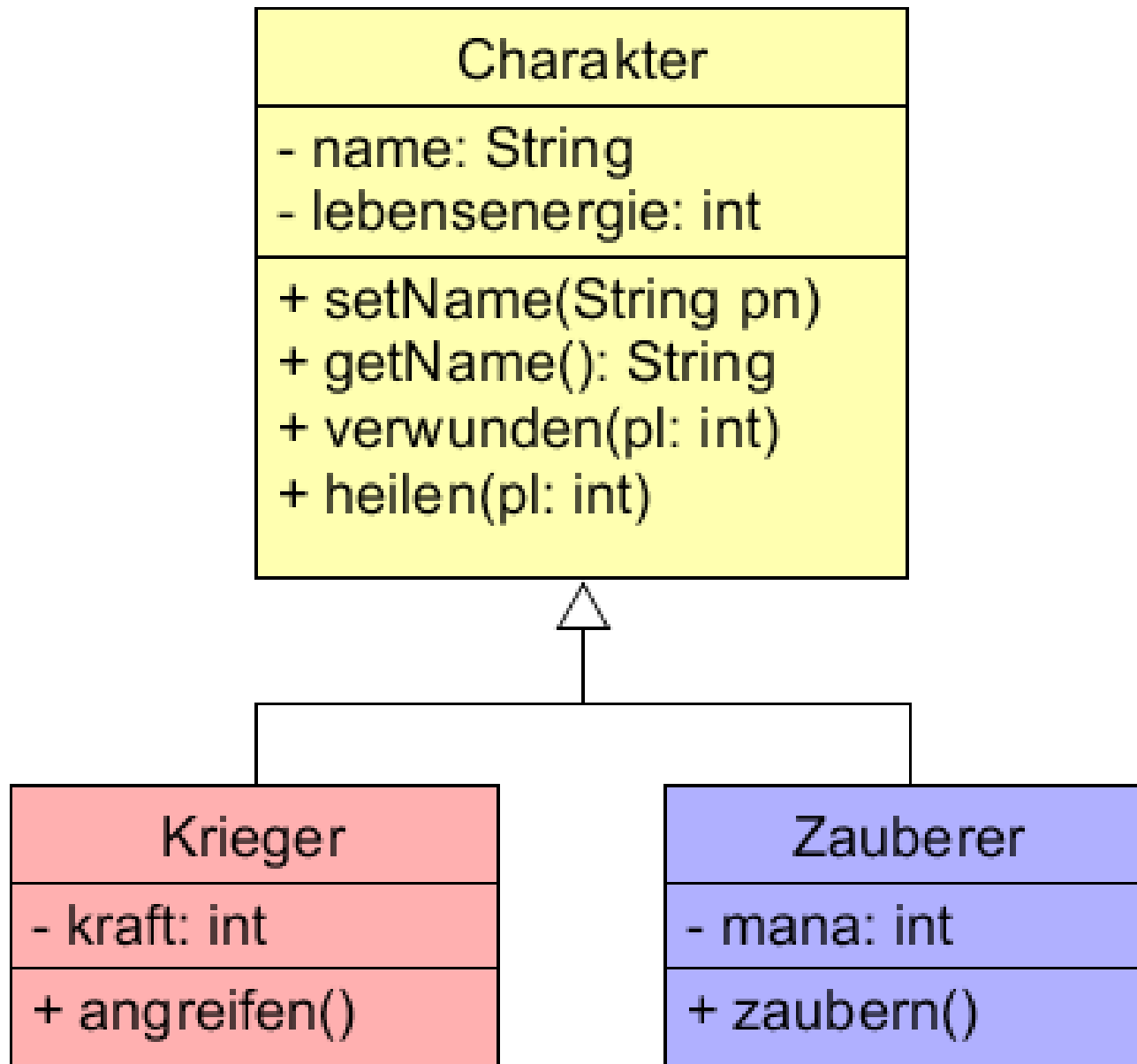


**OOP mit Java**

**Vererbung  
im Quellcode**

# Beispiel: Rollenspiel



# Quellcode Oberklasse

**keine Änderungen**

```
public class Charakter
{
    private String name;
    private int lebensenergie

    public void setName(String pn)
    {
        name = pn;
    }
    ...
}
```

# Quellcode Unterklasse

## Schlüsselwort „extends“

```
public class Krieger extends Charakter
{
    private int kraft;

    public void angreifen()
    {
        ...
    }
    ...
}
```

Die Klasse Krieger „erweitert“ die Klasse Charakter um neue Attribute und Methoden ...

