

Version 1

```
class Tier {  
    int alter;  
  
    Tier(int alter) { this.alter = alter; }  
}
```

weder hashCode noch equals überschrieben.

- hashCode der Klasse Object liefert für jedes neue Objekt seine Zahl
- equals aus der Klasse Object vergleicht die Referenzen

... main...

```
Set set = new HashSet();
```

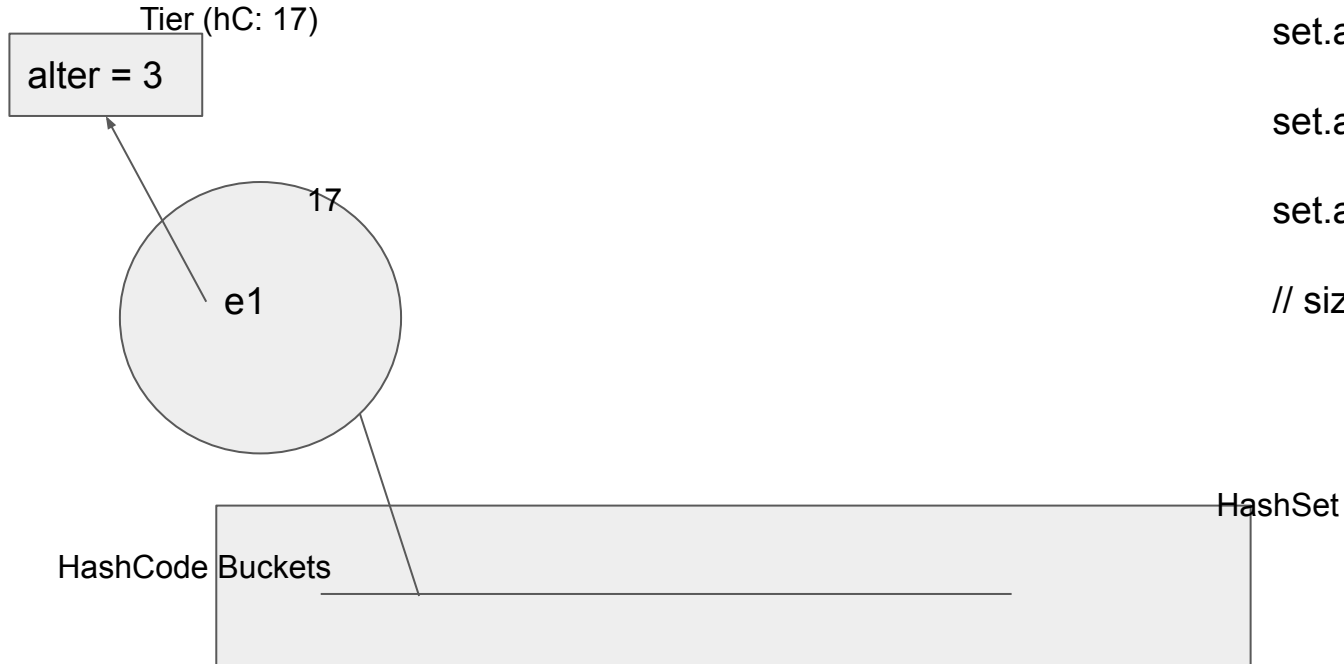
```
Tier t1 = new Tier(3);  
set.add(t1);  
set.add(t1);
```

```
set.add(new Tier(7));
```

```
set.add(new Tier(3));
```

```
// size = 3
```

1. Die add ruft hashCode-Methode zu dem Parameter auf: **param.hashCode()**. **Ergebnis ist z.B. 17**
2. Wenn es kein hashCode Bucket für ID=17 gibt: neues hashCode Bucket erzeugen, und den Parameter im hashCode Bucket kopieren



... main...

```
Set set = new HashSet();
```

```
Tier t1 = new Tier(3);  
set.add(t1);  
set.add(t1);
```

```
set.add(new Tier(7));
```

```
set.add(new Tier(3));
```

```
// size = 3
```

1. Die add ruft hashCode-Methode zu dem Parameter auf: **param.hashCode()**. **Ergebnis ist wieder 17**
2. Wenn es ein hashCode Bucket für ID=17 gibt: im gefundenen hashCode Bucket wird nach dem Element gesucht. Für den Vergleich wird die equals-Methode verwendet: `e1.equals(param)`. Falls es das Element bereits gibt, wird der **Parameter verworfen**

... main...

