

Datenstrukturen

Was sind Datenstrukturen?



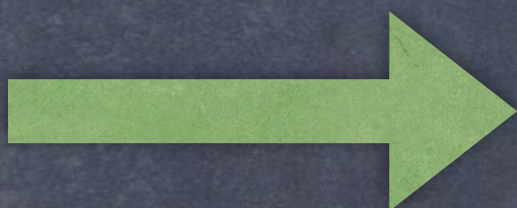
- Daten in einer bestimmten Art und Weise angeordnet
- Effiziente Verwaltung von Daten



```
String name1 = "user1";  
String name2 = "user12";  
String name3 = "user123";  
String name4 = "user1234";  
String name5 = "user12";  
String name6 = "user113";  
String name7 = "user114";  
String name8 = "user15";  
String name9 = "user16";  
String name10 = "user100";  
String name11 = "user1000";  
String name12 = "user12345";  
String name13 = "user123456";
```

```
String name100 = "user4567";
```

Daten in eine Struktur einfügen



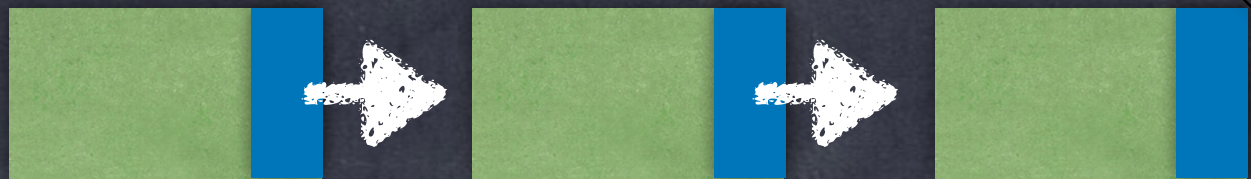
```
Post post1 = new Post();  
Post post2 = new Post();  
Post post3 = new Post();  
Post post4 = new Post();  
Post post5 = new Post();  
Post post6 = new Post();  
Post post7 = new Post();  
Post post8 = new Post();  
Post post9 = new Post();  
Post post10 = new Post();  
Post post11 = new Post();  
Post post12 = new Post();  
Post post13 = new Post();
```

```
Post post1000 = new Post();
```

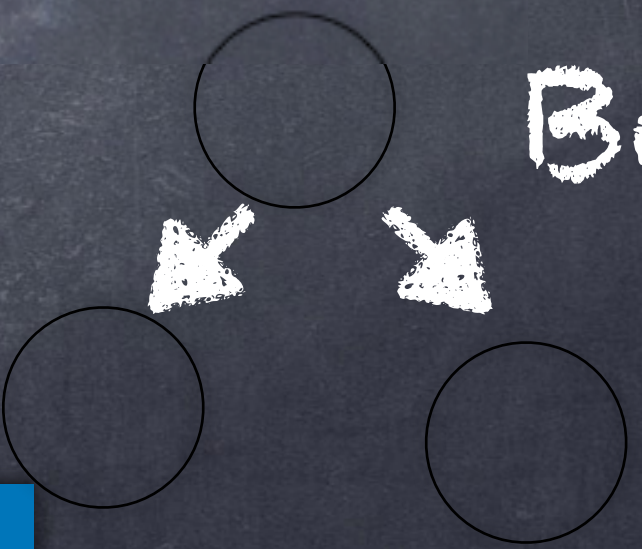
Index	Inhalt
[0]	
[1]	
[2]	
[3]	

Array

Listen



Bäume



Array

Index	Inhalt
[0]	
[1]	
[2]	
[3]	
[4]	
[5]	
[6]	
[7]	
[8]	
[9]	

Details:

- Zugriff über Index
- Feste Größe
- Nur ein Datentyp möglich

```
int[] arr = new int[10];
```



Größe muss angegeben werden.

