

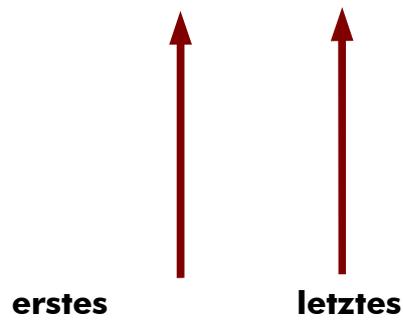
## Die Warteschlange ist eine besondere Form der Liste

Es soll ähnlich aufgebaut sein wie der eben besprochene Keller, nur soll er nach dem FIFO-Prinzip funktionieren, d.h. das zuerst eingefügte Element soll als erstes wieder herausgenommen werden.

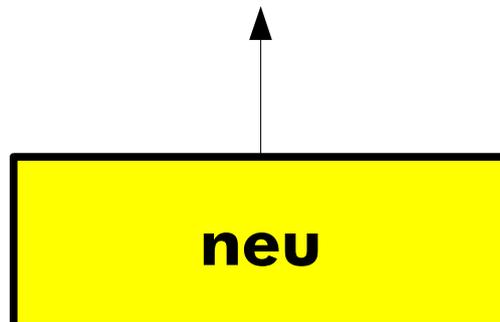
Zu besserer Unterscheidung verwenden wir andere Methodennamen, nämlich PUT und GET.

Der Konstruktor einer LISTE erzeugt jetzt zwei Pointer auf **null**. Später soll der eine auf den Anfang der Liste und der andere auf das Ende der Liste zeigen. Ein Element selbst besteht wie vorher aus einem Datenfeld, z.B. einem ganzzahligen Wert und aus einem Pointer auf einen Vorgänger.

1) Zeiger (Kopf) zeigen beide Pointer auf **null**



2) Ein Element wird erzeugt, es hat einen Wert und einen Pointer auf ein vorheriges Element; der Konstruktor legt den Pointer erst einmal auf **null** fest.

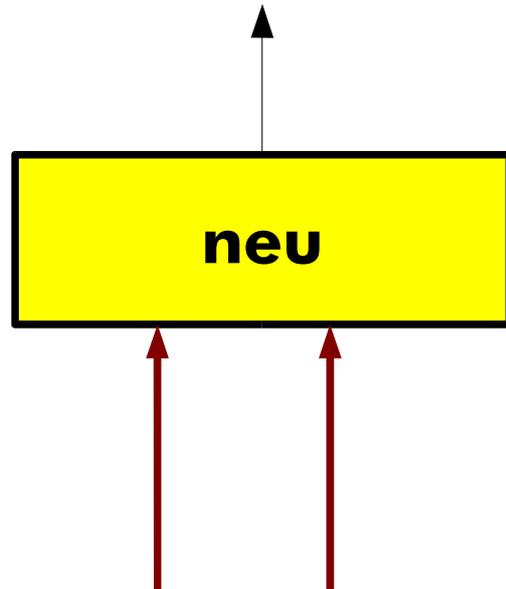


3) Das Element wird an die Liste angefügt mit der Methode `append($neu)`

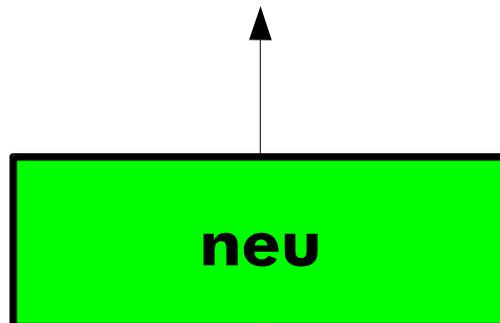
```
// hat hier noch keine bedeutung, da $this->kopf null ist!  
$neu->voriges=$this->kopf;
```

```
// der Kopf wird auf das Element umgebogen  
if ($this->erstes==null) $this->letztes=$neu;  
$this->erstes=$neu;
```

