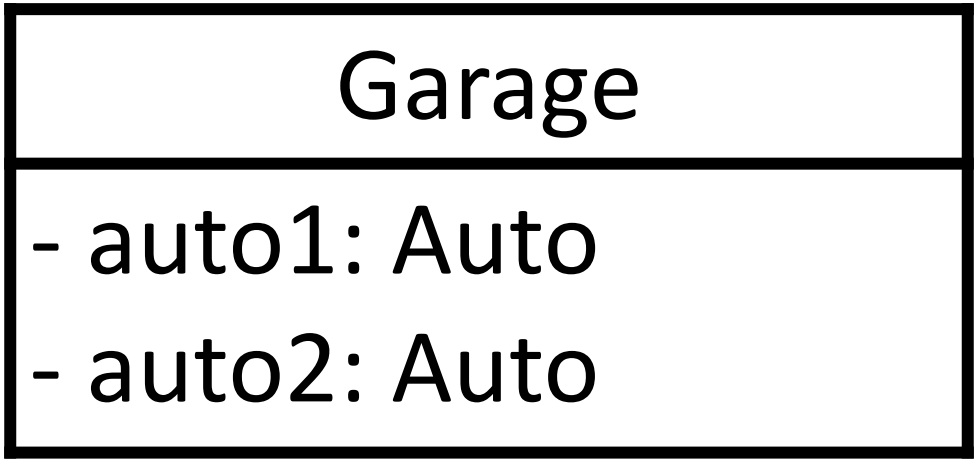
Felder in der Informatik

Quelle: http://www.hotelducommerce.lu/uploads/pics/kornfeld_1024x768_03.jpg, aufgerufen am 16.01.2017

Garage

Stelle dir vor, du programmierst eine Garage.

Jedes Objekt der Klasse Garage referenziert 2 Objekte der Klasse Auto:



bisher...

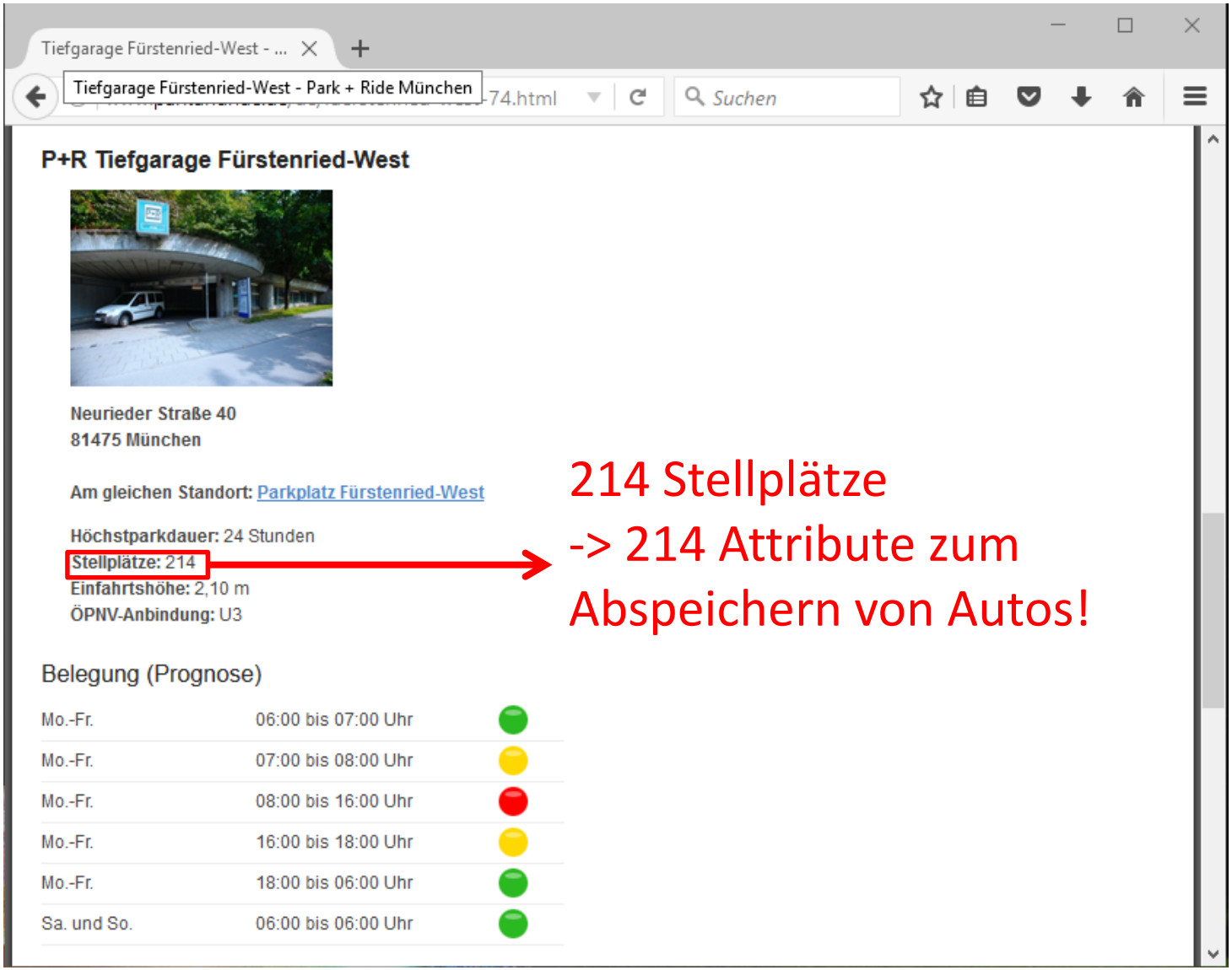
… wird für jedes Objekt eine eigene Variable deklariert.