ArrayList

Details:

- Zugriff über Index
- Beliebige Größe
- Beliebige Datentypen „möglich“
- Beliebige Objekte
- Dynamisch, Objekte können jederzeit gelöscht oder hinzugefügt werden

```
ArrayList<Object> list = new ArrayList<Object>();  
  
list.add(Double.valueOf(20.4));  
list.add("Einkaufen");  
list.add(Integer.valueOf(100));  
list.add("Schlüssel suchen");  
  
for (Object object : list) {  
    System.out.println(object);  
}
```

Ausgabe:

```
20.4  
Einkaufen  
100  
Schlüssel suchen
```

Index	Inhalt
[0]	20.4
[1]	Einkaufen
[2]	100
[3]	Schlüssel suchen
[4]	1992

Verkettete Liste

Details:

- Ein Listenelement nacheinander
- Beliebige Größe
- Beliebige Datentypen „möglich“
- Beliebige Objekte
- Dynamisch, Objekte können jederzeit gelöscht oder hinzugefügt werden

Verkettete Liste



```
LinkedList<String> list = new LinkedList<String>();  
list.add("01233");  
list.add("3456");  
list.add("8765");
```

ArrayList

Index	Inhalt
[0]	20.4
[1]	Einkaufen
[2]	100
[3]	Schlüssel suchen



Stack (Stapel)

Details:

- LIFO = Last in first out
- Dynamisch, Objekte können jederzeit gelöscht oder hinzugefügt werden



www.youtube/video4.de

www.youtube/video3.de

www.youtube/video2.de

www.youtube/video1.de

post1000

post999

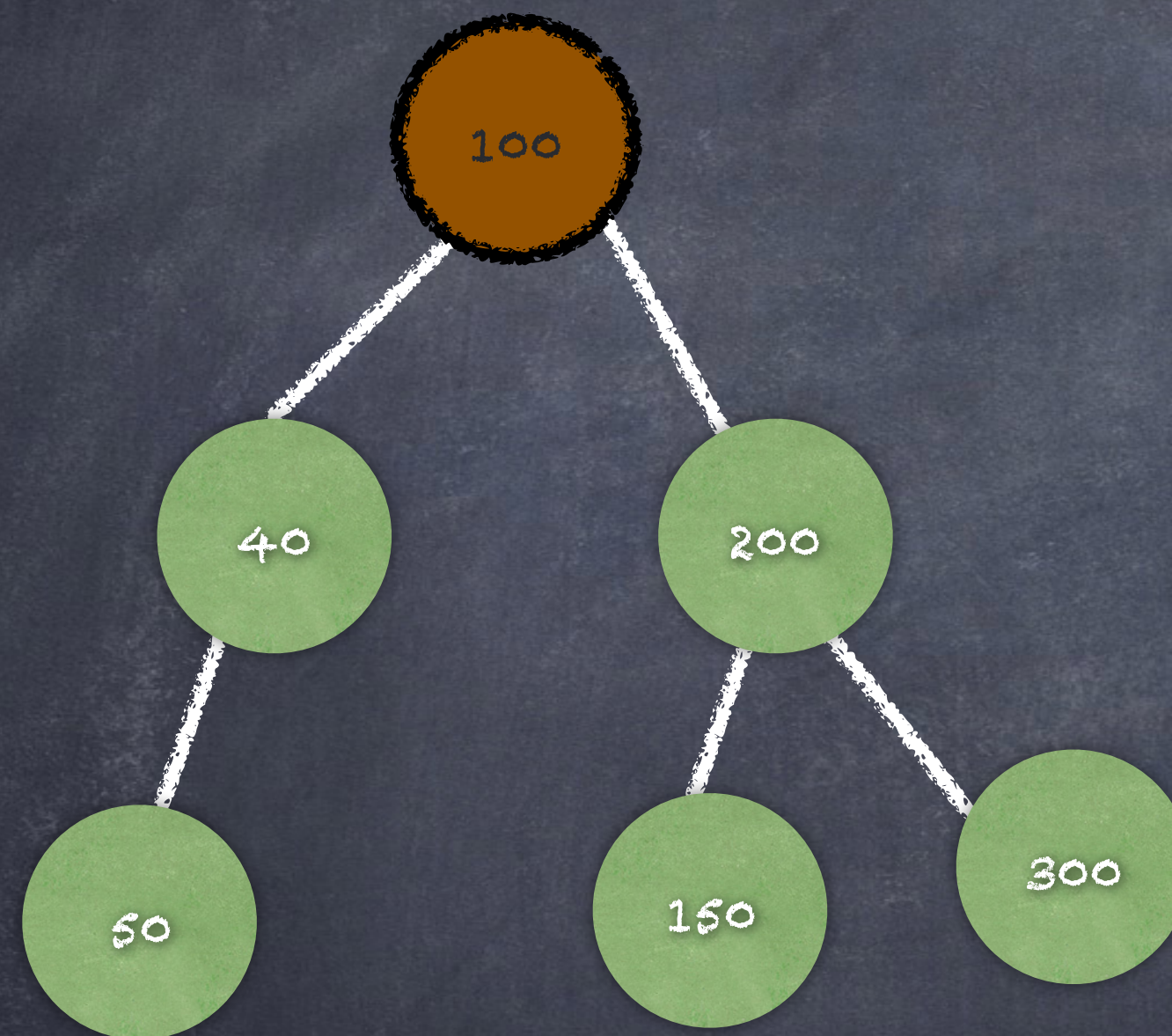
post998

...

