

# Objektorientierte Programmierung mit einem Raytracer

## Methoden

Wir können uns mit der Methode `tr.getObjekte()`; alle Objekte in unserer Szene als Array holen. Hierfür müssen wir das Paket `raytracing.objekt.Objekt` importieren. Anschließend reicht der Aufruf

```
Objekt[] obj = tr.getObjekte();
```

Listing 1: Holen der Objekte

um alle in der Szene befindlichen Objekte in einem Array zu speichern.

Für jedes Objekt `obj[i]` gibt es dann wiederum eine Methode `treffer(Gerade g)`, welche testet, ob die Gerade `g` das Objekt schneidet und dann einen entsprechenden `boolean`-Wert zurückgibt.

Die Gerade `g` erhalten wir wiederum über die Methode `tr.getGerade(int x,int y)` welche `x`- und `y`-Koordinate eines Pixels annimmt und als Ergebnis eine `Gerade` liefert. Dazu müssen wir jedoch das Paket `raytracing.math.Gerade` importieren.

## 1. Aufgabe

Ziel wird es sein, die Methode `tr.trifft(int x,int y)` so nachzubilden, dass wir diese später erweitern können.

Lege dazu im OOP-Projekt in der `Start`-Klasse eine neue Methode `trifft` an:

- die Methode soll wie bisher `public static` sein
- als Ergebnis soll die Methode einen `boolean`-Wert zurückgeben
- es werden 3 Parameter angenommen:
  - im ersten Parameter soll der `Tracer tr` übergeben werden
  - im zweiten Parameter soll die `x`-Koordinate übergeben werden
  - im dritten Parameter soll die `y`-Koordinate übergeben werden

Diese können wir anschließend in unserer `main`-Methode benutzen und damit die `tr.trifft`-Methode ersetzen.

Auf die gleiche Art können wir die Methoden `tr.rot`, `tr.gruen` und `tr.blau` ersetzen. Den Farbwert eines einzelnen Objektes in unserer Szene bekommen wir über die Methoden `obj[i].rot()`, `obj[i].gruen()` und `obj[i].blau()`.