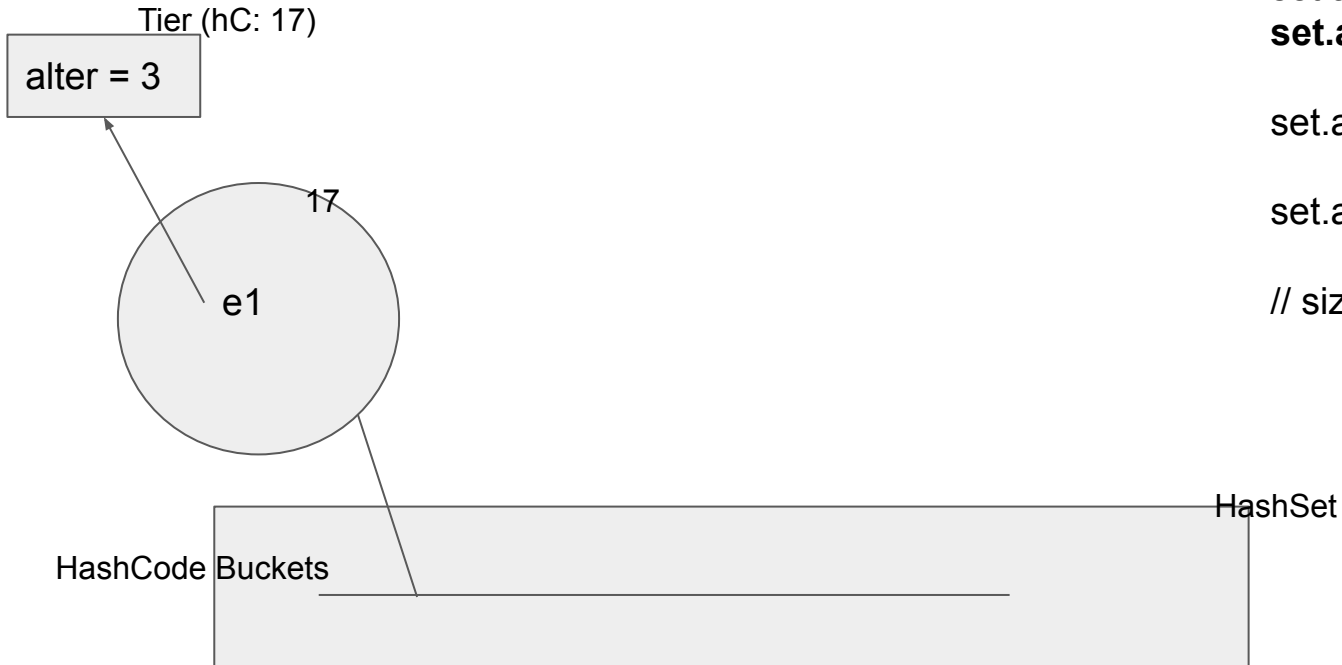
```
Set set = new HashSet();
```

```
Tier t1 = new Tier(3);  
set.add(t1);  
set.add(t1);
```