post1

Stack (Stapel)

Bäume



Details:

- Zwei (Binär) oder beliebige Nachfolger
- Es gibt immer nur eine Wurzel!
- Besteht aus Knoten (Kreise) und Kanten (Striche)
- Dynamisch, Objekte können jederzeit gelöscht oder hinzugefügt werden

Vergleich mit:

ArrayList	
Index	Inhalt
[0]	20.4
[1]	Einkaufen
[2]	100
[3]	Schlüssel suchen



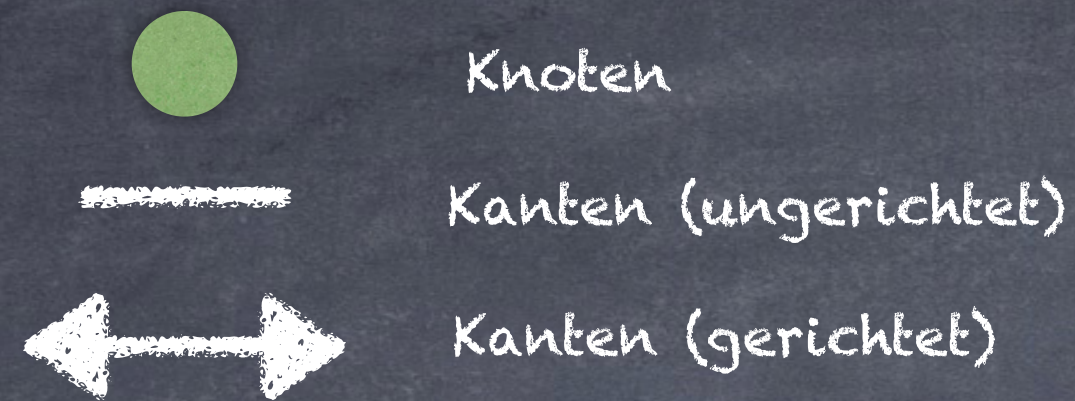
Bäume



Bäume

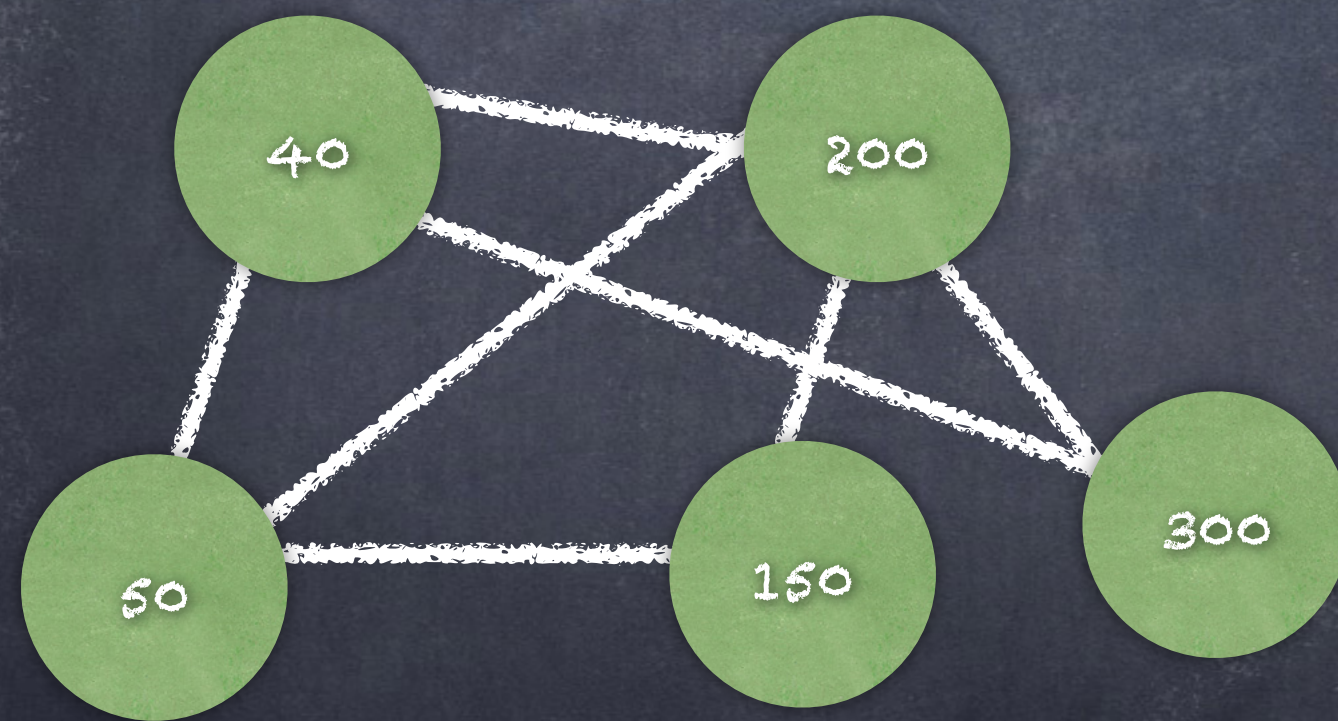


Graphen

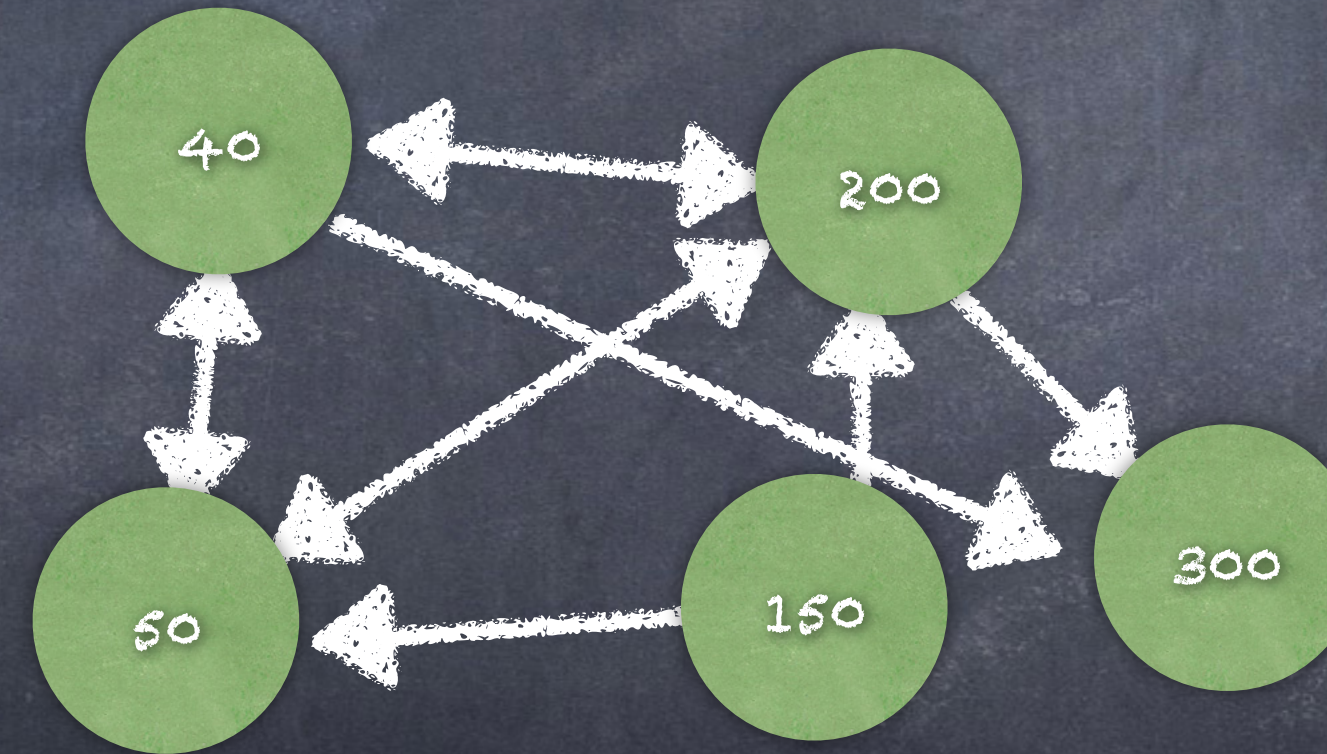


Details:

- Wie ein Baum, nur das jeder mit jedem kann d.h. ein Graph besteht aus einer Menge von Knoten und Kanten
- Ein Knoten kann mehrere Kanten (Verbindungen) haben



Ungerichteten Graphen



Gerichteten Graphen