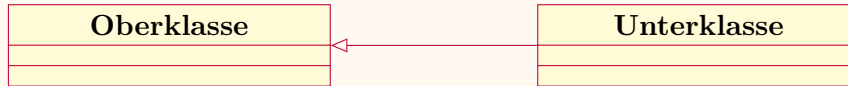


1.1 🖥️ Schießen aber bitte einfach!

Vererbung

Eine Klasse kann alle Methoden und Attribute einer anderen Klasse **erben**. Vererbung wird in Java mit dem Schlüsselwort **extends** Oberklasse in der Klassendefinition der Unterklasse ausgedrückt.



Im Klassendiagramm wird die Vererbung durch eine unausgefüllte Pfeilspitze ausgedrückt.

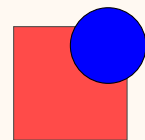
1. Erzeuge eine neue Klasse `Projektil` als Unterklasse von `Actor` (wähle kein Bild aus).
2. Implementiere die Bewegung sowie das Entfernen des Projektils, falls es einen Rand berührt, in der `act`-Methode der Klasse `Projektil` (vgl. deine `Laser` Implementierung).
3. Definiere die Klasse `Laser` als Unterklasse von `Projektil` und initialisiere die Geschwindigkeit mit dem Wert 6 im Konstruktor der Klasse `Laser`.
4. Implementiere die Schüsse der Aliens in der Klasse `Lazer` (wähle das Bild `laser2`) und initialisiere die Geschwindigkeit mit dem Wert `-5`.
5. Passe die Methode `schiessen` der Klasse `Alien` entsprechend an, sodass neue `Lazer` Objekte erzeugt werden.

1.2 🖥️ Oh nein getroffen!

Um zu überprüfen, ob ein Alien oder das Raumschiff von einem feindlichen Projektil getroffen wurden, muss überprüft werden, ob die Objekte sich berühren.

Kollision

Von einer Kollision spricht man, wenn zwei Objekte sich **überschneiden**. Eine solche Überschneidung zu erkennen kann je nach Form der Objekte kompliziert sein. In *Greenfoot* bietet dazu die Klasse `Actor` die Methode `isTouching(Class)` an.



1. Implementiere in der Klasse `Raumschiff` die Methode `pruefeKollision()`, in der du mittels `isTouching(Lazer.class)` überprüfst, ob das Raumschiff getroffen wurde und gegebenenfalls die Methode `verloren` der Welt aufrufst (`((MyWorld)getWorld()).verloren();`).
Ergänze die Methode `act` der Klasse `Raumschiff` um einen Aufruf der Methode `pruefeKollision`.
2. Implementiere in der Klasse `Alien` die Methode `pruefeKollision()`, in der du mittels `isTouching(Lazer.class)` überprüfst, ob das Alien getroffen wurde. Falls das Alien getroffen wurde, entferne das Projektil (`removeTouching(Lazer.class);`), sowie das Alien selbst (`getWorld().removeObject(this);`).
Spiele zusätzlich einen Explosions-Ton ab (`Greenfoot.playSound("Explosion.wav");`).
3. Wurden alle Aliens zerstört?
 - a) Implementiere in der Klasse `MyWorld` eine Methode `reduziereAlienAnzahl`, in der du die `alienAnzahl` um 1 reduzierst.
 - b) Reduziere die Alien Anzahl in der Methode `pruefeKollision`, falls das Alien getroffen wurde (`((MyWorld)getWorld()).reduziereAlienAnzahl();`).

- c) Ergänze die Methode `act` der Klasse `MyWorld` um eine Überprüfung ob alle Aliens zerstört wurden und rufe gegebenenfalls die Methode `gewonnen` auf.

1.3 Die Erde ist verloren!

Das Spiel ist auch verloren, wenn eins der Aliens den unteren Bildschirmrand, die Erde, erreicht hat.

1. Implementiere eine Methode `pruefeSieg` in der Klasse `Alien`, die überprüft ob die aktuelle Y Position (`getY()`) größer oder gleich 590 ist. Falls ja soll die Methode `verloren` der Klasse `MyWorld` aufgerufen werde.
2. Rufe die Methode `pruefeSieg` in der Methode `act` des Aliens auf.

1.4 Für Schnelle: Tob dich aus!

Sei kreativ und erweitere das Spiel um weitere Level, andere Gegner, Projektile oder einen Highscore.