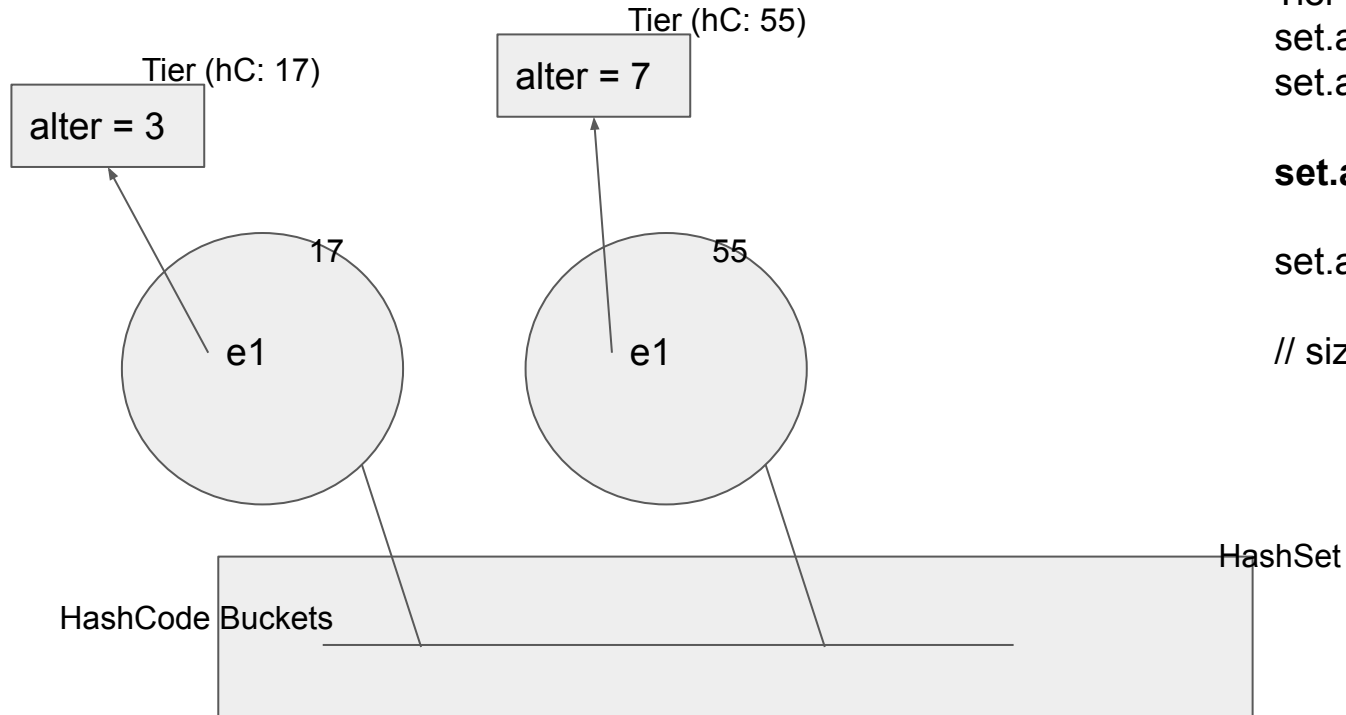
```
set.add(new Tier(7));
```

```
set.add(new Tier(3));
```

```
// size = 3
```



1. Die add ruft hashCode-Methode zu dem Parameter auf: **param.hashCode()**. **Ergebnis ist z.B. 55**
2. Wenn es kein hashCode Bucket für ID=55 gibt: neues hashCode Bucket erzeugen, und den Parameter im hashCode Bucket kopieren



