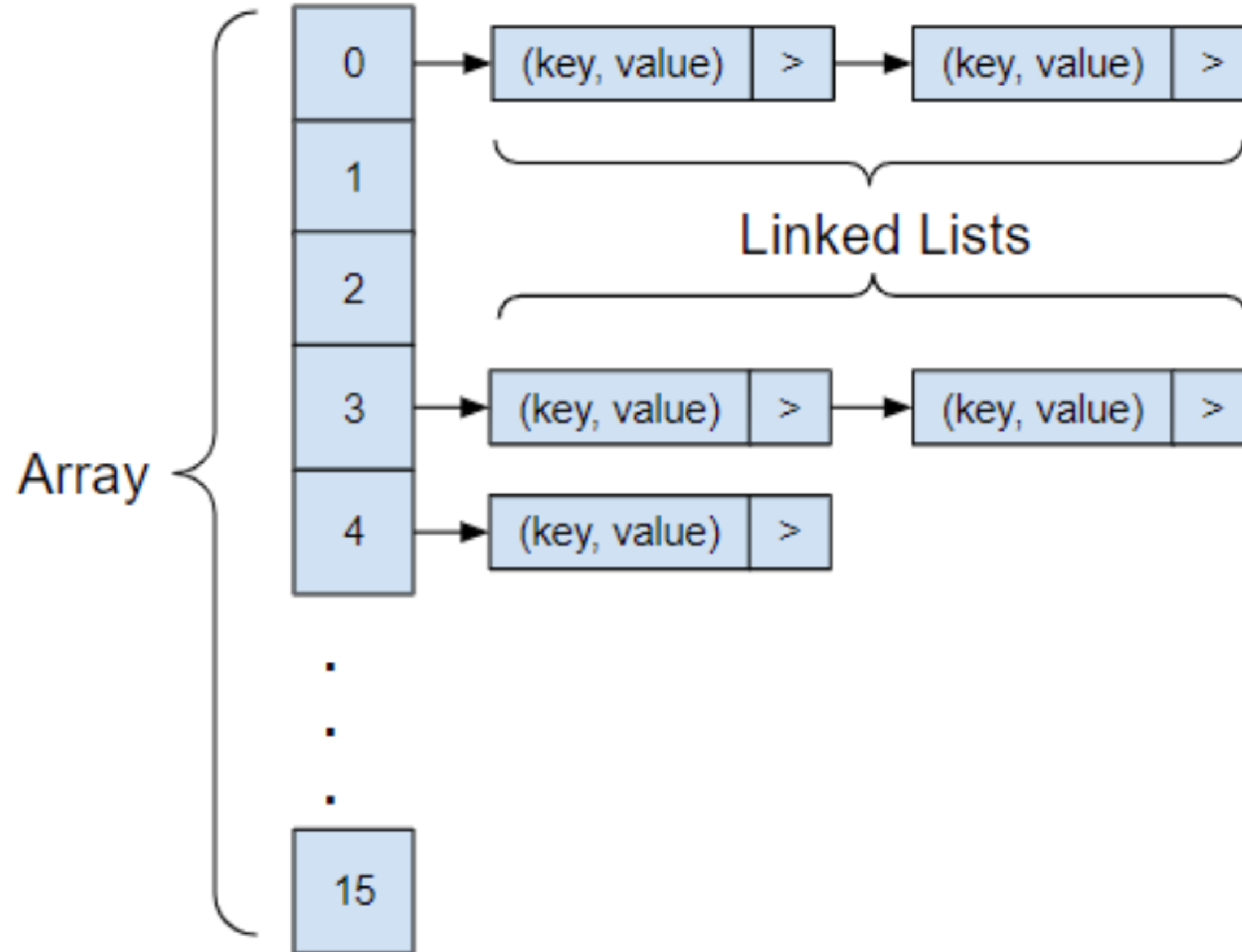


Важное замечание

Ключи и значения в `HashMap` на самом деле являются объектами. В приведенных выше примерах мы использовали объекты типа «`String`».

Мы можем использовать любой класс в качестве ключа в нашем `HashMap`. Однако, чтобы `map` работал правильно, нам нужно предоставить реализацию для `equals()` и `hashCode()` в классе ключа.

HashMap

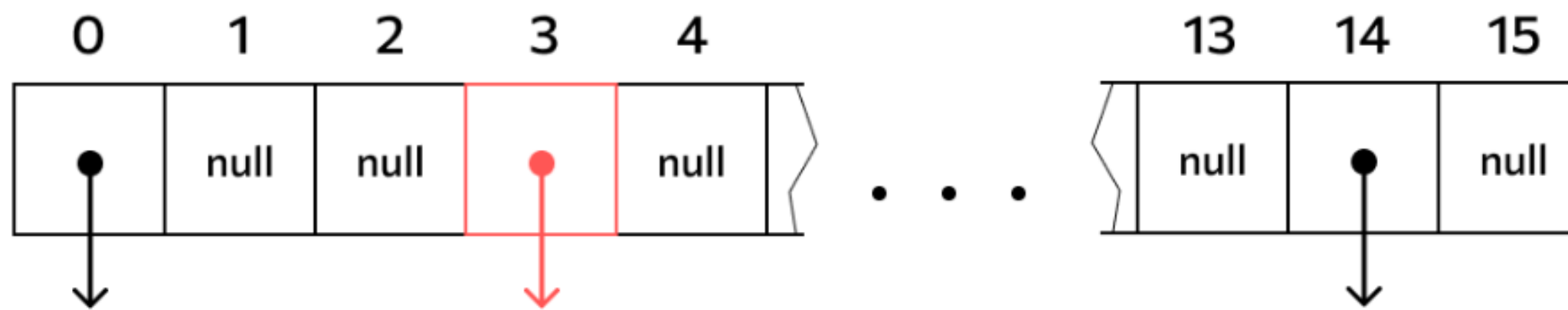




Важное замечание

HashMap хранит элементы в так-называемых корзинах, количество таких корзин называется **capacity**.