Kleine Anzahl von Objekten -> überschaubarer Aufwand

Große Anzahl von Objekten -> großer Aufwand

Szenario: *Unsere* ***Garage*** *wird zum* ***Parkhaus***

Beispiel Tiefgarage Fürstenried-West



Lösung:

Wir benötigen eine ***„Datenstruktur“***!

1. Sie soll ***viele Elemente gleichen Datentyps*** fassen können.

2. Der ***Zugriff auf jedes einzelne Element*** muss möglich sein.

Deklarien von Feldobjekten

Notation in Schreibweise analog zu normalen Objekten, nur mit [] hinter dem Datentyp:

```
Auto[] autos; //Feld mit Elementen vom Typ Auto
```

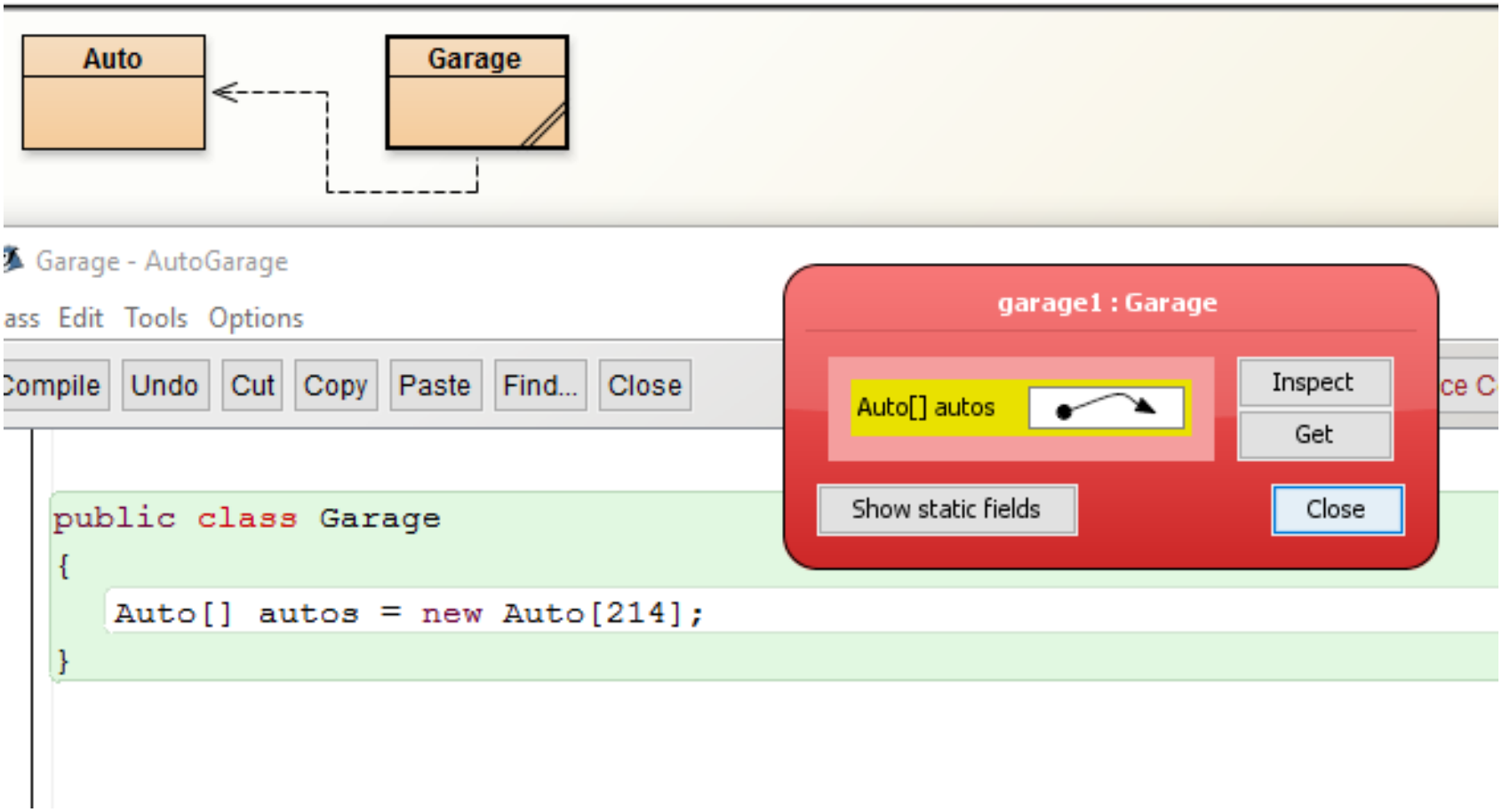
Die angehängten Klammern enthalten die Anzahl der Feldelemente.

```
autos = new Auto[214]; //Feld mit 214 Elementen
```