... main...

```
Set set = new HashSet();
```

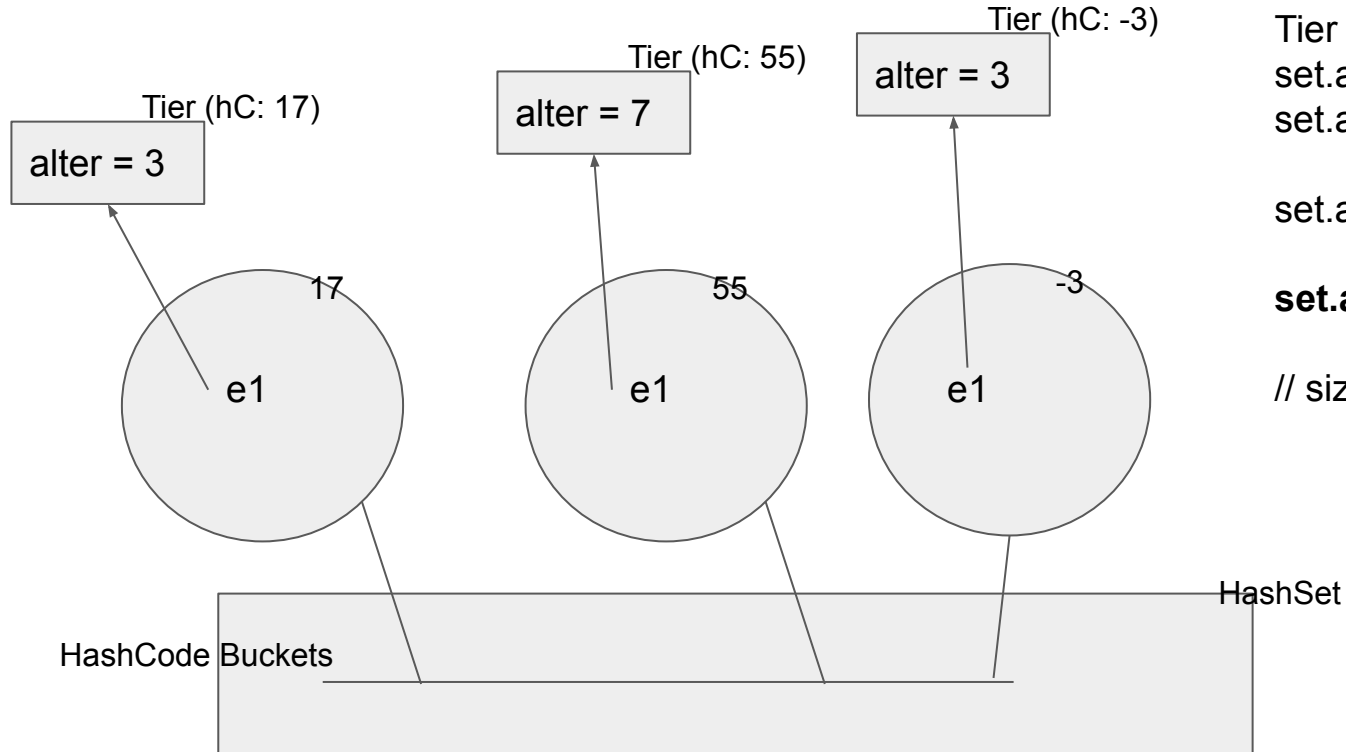
```
Tier t1 = new Tier(3);  
set.add(t1);  
set.add(t1);
```

```
set.add(new Tier(7));
```

```
set.add(new Tier(3));
```

```
// size = 3
```

1. Die add ruft hashCode-Methode zu dem Parameter auf: **param.hashCode()**. **Ergebnis ist z.B. -3**
2. Wenn es kein hashCode Bucket für ID=-3 gibt: neues hashCode Bucket erzeugen, und den Parameter im hashCode Bucket kopieren



... main...

```
Set set = new HashSet();
```

```
Tier t1 = new Tier(3);  
set.add(t1);  
set.add(t1);
```

```
set.add(new Tier(7));
```

```
set.add(new Tier(3));
```

```
// size = 3
```

Version 2

```
class Tier {  
    int alter;  
  
    Tier(int alter) { this.alter = alter; }  
  
    public int hashCode() { return 7; }  
  
}
```

equals nicht überschrieben. hashCode korrekt (nicht sinnvoll)
überschrieben

- equals aus der Klasse Object vergleicht die Referenzen

... main...