# Quellcode Unterklasse

## Schlüsselwort „extends“

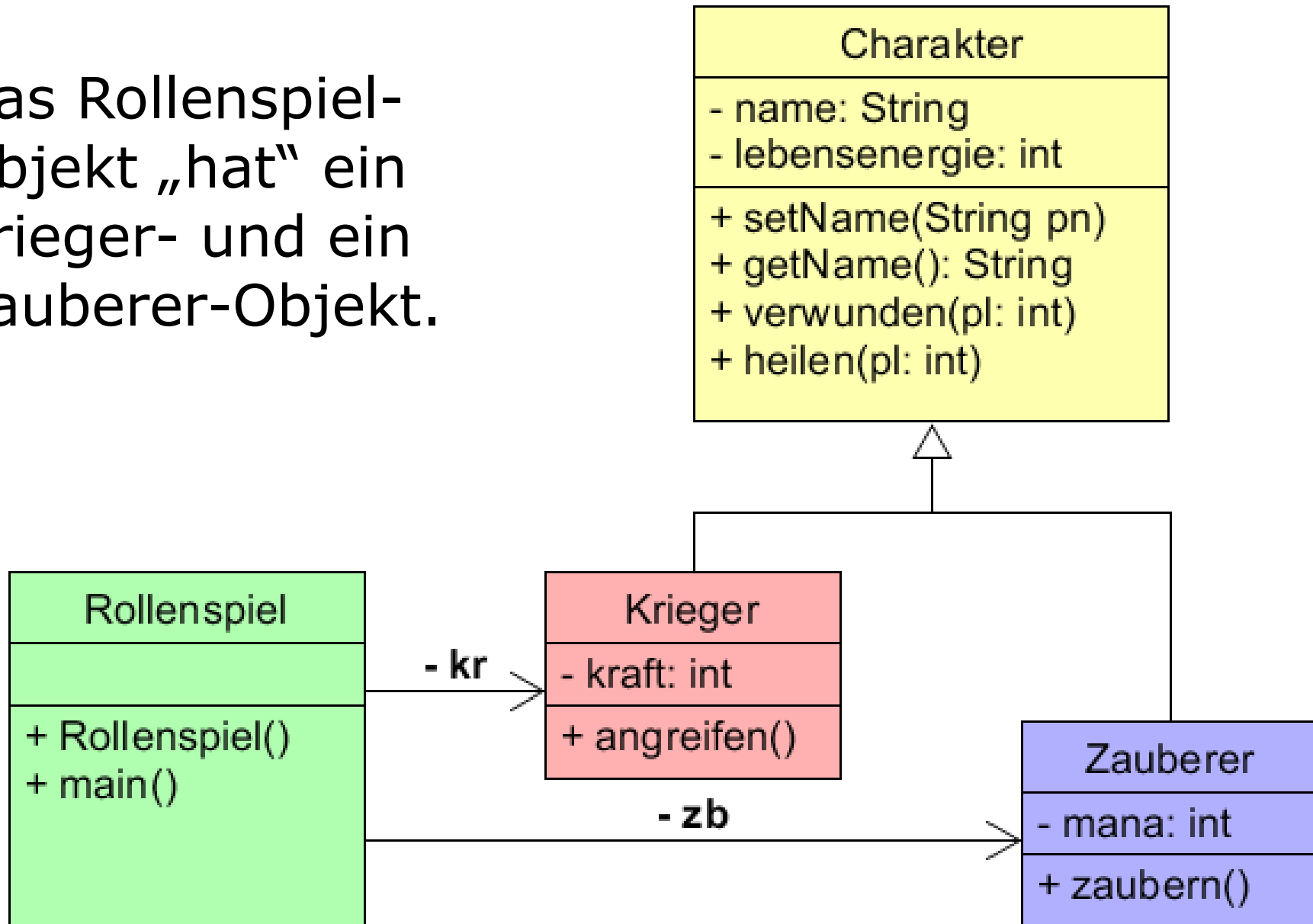
```
public class Zauberer extends Charakter
{
    private int mana;

    public void zaubern()
    {
        ...
    }
    ...
}
```

... und ebenso die Klasse Zauberer.

# Hauptklasse

Das Rollenspiel-Objekt „hat“ ein Krieger- und ein Zauberer-Objekt.



# Quellcode Hauptklasse

```
public class Rollenspiel
{
    private Krieger kr;
    private Zauberer zb;

    public Rollenspiel()
    {
        kr = new Krieger();
        zb = new Zauberer();
    }
    ...
}
```

Objekte von Unterklassen werden genauso deklariert und erzeugt wie andere auch.

# Quellcode Hauptklasse

```
...
public void main()
{
    kr.setName("Angor");           // → Charakter
    kr.verwunden(10);             // → Charakter
    kr.angreifen();               // → Krieger

    zb.setName("Belsatar");       // → Charakter
    zb.zaubern();                 // → Zauberer
}
...
```

Es können Methoden der Oberklasse und der Unterklasse aufgerufen werden.



# Autor / Quellen

Autor:

- Christian Pothmann (cpothmann.de)  
Freigegeben unter CC BY-NC-SA 4.0, Juni 2021