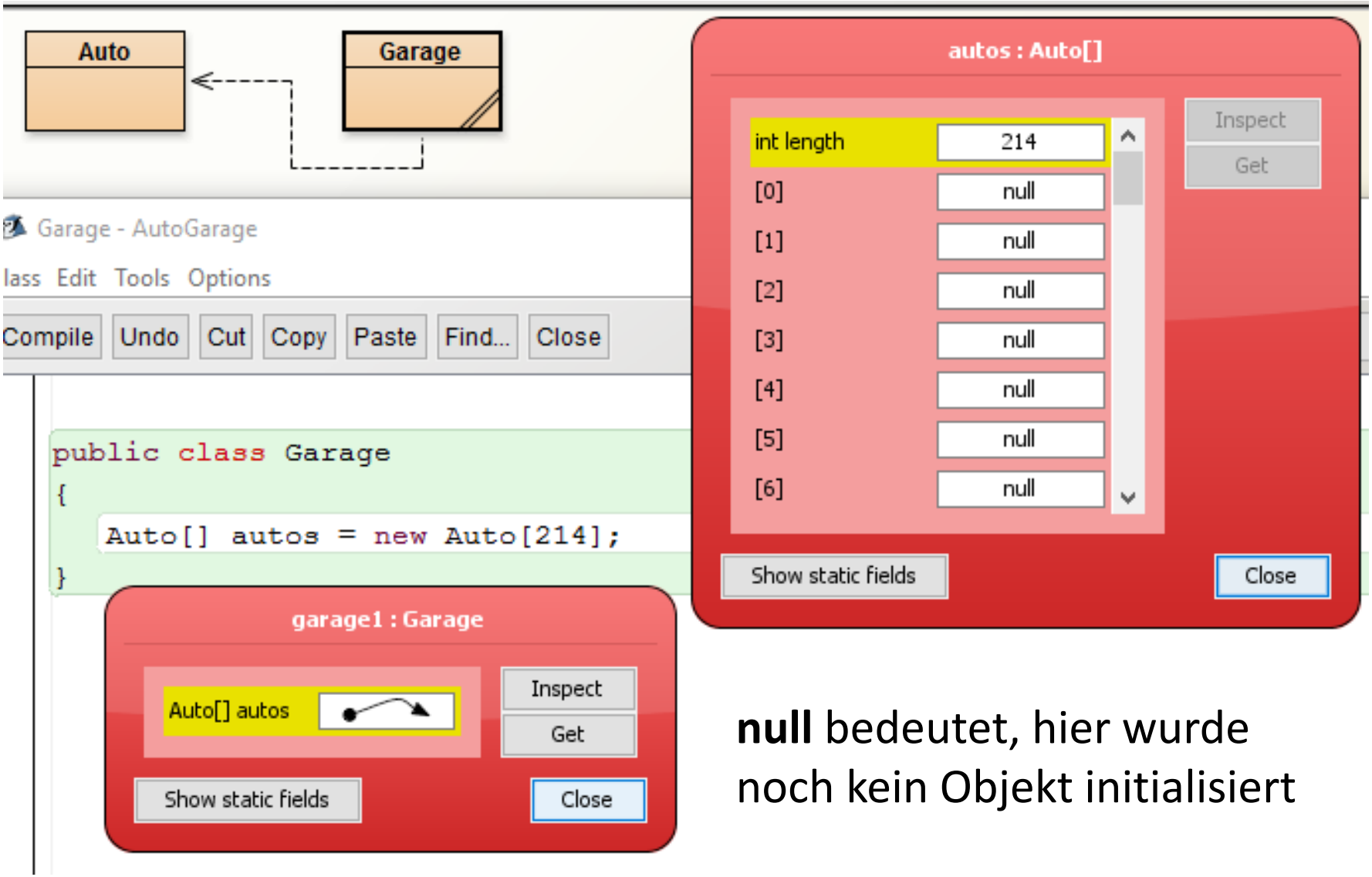
Zugriff auf die Länge des Elements:

```
int anzahlParkplaetze = autos.length;
```

