

Arbeitsauftrag – Mitarbeiter (Gen./Spez. mitabstrakter Oberklasse)

Die Gehaltsabrechnung einer Firma steht an. Ein Arbeiter bekommt seine monatlich geleisteten Stunden mit dem entsprechenden Stundenlohn bezahlt. Die Angestellten erhalten ein Grundgehalt plus anteilige Prämie. Der Manager verdient 1 Million Euro.

- a)  Erstelle eine abstrakte Oberklasse `MITARBEITER` mit den Attributen `Name` und `Festanstellung`. Diese Attribute sollen jeweils geschützt, also `protected` sein (bei `abstract` aber eigentlich nicht nötig). Außerdem soll die Klasse eine abstrakte Methode `gehaltRechnen()` besitzen, die eine Eurobetrag zurückgibt. Vergleiche mit der Lösung.
- b)  Erstelle nun drei Unterklassen mit den Namen `ARBEITER`, `ANGESTELLT` und `MANAGER`. Bei `ARBEITER` benötigt man zusätzlich die Attribute `stundenlohn` und `stunden`, bei `ANGESTELLT` `grundgehalt` und `prämiquote`. Vergleiche mit der Lösung.
- c)  Erstelle für jede Unterklasse die Konstruktormethode. Welche Attributwerte müssen jeweils eingegeben werden? Weise die eingegebenen Werten den Variablen zu. (Was muss ich bei einem neuen Arbeiter wissen? Name, Stundenzahl und Stundenlohn). Vergleiche mit der Lösung.
- d)  Erstelle in den Unterklassen die jeweiligen `gehaltBerechnen()`-Methoden.
- e)  Erstelle eine Klasse `ABRECHNUNG` die in einem Feld `mListe` 100 Mitarbeiter verwaltet. Dazu brauchst du also ein Feld `mListe` vom Typ `MITARBEITER` und einen Zähler (`anzahl`) der mitzählt, wie viele Mitarbeiter bereits gespeichert sind.
- f)  Nun benötigst du eine Methode `mitarbeiterDazu` um neue Mitarbeiter (an die entsprechende Stelle im Feld) einfügen zu können. Dieser Methode werden Objekte der Klasse `MITARBEITER` übergeben. Denk daran, die Anzahl der gespeicherten Mitarbeiter entsprechend zu erhöhen.
- g)  Eine Methode `gesamtverdienst()` durchläuft nun alle Mitarbeiter, berechnet jeweils deren Gehalt, summiert diese Einzelgehälter in einer Variable auf und gibt am Ende eine Gehaltssumme aus.
- h)  In einer Klasse `TEST` erstellst Du nun noch eine `testmethode`, die zuerst ein Objekt der Klasse `Abrechnung` und je ein Objekt der drei Unterklassen erstellt, dann die Objekte der Liste hinzufügt und anschließend die Methode `gesamtverdienst` aufruft.
- i)  Ergänze die Oberklasse um eine Methode `namenGeben()`, die den Namen des jeweiligen Mitarbeiters zurück gibt. Optimierte anschließend die Methode `gesamtverdienst()`, sodass zuerst die Namen aller Mitarbeiter mit jeweiligem Gehalt ausgegeben werden und erst anschließend wie schon bisher die Gehaltssumme.