А вот количество элементов(ключ-значение) называется как обычно **size**.

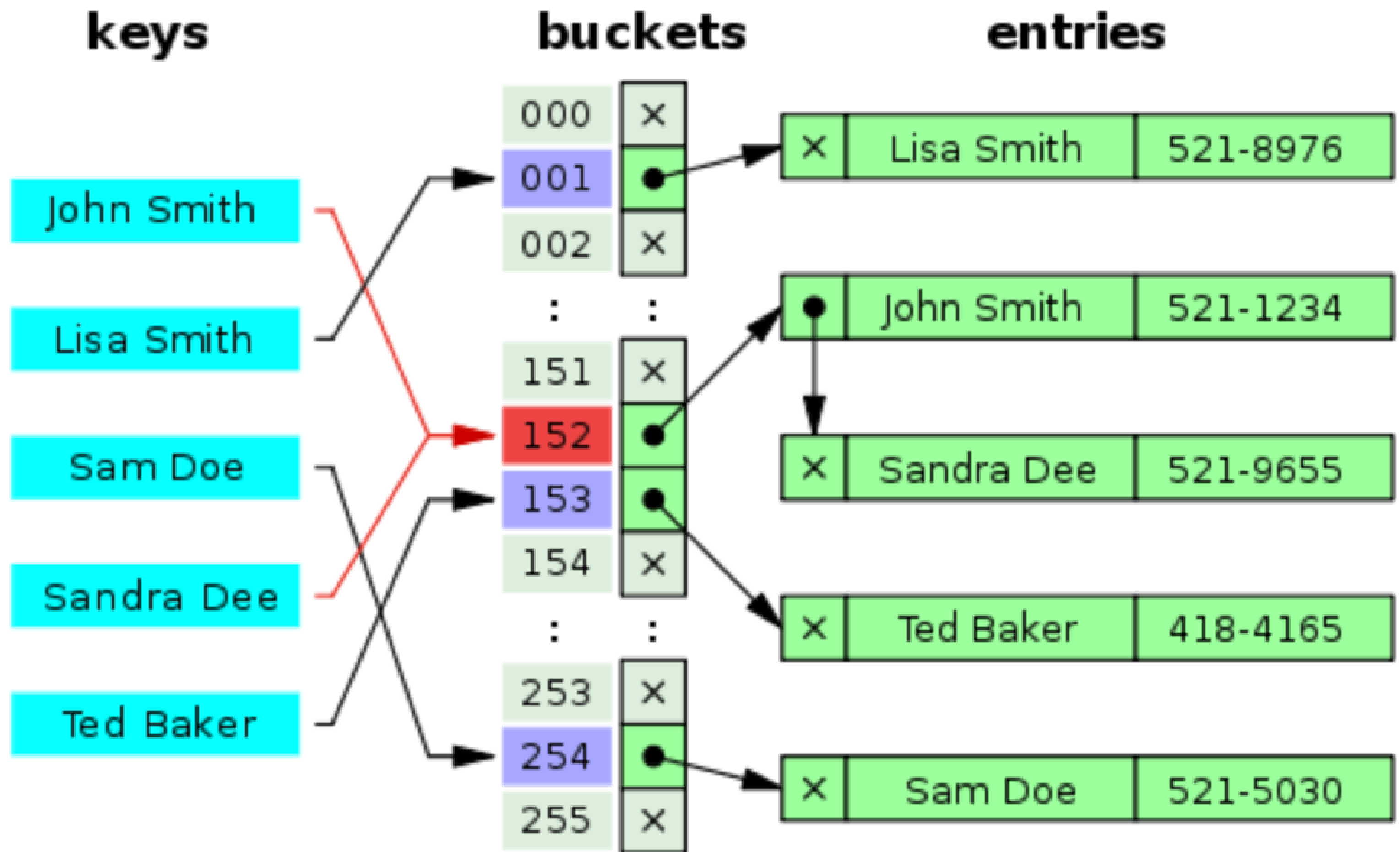


Entry	
hash	0
key	null
next	null
value	null

Entry	
hash	101603
key	"idx"
next	
value	two

Entry	
hash	99486
key	"key"
next	null
value	"one"

Entry	
hash	51
key	"0"
next	null
value	"zero"





Важное замечание

HashMap хранит элементы в так-называемых корзинах (bucket), количество таких корзин называется **capacity**.

Определяем индекс корзины:

$$\text{index} = \text{hashCode}(\text{key}) \& (n-1)$$

где n – это capacity

