

Aufgabe 1 (Klassen)

1. Entwerfen Sie ein Objekt "Student" und initialisieren Sie dessen Attribute (bzw. Datenfelder) direkt mit sinnvollen Werten.
2. Definieren Sie eine Klasse "**CustomArray**" mit folgenden Eigenschaften:
 - array [*attribute*]: enthält ein Standard-JavaScript-Array
 - print [*method*]: gibt den Inhalt von 'array' wie folgt aus: [*elem₀*, *elem₁*, *elem₂*, ... , *elem_n*]
 - sort [*method*]: sortiert mit einem beliebigen Sortieralgorithmus aufsteigend den Inhalt von 'array'

Alle Attribute müssen durch den Konstruktor erzeugt und sinnvoll initialisiert werden.

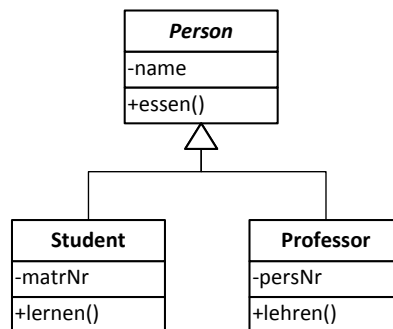
Testen Sie Ihre Implementierung durch folgende Arbeitsschritte:

- Erzeugen Sie von "**CustomArray**" eine Instanz namens "*randomNumbers*", in der 'array' zweihundert zufällig erzeugte, natürliche Zahlen enthält.
 - Zeigen Sie anschließend im Browser den Inhalt von "*randomNumbers*" an.
 - Sortieren Sie "*randomNumbers*" in aufsteigender Reihenfolge und zeigen Sie den sortierten Inhalt wieder an.
3. Müssen/sollten alle Methoden einer JavaScript-Klasse in der Klasse selbst definiert werden? Nennen Sie ein Beispiel bzw. Gegenbeispiel!
 4. Was ist die Bedeutung von „prototype“?

Aufgabe 2 (Vererbung)

1. An einer Hochschule gibt es u.a. Professoren und Studenten. Überarbeiten Sie Ihren Entwurf aus Aufgabe 1.1 entsprechend dem nachfolgend abgebildeten Klassendiagramm (nutzen Sie dazu Vererbung).

Testen Sie Ihre Vererbungshierarchie, indem Sie eine Instanz von „Professor“ und mind. drei Instanzen von „Student“ erzeugen. Implementieren Sie dazu eine Testausgabe im Konstruktor, die relevante Informationen ausgibt. Rufen Sie dann testweise von beiden die Methoden *essen()* und *lernen()* bzw. *lehren()* auf – eine Testausgabe genügt auch hier wieder.



2. Ergänzen Sie die Klasse „Person“ dergestalt, dass mitgezählt wird, wie viele Instanzen davon erzeugt werden. Geben Sie diese Information im Konstruktor ebenfalls mit aus.