

Die Methode **hintenEntnehmen** der einfach verketteten Liste

entfernen eines Knoten mit
einem bestimmten Inhalt



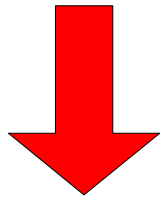
entnehmen eines Knoten mit
einem bestimmtem Inhalt



hintenEntnehmen

Entfernen eines Knoten mit einem bestimmten Inhalt

```
class Liste
{
...
    public void entfernen(String datenwert)
    {
        erster.entfernen(datenwert);
    }
...
}
```



```
abstract class Listenelement
{
...
    public abstract void entfernen(String suchwert);
...
}
```

```
class Datenknoten extends Listenelement
{
    ...
    public void entfernen(String suchwert)
    {
        // wenn datenwert von inhalt mit suchwert überein-
        // stimmt, soll der Knoten entfernt werden
    }
    ...
}

class Abschluss extends Listenelement
{
    ...
    public void entfernen(String suchwert)
    {
        // kein inhalt, keine Übereinstimmung, nichts passiert
    }
    ...
}
```

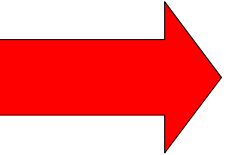
```
class Datenknoten extends Listenelement
{
    ...
    public Listenelement entfernen(String suchwert)
    {
        if(inhalt.datenwertGeben() == suchwert)
            { return naechster;}
        else
            { naechster = naechster.entfernen(suchwert);
              return this; }
    }
    ...
}

class Abschluss extends Listenelement
{
    ...
    public Listenelement entfernen(String suchwert)
    {
        return this;
    }
    ...
}
```

Entnehmen eines Knoten mit einem bestimmten Inhalt

```
class Liste
{
...
    public Datenelement entnehmen(String datenwert)
    {
        // zuvor muss der alte Knoteninhalt fest-
        // gehalten werden
        erster.entfernen(datenwert);
        // um jetzt zurückgegeben werden zu können
    }
...
}
```

```
class Liste
{
...
    public Datenelement entnehmen(String datenwert)
    {
        Datenelement alterKnoteninhalt =
            datenknotenGeben(datenwert).inhaltGeben();
        erster.entfernen(datenwert);
        return alterKnoteninhalt;
    }
...
    public Datenknoten datenknotenGeben(String datenwert)
    {
        return erster.datenknotenGeben(datenwert);
    }
}
```



```
class Datenknoten extends Listenelement
{
    ...
    public Datenknoten datenknotenGeben(String suchwert)
    {
        if(inhalt.datenwertGeben() == suchwert)
        { return this;}
        else
        {
            return naechster.datenwertGeben(suchwert);
        }
    }
    ...
}

class Abschluss extends Listenelement
{
    ...
    public Datenknoten datenknotenGeben(String suchwert)
    {
        return null;
    }
    ...
}
```

Entnehmen des letzten Knoten aus der Liste
`public Datenelement hintenEntnehmen()`

- Inhalt des letzten Knoten wird für entnehmen benötigt
- Suche nach inhaltLetzter mit inhaltLetzterGeben
- Datenwert von inhaltLetzter kann der Methode entnehmen übergeben werden
- Rückgabewert von entnehmen(inhaltLetzter) liefert Inhalt des ehemals letzten Knoten


```
class Datenknoten extends Listenelement
{
...
    public Datenelement inhaltLetzterGeben(Datenelement
                                           aktInhalt)
    {
        return naechster.inhaltLetzterGeben(inhalt);
    }
...
}
```

```
class Abschluss extends Listenelement
{
...
    public Datenelement inhaltLetzterGeben(Datenelement
                                           aktInhalt)
    {
        return aktInhalt;
    }
...
}
```

Auch für den ersten Aufruf der Methode `inhaltLetzterGeben` innerhalb der Methode `hintenEntnehmen` der Klasse `Liste` wird ein Eingabewert benötigt.

Dazu wird einmalig der Inhalt des ersten Knoten mittels `erster.inhaltGeben()` verwendet:

```
class Liste
{
    ...
    public Datenelement hintenEntnehmen()
    {
        Datenelement inhaltLetzter =
            erster.inhaltLetzterGeben
                (erster.inhaltGeben());
        return
            entnehmen(inhaltLetzter.datenwertGeben());
    }
    ...
}
```