Mal sehen, was sich getan hat



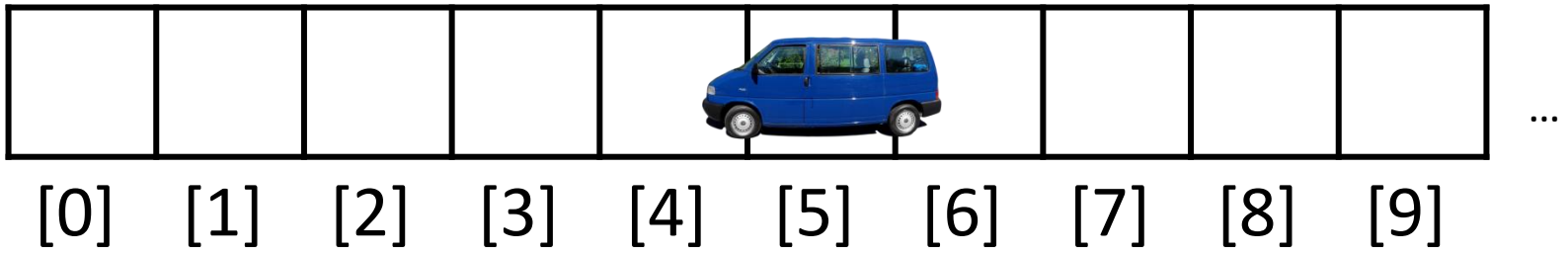
Mal sehen, was sich getan hat



null bedeutet, hier wurde noch kein Objekt initialisiert

Zugriff auf die Elemente

Zum ***Initialisieren*** und ***Verwenden*** der einzelnen Elemente können wir über deren ***Position*** Zugriff erlangen:



Initialisieren des Elements mit Index 5:

```
autos[5] = new Auto(„VW“, „T4“, 85, 1997); //Element 5 steht an 6. Stelle!
```

ACHTUNG! Der Informatiker zählt 0, 1, 2, 3…

Die Länge des oben sichtbaren Elements ist 10, es beinhaltet die Elemente 0 bis 9!

Übung

a) ***Instanziiere*** ein Array vom typ int mit 7 Feldern. Gib jedem Element den Wert seines Indizes. *(Index = Nummer des Elements)*

b) ***Verwende*** eine While-Schleife für eine automatische Zuweisung der Werte.

c) ***Implementiere*** eine Methode, die automatisch alle Elemente des Arrays auf der Konsole ausgibt.