# Rennschnecke

1. Erstellen Sie eine Klasse "Rennschnecke".
2. Rennschnecken sollen folgende Eigenschaften (Attribute) besitzen:
   * einen Namen,
   * eine Rasse,
   * eine Maximalgeschwindigkeit (Veränderung der Strecke pro Schritt),
   * die Schnecke soll wissen welchen Weg sie bereits zurückgelegt hat.
3. Erstellen Sie für die Klasse Rennschnecke einen Konstruktor, der den Instanzvariablen beim Erstellen einer neuen Instanz (Objekt zu dieser Klasse) Werte zuweist.
4. Legen Sie in der Klasse "Rennschnecke" eine Methode "krieche()" an, welche die Schnecke abhängig von ihrer Maximalgeschwindigkeit eine zufällige Strecke weiter bewegt. Soll heißen: Sie kriecht eine zufällige Strecke größer null und kleiner ihrer Maximalgeschwindigkeit. Nimm als Zeiteinheit 1 an.
   * Tipp: Schauen Sie sich die Methode Math.random() aus der Java API an.
5. Legen Sie in der Klasse "Rennschnecke" eine Methode "public String toString()" an, welche die Daten der Schnecke als String zurückgibt.
6. Testen Sie Ihre Klasse, indem Sie probehalber ein Rennschneckenobjekt erzeugen und seine Daten auf der Konsole ausgeben.
   * Tipp: Verwenden Sie zum Ausgeben der Daten die toString() Methode der Rennschnecke.

# Rennen

1. Erstellen Sie eine Klasse "Rennen".
2. Ein Rennen hat folgende Eigenschaften:
   * einen Namen,
   * die Anzahl der teilnehmenden Schnecken,
   * die teilnehmenden Schnecken selbst, als Array: einfachere Umsetzung mit konstanter Teilnehmerzahl,
   * die Länge der zu kriechenden Strecke.
3. Überlegen Sie sich, welche dieser Werte bereits im Konstruktor gesetzt werden sollten.
4. Legen Sie in der Klasse "Rennen" eine Methode "void addRennschnecke(Rennschnecke neueSchnecke)" an, welche dem Rennen eine Schnecke hinzufügt.
5. Legen Sie in der Klasse "Rennen" eine Methode "void removeRennschnecke(String name)" an, welche eine Schnecke aus dem Rennen entfernt.
6. Legen Sie in der Klasse "Rennen" eine Methode "public String toString()" an, welche die Daten des Rennens als String zurückgibt.
   * Tipp: Um die Daten der beteiligten Schnecken zurückzugeben, können Sie deren toString() Funktion benutzen.
7. Testen Sie Ihre Klasse vom Hauptprogramm aus!
8. Legen Sie in der Klasse "Rennen" eine Methode "Rennschnecke ermittleGewinner()" an, welche null zurückliefert, wenn noch keine der teilnehmenden Schnecken das Ziel erreicht hat und anderenfalls die Gewinnerschnecke zurückgibt.
9. Legen Sie in der Klasse "Rennen" eine Methode "void lasseSchneckenKriechen()" an, welche alle teilnehmenden Schnecken einmal kriechen lässt.
10. Legen Sie in der Klasse "Rennen" eine Methode "void durchfuehren()" an, welche so lange lasseSchneckenKriechen() aufruft, bis eine der Schnecken das Ziel erreicht hat.
    * Tipp: Ob eine Schnecke im Ziel angekommen ist, können Sie mit der Methode ermittleGewinner() herausfinden.

# Wettbüro

1. Erstellen Sie eine Klasse Wettbuero.
2. Ein Wettbuero hat die folgenden Eigenschaften:
   * Es weiß, für welches Rennen es seine Wetten entgegennimmt.
   * Es verfügt über eine Liste (z.B. ein Array) von angenommenen Wetten.
   * Es hat einen festen Faktor, mit welchem Wetteinsätze bei einem Gewinn multipliziert werden.
3. Legen Sie in der Klasse "Wettbuero" eine Methode "wetteAnnehmen(String schneckenName, int wettEinsatz, String spieler)" an, welche eine Wette entgegennimmt. Die Wette ist bezogen auf eine Schnecke für das Rennen, das von dem Büro bearbeitet wird.
   * Um die einzelnen Wetten speichern zu können, sollten ihre Daten in eigenen Objekten der Klasse "Wette" gespeichert werden. Erstellen Sie diese Klasse selbst.
   * Tipp: Denken Sie an toString().
4. Legen Sie in der Klasse "Wettbuero" eine Methode "rennenDurchfuehren()" an, welche das betreute Rennen durchführt.
5. Legen Sie in der Klasse "Wettbuero" eine Methode "toString()" an, welche die Daten des Wettbueros, die Daten des Rennens sowie sämtliche abgeschlossene Wetten als String zurückgibt.
6. Testen Sie Ihr Programm!
7. Tauschen Sie die Klasse Rennen mit einem Teilnehmer, der ebenfalls bereits fertig ist.
   * Sollten ihre Programme immer noch laufen?
   * Treten Fehler auf? Wenn ja: ändern Sie die Programme so ab, dass Sie problemlos Klassen austauschen können.

# Zusatz

Es gibt noch ein paar Dinge, die in unserem Wettbüro nicht so laufen wie sie sollten. Machen Sie sie besser.

* Dieselbe Schnecke kann in ein Rennen zweimal eingetragen werden.
* Es können negative Wetten abgeschlossen werden.
* Gehen zwei Schnecken gleichzeitig durchs Ziel, wird die Schnecke ausgegeben, auf die die Suche nach dem Gewinner zuerst stößt. Das ist ganz schön ungerecht.