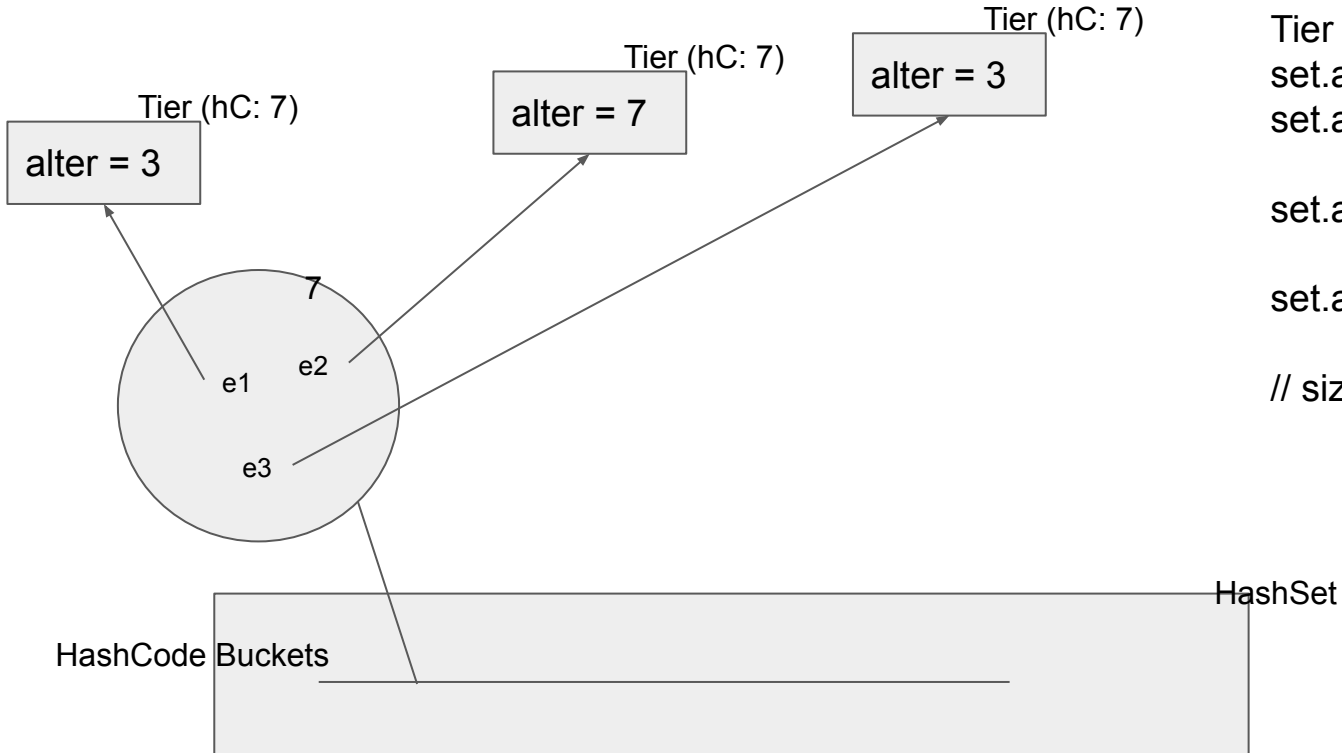
```
Set set = new HashSet();
```

```
Tier t1 = new Tier(3);  
set.add(t1);  
set.add(t1);
```

```
set.add(new Tier(7));
```

```
set.add(new Tier(3));
```

```
// size = 3
```



... main...

```
Set set = new HashSet();
```

```
Tier t1 = new Tier(3);  
set.add(t1);  
set.add(t1);
```

```
set.add(new Tier(7));
```

```
set.add(new Tier(3));
```

```
// size = 3
```

Version 3

```
class Tier {  
    int alter;  
  
    Tier(int alter) { this.alter = alter; }  
  
    public int hashCode() { return 7; }  
  
    public boolean equals(Object obj) {  
        return ((Tier)obj).alter == alter;  
    }  
}
```

sowohl equals als auch hashCode überschrieben.
hashCode ist korrekt (nicht sinnvoll) überschrieben.

