

Aufgabe 20: Array von Objektreferenzen

die nachfolgende Angabe sollte Euch bekannt vorkommen (aus Aufgabe 7 bzw. 12). Die damals erstellte Lösung kann für diese Aufgabe verwendet werden. D.h. es sind die existierenden Klassen so anzupassen, dass die **Wagons** nicht einzelnen Attributen, sondern **Feldern eines Arrays zugeordnet** werden.

Die Klasse Personenwagon bleibt unverändert (eventuelle zusätzliche Methoden ausgenommen):

Klasse *Personenwagon*:

Personenwagon
- maximum : int - belegt : int - klasse : int
+ Personenwagon(maximum : int, klasse : int) + einsteigen(anzahl : int) : int + aussteigen(anzahl : int) : int + getMaximum() : int + getBelegt() : int + getKlasse() : int

Details: siehe Aufgabe 7

Bei der *Personenzug*-Klasse sollen die Attribute durch ein entsprechendes Array ersetzt werden:

Klasse *Personenzug*:

Personenzug
- wagons: Personenwagon [] // Array von Wagons - freiePos : int
+ Personenzug(wagon : Personenwagon) + wagonHinzufuegen(wagon : Personenwagon) : boolean + getAnzahlFreieSitzplaetze(klasse : int) : int + verteilen(anzahl : int, klasse : int) : int

Vorgaben:

Ein Zug kann aus bis zu 10 Wagons bestehen, die über ein Array angesprochen werden. Eine Lokomotive wird hier nicht eigens behandelt. Das Array soll **lückenlos befüllt sein** und der Index des ersten freien Feldes im Attribut **freiePos** gespeichert sein.

Achtung: In den Theoriestunden haben wir eine Lösung ohne diesem Attribut realisiert!

Konstruktor *Personenzug(...)*:

Es kann ein gültiges Objekt vom Typ Personenwagon übergeben werden, das dann als erster Wagon hinzugefügt wird. Sollte eine Nullreferenz (*null*) übergeben werden, dann ist ein „Default-Wagon“ zu erzeugen (... *new Personenwagon(...)*) und einzufügen.

wagonHinzufuegen(...):

Der als Parameter übergebene Wagon wird hinzugefügt, sofern noch ein Feld im Array frei ist.

Rückgabewert: *true* bei erfolgreichem Hinzufügen, andernfalls *false* (auch bei Übergabe von *null*)

getAnzahlFreieSitzplaetze(...):

Rückgabewert: wenn Parameter *klasse* = 0 ... die Anzahl aller freien Sitzplätze in allen Wagons,

wenn Parameter *klasse* = 1 ... die Anzahl aller freien Sitzplätze in 1. Klasse Wagons

wenn Parameter *klasse* = 2 ... die Anzahl aller freien Sitzplätze in 2. Klasse Wagons

verteilen(...):

Diese Methode geht Wagon für Wagon durch (beginnend beim Ersten) und befüllt sozusagen die Sitzplätze. Wenn nicht alle Personen im 1. Wagon Platz finden, dann geht es weiter zum 2., ...

Wenn nicht alle Personen einen Platz gefunden haben, dann fahren sie nicht mit. D.h. der

Rückgabewert liefert die Anzahl der Personen, die NICHT Platz hatten.

Achtung: Es gibt Wagons 1. und 2. Klasse. Die Personen dürfen nur jenen Wagons zugeteilt werden, deren Klasse mit der hier Übergebenen (Parameter *klasse*) übereinstimmt!

Bitte unbedingt beachten:

- Namenskonventionen (Groß- und Kleinschreibung)
- Vorgegebenen Methoden- und Attributnamen beibehalten
- Fehlerfälle (ungültige Werte bei Parametern oder null-Referenzen) wirklich testen
→ es empfiehlt sich, eine **Testklasse** mit entsprechenden Testmethoden zu erstellen!