3) Damit entsteht folgendes Konstrukt

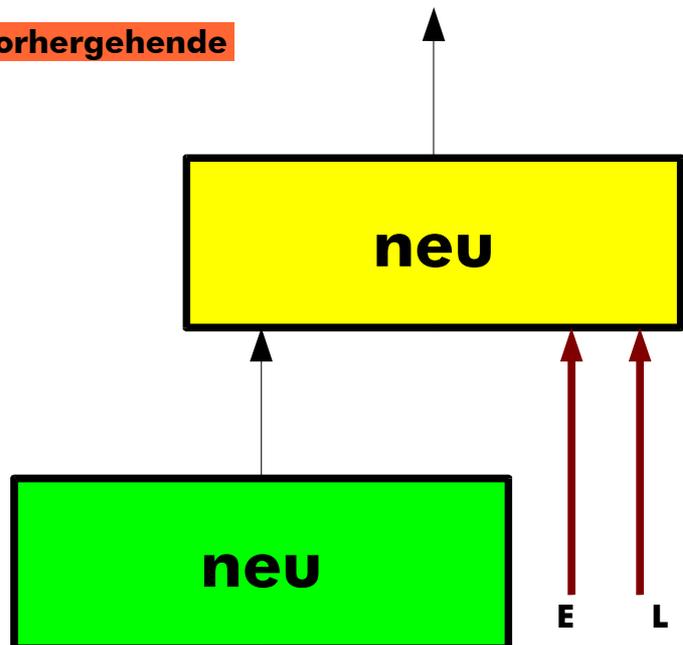


4) Ein weiteres Element wird erzeugt

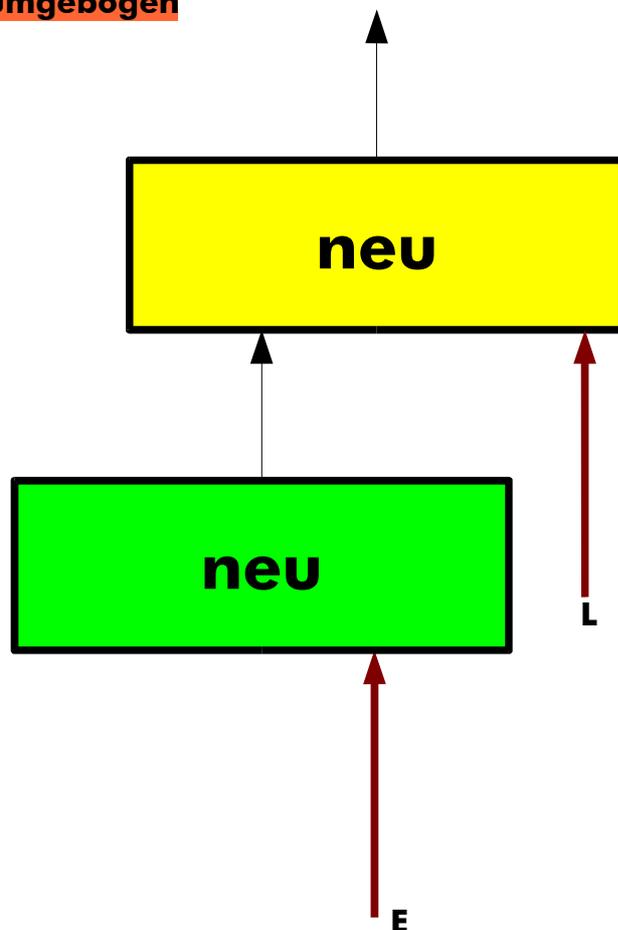


5) Auch hier wird der Append-Befehl durchgeführt:

```
// das neue Element zeigt auf das vorhergehende  
$neu->voriges=$this->erstes;
```



```
// der Kopf wird auf das Element umgebogen  
$this->erstes=neu;
```

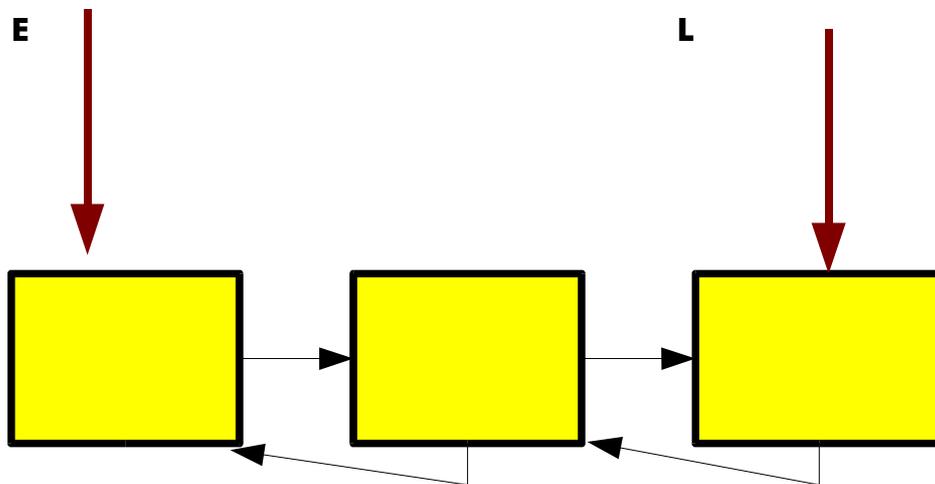


**Ergebnis:**

**Das neue Element ist immer das, auf den das erste zeigt.**

**Wenn ich mir mit der Funktion PUT() ein neues Element einfüge, so steht dies am neuen Anfang. Die Methode GET() muss nun einfach das Element zurückliefern, das als letztes eingetragen ist. Ein Problem ergibt sich dann aber daraus, dass der Zeiger auf das letzte Element auf das vorletzte "umgebogen" werden muss. Dafür muss mit Hilfe eines Schleppezigers die Liste von vorne bis hinten durchgezählt werden.**

**Eine andere Möglichkeit bietet sich an, doppelt verkettete Liste zu bilden, dabei hat ein Element jeweils einen Vorgänger und einen Nachfolger.**



**Wenn Elemente hintereinander gefügt werden**