

### Aufgabe 6.4: Pferderennen (Scheinaufgabe)

In dieser Aufgabe wollen wir ein Pferderennen simulieren. Hierzu benötigen wir Pferde (als Objekte), die in unterschiedlichen Ausprägungen existieren (normale Pferde, Rennpferde und Champions), welche sich jeweils in Ihrer Geschwindigkeit unterscheiden. Rennpferde und Champions sollen hierbei durch Vererbung die Elternklasse erweitern. Zusätzlich werden wir in einer weiteren Klasse ein Rennen von fünf Pferden durchführen.

#### Klasse Horse (Pferd):

- Ein Pferd hat die Attribute Name, eine Nummer sowie eine Maximalgeschwindigkeit. Der Name soll beim Erzeugen eines `Horse`-Objekts mit angegeben werden und nachträglich nicht mehr veränderbar sein.
- Die Nummer eines Pferdes soll *automatisch* vergeben und immer um eins hochgezählt werden, d.h., das erste erstellte `Horse`-Objekt bekommt die Nummer 1, das zweite die Nummer 2, usw. Die Nummer soll ebenso nach Erstellung nicht mehr veränderbar sein.
- Die Maximalgeschwindigkeit beträgt für Pferde initial 40,0 km/h.
- Schreiben Sie entsprechend der Vorgaben oben entsprechende Getter- und Setter-Methoden und wählen Sie die Zugriffsmodifizierer passend. Überprüfen Sie beim setzen der Maximalgeschwindigkeit, dass diese immer zwischen 1 und 40,0 km/h liegt!
- Implementieren Sie eine `toString`-Methode, die Informationen über das Pferd zusammenstellt und als String zurück gibt, z.B.  
„Pferd #1 Max, Maximalgeschwindigkeit 40.0“
- Implementieren Sie nun eine Methode `double gallop()`, die eine Wegstecke für einen 1 minütigen Galopp in Metern berechnet und zurück gibt. Die Geschwindigkeit für den Galopp soll zufällig bestimmt werden und zwischen 1 km/h und der im Attribut gespeicherten Maximalgeschwindigkeit liegen.

---

#### Klasse RacingHorse (Rennpferd):

- Ein Rennpferd erweitert nun die Klasse `Horse`. Beachten Sie bitte, dass Konstruktoren nicht vererbt werden.
- Ein Rennpferd hat nun eine initiale Maximalgeschwindigkeit von 60 km/h.
- Überschreiben Sie die Methode `toString()`, so dass der String mit „Rennpferd“ anstatt „Pferd“ beginnt. Überschreiben Sie auch die Methode zum Setzen der Maximalgeschwindigkeit mit einer passenden Implementierung!

### Klasse Champion:

- Ein `Champion` erweitert nun die Klasse `RacingHorse`. Beachten Sie bitte, dass Konstruktoren nicht vererbt werden.
- Ein `Champion` hat nun eine initiale Maximalgeschwindigkeit von 75 km/h.
- Champions werden zusätzlich mit einem Sterne-Ranking versehen (zwischen einem und fünf Sternen). Erweitern Sie diese Kindklasse so, dass die Anzahl der Sterne als zusätzliches `int`-Attribut mit gespeichert wird. Die Anzahl der Sterne soll im Konstruktor gesetzt werden und über eine Getter-Methode auslesbar sein. Achten Sie bitte auch darauf, dass keine Sterne außerhalb des Intervalls [1,5] gespeichert werden können.
- Überschreiben Sie die Methoden `toString()` und das Setzen der Maximalgeschwindigkeit mit einer passenden Implementierung! Die `toString()`-Methode soll auch die Sterne „grafisch“ mit ausgeben, z.B. so:  
`Champion (***) #2 Heinz, Maximalgeschwindigkeit 75.0`

### Klasse Race (Rennen):

In der Klasse `Race` soll nun ein Rennen in der `main()`-Methode durchgeführt werden.

- Erstellen Sie sich hierzu ein Array der Länge 5 vom Datentyp `Horse`. Dank Polymorphie können wir hier sowohl `Horse`, `RacingHorse` als auch `Champions` als Elemente speichern.
- Speichern Sie im Array drei Pferde, ein Rennpferd und einen 5-Sterne Champion
- Ein Rennen geht über 10 Minuten, d.h. die Methode `gallop()` wird für jedes Pferd 10x aufgerufen.
- Am Ende des Rennens geben Sie jedes Pferd und die zurückgelegte Strecke aus. Zusätzlich bestimmen Sie auch noch den Gewinner und geben dessen Namen aus. Ihre Ausgabe könnte z.B. so aussehen:

---

```
Pferd #1 Max, Maximalgeschwindigkeit 40.0: 4497 Meter.  
Pferd #2 Anton, Maximalgeschwindigkeit 40.0: 3829 Meter.  
Pferd #3 Claudia, Maximalgeschwindigkeit 40.0: 2360 Meter.  
Rennpferd #4 Wirbelwind, Maximalgeschwindigkeit 60.0: 5520 Meter.  
Champion (*****) #5 Eclipse, Maximalgeschwindigkeit 75.0: 5827 Meter.  
Gewinner: Eclipse
```

Zur Abgabe wählen Sie Ihre **4 Quellcodedateien** im Project Explorer aus und exportieren Sie diese über *File* → *Export...* → *General / Archive File* in ein ZIP-Archiv

**Abgabe des ZIP-Archivs (Scheinaufgabe) über Moodle.**

**Bitte schreiben Sie als Kommentar im Quelltext Ihren Namen und Ihre Matrikelnummer.**