

Klassen

Einstieg

1. Erstellen Sie eine Klasse "Person", die die Eigenschaften Name, Alter und Wohnort hat. Schreiben Sie eine Methode, um die Informationen einer Person auszugeben.
2. Erstellen Sie eine Klasse "Rechteck", die Länge und Breite als Eigenschaften hat. Schreiben Sie Methoden, um den Umfang und die Fläche des Rechtecks zu berechnen.
3. Erstellen Sie eine Klasse "Schüler", die Name, Alter und Klassenstufe als Eigenschaften hat. Schreiben Sie eine Methode, um die Informationen des Schülers auszugeben.
4. Erstellen Sie eine Klasse "Bankkonto", die die Eigenschaften Kontonummer und Kontostand hat. Schreiben Sie Methoden, um Geld abzuheben und einzuzahlen.
5. Erstellen Sie eine Klasse "Auto", die Marke, Modell und Farbe als Eigenschaften hat. Schreiben Sie eine Methode, um die Informationen des Autos auszugeben.
6. Erstellen Sie eine Klasse "Würfel", die die Anzahl der Seiten als Eigenschaft hat. Schreiben Sie eine Methode, um eine zufällige Zahl zwischen 1 und der Anzahl der Seiten zu würfeln.
7. Erstellen Sie eine Klasse "Stundenlohn", die den Stundenlohn und die Anzahl der gearbeiteten Stunden als Eigenschaften hat. Schreiben Sie eine Methode, um den Lohn zu berechnen.
8. Erstellen Sie eine Klasse "Buch", die Titel, Autor und Erscheinungsjahr als Eigenschaften hat. Schreiben Sie eine Methode, um die Informationen des Buches auszugeben.
9. Erstellen Sie eine Klasse "Kreis", die den Radius als Eigenschaft hat. Schreiben Sie Methoden, um den Umfang und die Fläche des Kreises zu berechnen.
10. Erstellen Sie eine Klasse "Konto", die die Eigenschaften Kontoinhaber, Kontonummer und Kontostand hat. Schreiben Sie Methoden, um Geld abzuheben, einzuzahlen und den Kontostand abzurufen.
11. Erstellen Sie eine Klasse "Hund", die Name, Alter und Rasse als Eigenschaften hat. Schreiben Sie eine Methode, um den Hund bellen zu lassen.
12. Erstellen Sie eine Klasse "Student", die Name, Alter und Notendurchschnitt als Eigenschaften hat. Schreiben Sie eine Methode, um den Status des Schülers basierend auf seinem Notendurchschnitt zurückzugeben (z.B. "Erfolgreich", "Durchschnittlich", "Mangelhaft").
13. Erstellen Sie eine Klasse "Bank", die ein Array von Bankkonten verwaltet. Schreiben Sie Methoden, um ein neues Konto hinzuzufügen, den Gesamtstand aller Konten zu berechnen und das Konto mit dem höchsten Kontostand zurückzugeben.
14. Erstellen Sie eine Klasse "Zähler", die einen Zählerwert als Eigenschaft hat. Schreiben Sie Methoden, um den Zähler zu inkrementieren, zu dekrementieren und den aktuellen Wert zurückzugeben.
15. Erstellen Sie eine Klasse "Fahrzeug", die Marke, Modell und Baujahr als Eigenschaften hat. Schreiben Sie eine Methode, um das Fahrzeug zu starten.
16. Erstellen Sie eine Klasse "Rechteck", die Länge und Breite als Eigenschaften hat. Schreiben Sie Methoden, um den Umfang, die Fläche und zu überprüfen, ob es sich um ein Quadrat handelt.
17. Erstellen Sie eine Klasse "Telefonbuch", die ein Array von Kontakten verwaltet. Schreiben Sie Methoden, um einen neuen Kontakt hinzuzufügen, einen Kontakt zu suchen und alle Kontakte auszugeben.
18. Erstellen Sie eine Klasse "Stundenplan", die ein Array von Kursen verwaltet. Jeder Kurs hat einen Namen und eine Uhrzeit. Schreiben Sie Methoden, um einen neuen Kurs hinzuzufügen, einen Kurs zu suchen und den gesamten Stundenplan auszugeben.

19. Erstellen Sie eine Klasse "Person", die Name, Alter und Geschlecht als Eigenschaften hat. Schreiben Sie eine Methode, um das Alter der Person zu aktualisieren.
20. Erstellen Sie eine Klasse "Wetter", die Temperatur und Wetterbedingungen als Eigenschaften hat. Schreiben Sie Methoden, um die Temperatur in Fahrenheit umzurechnen und das aktuelle Wetter anzuzeigen.

Bankkonto

Erstellen Sie eine Klasse "Bankkonto", die die Eigenschaften Kontonummer, Kontostand und Inhaber hat. Schreiben Sie Methoden, um Geld abzuheben und einzuzahlen. Implementieren Sie außerdem eine Überziehungsmöglichkeit für das Konto, bei der ein bestimmter Betrag über den Kontostand hinaus abgehoben werden kann. Für Überziehungen werden für jede Abbuchung 2€ Gebühren verrechnet.

Zugverspätungen

Erstellen Sie eine Klasse "Zug", die Informationen über einen Zug wie Zugnummer, Abfahrtsbahnhof, Ankunftsbahnhof und Abfahrtszeit enthält. Implementieren Sie eine Methode, die prüft, ob ein Zug verspätet ist, basierend auf der aktuellen Uhrzeit und der Abfahrtszeit des Zuges. ('LocalDateTime')

Hunde

Erstellen Sie eine Applikation, die eine CSV-Datei (<https://www.wien.gv.at/finanzen/ogd/hunde-wien.csv>) einliest und deren Datensätze verarbeitet. Bauen Sie die Applikation wie folgt auf:

1. Erstellen Sie eine Klasse (ein Objekt), die einen Zählpunkt repräsentiert (Postleitzahl, Hundart, Datum, Anzahl,...)
2. Die Methode **readData()** soll die CSV-Datei einlesen und ein Array von Hunden zurückliefern.
3. Die Methode **printDogtypes()** gibt alle Hunde-Rassen formatiert auf die Konsole aus
4. Die Methode **printAll()** gibt alle Objekte formatiert auf die Konsole aus.
5. Die Methode **export(String text)** exportiert alle Hunde, deren Hunde-Typ den Text beinhalten. (zB. **export("Pudel")** gibt alle Mischungen von Pudel aus: **"Pudel / Dackel"**, **"Pudel / Golden Retriever"**, ...)
6. **readData()** wird bei Programmstart aufgerufen.
7. Erstellen Sie ein Menü, das in einer Schleife immer die nächste Aktion vom Benutzer einliest und die dementsprechende Methode aufruft.
 - printDogtypes
 - printAll
 - export
 - Hier wird zuerst der Dateiname vom Benutzer abgefragt, danach der Text.
 - Wenn der Pfad nicht existiert, soll das Verzeichnis erzeugt werden.
 - Die Datei soll alle relevanten Daten mit ; getrennt beinhalten. Erstellen Sie auch einen Header.
 - Nach vollzogenem Export, soll eine Meldung ausgegeben werden, wie viele Datensätze exportiert wurden.