

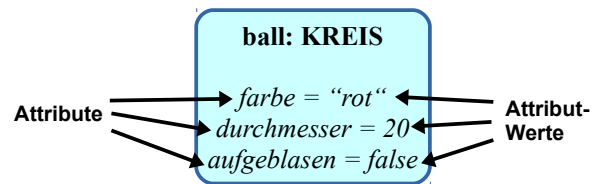
# Referenz-Attribute

**Primitives Attribut** bedeutet **Eigenschaft**.

Gängige Datentypen für primitive Attribute sind

boolean (true / false) , int (ganze Zahl) ,

String (Text) , float (Kommazahl) , ...



**Referenz-Attribut** bedeutet **Bauteil**.

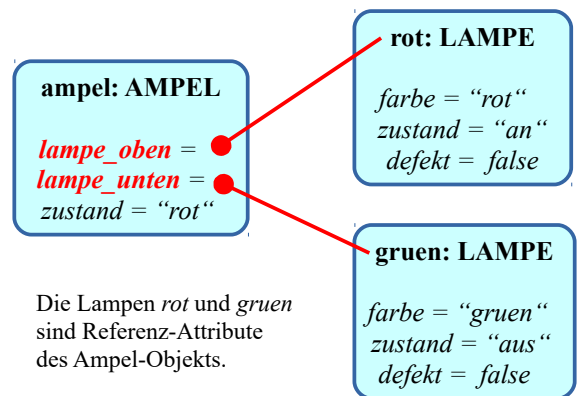
Datentypen von Referenz-Attributen sind Klassen

und damit Nomen wie KREIS, BILD, TEXT, ... .

Deshalb muss man ein Referenz-Attribut beim

Initialisieren zuerst erzeugen (z.B. `new KREIS()`).

Referenz-Attribute sind also eigenständige Objekte.



## Code-Beispiel:

```
public class AMPEL
{
    private LAMPE lampe_oben; ← Referenz-Attribute deklarieren
    private LAMPE lampe_unten; ← Referenz-Attribute deklarieren
    private String zustand; ← primitives Attribut deklarieren

    public AMPEL( int x , int y )
    {
        this.lampe_oben = new LAMPE( x , y-50 , "rot" ); ← Referenz-Attribute erzeugen
        this.lampe_unten = new LAMPE( x , y+50 , "gruen" ); ← Referenz-Attribute erzeugen
        this.zustand = "aus"; ← primitives Attribut initialisieren
    }

    public void rot()
    {
        this.lampe_oben.an(); ← mit Referenz-Attributen arbeiten
        this.lampe_unten.aus(); ← mit Referenz-Attributen arbeiten
        this.zustand = "rot"; ← mit primitivem Attribut arbeiten
    }

    . . .
}
```