

## Übungsaufgabe 1 - Raumschiffe

Dienstag, 6. Oktober 2015 13:30

TECHNICAL  
UNIVERSITY  
OF  
THURINGIA

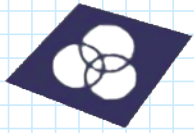
- 1) Erstellen Sie eine Klasse "Raumschiff" mit den Instanzvariablen:
  - Länge
  - Breite
  - Höhe
- 2) Erzeugen Sie zwei Objekte:
  - Discovery 23,24m x 23,79m x 3,71m (LxBxH)
  - Enterprise 288,6m x 127,1m x 72,6m (LxBxH)
- 3) Führen sie anschließend eine Wertezuweisung sowie eine Referenzzuweisung durch

## Übungsaufgabe 2 - Datenkapselung

Dienstag, 6. Oktober 2015 13:34

T E C C  
E A H  
C M A  
H P M  
N U  
O S  
I  
O  
G  
I  
U

- 1) Erstellen Sie eine Klasse "Raumschiff" mit den Instanzvariablen
  - Länge
  - Breite
  - Höhe
- 2) Erzeugen Sie nach dem Prinzip der Kapselung zwei Objekte
  - Enterprise 288,6m x 127,1m x 72,6m (LxBxH)
  - Voyager 344,5m x 132,1m x 64,4m (LxBxH)
- 3) Geben Sie die Maße der beiden Raumschiffe über die Konsole aus (ebenfalls nach Prinzip der Kapselung )

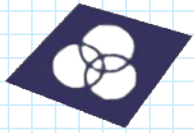


## Übungsaufgabe 3 - Auto

Dienstag, 6. Oktober 2015 13:35

T E C  
C A C  
H M H  
N P A  
O U M  
I S  
O  
G  
I  
U

- 1) Erstellen Sie eine Klasse "Golf" mit den private Instanzvariablen (Kapselung)
  - Länge (double), Breite (double), Höhe (double)
  - Motortyp (string)
  - Motorleistung (double)
- 2) Erzeugen Sie nach dem Prinzip der Kapselung zwei Objekte
  - Cabrio und Limousine
- 3) Weisen Sie bei der Instanziierung den beiden Objekten ihre Abmessungen zu:
  - Cabrio: 4.246 mm x 1.782 mm x 1.423 mm (LxBxH)
  - Limousine: 4.225 mm x 1.799 mm x 1.452 mm (LxBxH)
- 4) Nutzen Sie anschließend die "getter" und "setter" Methoden um den beiden Objekten einen Motortyp und eine Motorleistung zuzuweisen (Kapselung)
- 5) Sehen Sie einen Destruktor vor der bei Aufruf eine entsprechende Meldung absetzt (zum Test ist ein Erzwingen der Garbage Collection nötig)

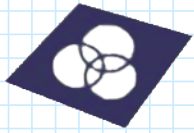


## Übungsaufgabe 4 - Vererbung

Dienstag, 6. Oktober 2015 13:35

T E C  
C A H  
H M A  
N P M  
O U  
I S  
O  
G  
I  
U

- 1) Erstellen Sie eine Klasse "Auto" mit den Instanzvariablen
  - Länge (double), Breite (double), Höhe (double)
- 2) Erstellen Sie zwei abgeleitete Klassen:
  - Golf mit der Instanzvariable Höchstgeschwindigkeit (double)
  - Passat mit der Instanzvariable Laderaum (double)
- 3) Erzeugen Sie zwei Objekte und weisen Sie bei der Instanziierung den beiden Objekten ihre Abmessungen zu:
  - Mein\_Golf: 4.225 m x 1.799 m x 1.452 m (LxBxH)
  - Mein\_Passat: 4.770 m x 1.820 m x 1.520 m (LxBxH)
- 1) Weisen Sie dem Modell (Objekt) Mein\_Golf ein Höchstgeschwindigkeit . von 235 km/h zu.
- 5) Weisen Sie dem Modell (Objekt) Mein\_Passat eine Ladefläche von 603 Liter zu.
- 6) Geben Sie die Daten der Modell über die Konsole aus.
- 7) Achten Sie darauf, dass die Objektintegrität (Kapselung) eingehalten wird.



## Übungsaufgabe 5 - Krieg\_der\_Sterne

Dienstag, 6. Oktober 2015 13:35

1) Erstellen Sie eine Basisklasse Flotte mit den Klassenvariablen:

- aktuelle Flottenzahl (int) (Initialwert 0)
- noch verfügbare Besatzung (int) (Initialwert 500.000)
- gefallene Besatzung (int) (Initialwert 0)
- zerstörte gegnerische Schiffe (int) (Initialwert 0)

2) Erstellen Sie drei Kindklassen XWing, Sternenzerstörer und Todesstern

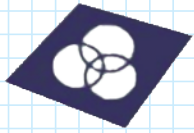
3) Generieren Sie 4 Objekte der Klasse XWing, 3 Objekte der Klasse Sternenzerstörer und ein Objekt der Klasse Todesstern.

4) Beachten Sie das ein XWing zwei Besatzungsmitglieder, ein Sternenzerstörer 37.085 Mitglieder und ein Todesstern 265.657 Besatzungsmitglieder benötigt.

5) Gefechtsverlauf: Todesstern Treffer, XWing zerstört, Todesstern Treffer, XWing Treffer, Sternenzerstörer Treffer, XWing Treffer, Sternenzerstörer zerstört, XWing Treffer

6) Geben Sie über die Konsole den abschließenden Gefechtsverlauf aus.

7) Arbeiten Sie mit Konstruktoren, Destruktoren, einer Funktion Treffer()



## Übungsaufgabe 6 - Timer

Mittwoch, 9. November 2016 18:45

T C C  
E A H  
C M A  
H P M  
N U  
O S  
I  
O  
G  
I  
U

- 1) Erstellen Sie eine Klasse Timer welche die Methode Timer\_Start enthält.
- 2) Dieser Methode soll ein Wert (Datentyp int) übergeben werden, welcher der gesetzten Zeit (in Sekunden) entspricht. In der Methode soll nach Ablauf der Zeit das Event TimerActivate aufgerufen werden.  
TIPP: Nutzen Sie eine Schleife und die System.Threading.Thread.Sleep Methode.
- 3) In Main muss ein Objekt der Klasse Timer angelegt werden und der Timer entsprechend einer Benutzereingabe (Konsoleneingabe) gestartet werden.
- 4) Wird das Event ausgelöst soll eine Konsolenausgabe erfolgen, welche auf den abgelaufenen Timer hinweist. Die Ausgabe sollte auch beinhalten, welche Zeit vom Benutzer vorgegeben wurde.  
TIPP: Lässt sich durch Ereignisse mit Übergabeparameter realisieren.