... main...

```
Set set = new HashSet();
```

```
Tier t1 = new Tier(3);  
set.add(t1);  
set.add(t1);
```

```
set.add(new Tier(7));
```

```
set.add(new Tier(3));
```

```
// size = 2
```


die sind mit der neuen equals gleich

... main...

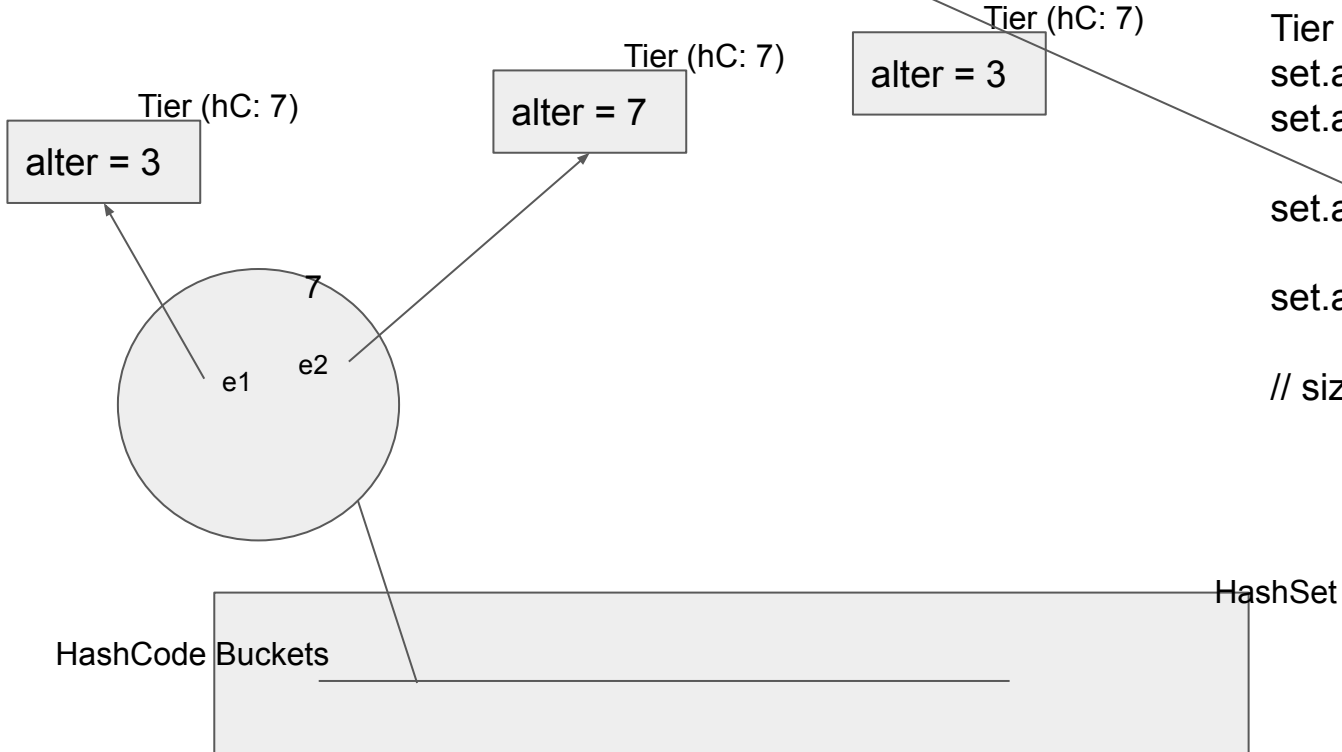
```
Set set = new HashSet();
```

```
Tier t1 = new Tier(3);  
set.add(t1);  
set.add(t1);
```

```
set.add(new Tier(7));
```

```
set.add(new Tier(3));
```

```
// size = 2
```



Version 4

```
class Tier {  
    int alter;  
  
    Tier(int alter) { this.alter = alter; }  
  
    public int hashCode() {  
        return alter;  
    }  
  
    public boolean equals(Object obj) {  
        return ((Tier)obj).alter == alter;  
    }  
}
```

sowohl equals als auch hashCode überschrieben.
hashCode ist sinnvoll überschrieben.

... main...

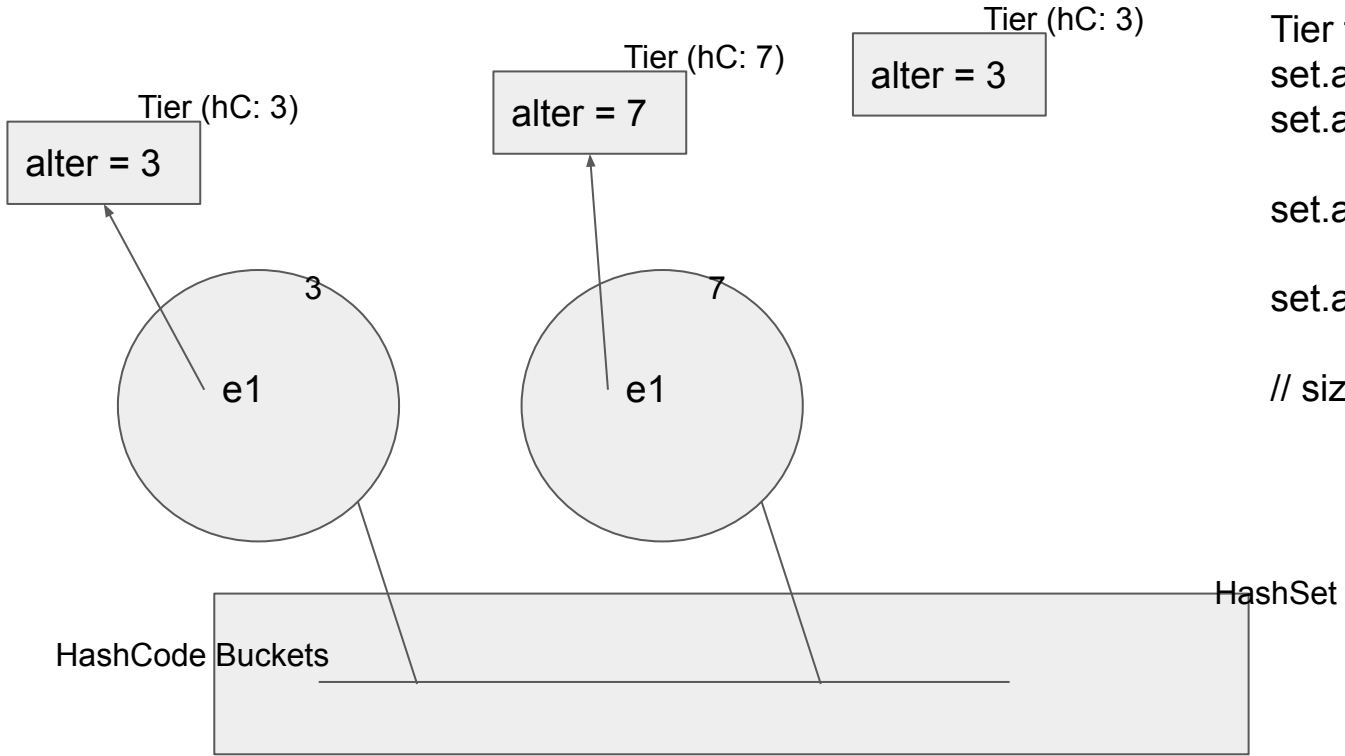
```
Set set = new HashSet();
```

```
Tier t1 = new Tier(3);  
set.add(t1);  
set.add(t1);
```

```
set.add(new Tier(7));
```

```
set.add(new Tier(3));
```

```
// size = 2
```



... main...

```
Set set = new HashSet();
```

```
Tier t1 = new Tier(3);  
set.add(t1);  
set.add(t1);
```

```
set.add(new Tier(7));
```

```
set.add(new Tier(3));
```

```
// size = 2
```