1. Innerhalb eines eigene Namensraum soll eine Klasse "Fahrzeug" erstellt werden. Die Klasse hat folgende private Attribute:

* Zeichenkette "marke"
* Zeichenkette "modell"
* Integer "baujahr"

folgende public Methoden:

* getMarke() => Zeichenkette
* setMarke(Zeichenkette) => nichts
* getModell() => Zeichenkette
* setModell(Zeichenkette) => nichts
* getBaujahr() => Integer
* setBaujahr(Integer) => nichts

und folgende Kontruktoren:

* Fahrzeug(Zeichenkette, Zeichenkette, Integer)
* Fahrzeug(Zeichenkette, Zeichenkette)
* Fahrzeug(Zeichenkette)

Dazu gibt es noch folgende statische Funktion:

* FahrzeugInfo(Fahrzeug als konstante Referenz) => nichts

Die Konstruktoren sollen aufeinander aufbauen und eventuelle Standardwerte vergeben ("unbekannt"/1900). Die Funktion "FahrzeugInfo" soll alle Informationen zu dem Fahrzeug ausgeben.

Es sollen drei Fahrzeuge erstellt werden (immer anderer Konstruktor) und an die Funktion "FahrzeugInfo" übergeben werden.

Ausgabe

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Name Trainer, Datum

**Aufgabe: Studentenverwaltungssystem mit Schleife**

Erstelle eine Klasse namens **Student**, die Informationen über einen Studenten speichern kann. Jeder Student hat einen Namen, eine Matrikelnummer und eine Liste von Kursen, die er belegt.

Die Klasse sollte folgende Funktionen unterstützen:

1. **Hinzufügen eines Kurses:** Eine Methode, die einen Kurs zum Kursverlauf des Studenten hinzufügt.
2. **Entfernen eines Kurses:** Eine Methode, die einen Kurs aus dem Kursverlauf des Studenten entfernt.
3. **Anzeigen des Kursverlaufs:** Eine Methode, die den Kursverlauf des Studenten ausgibt.

Erstelle dann eine separate Klasse namens **Studentenverwaltungssystem**, die eine Sammlung von **Student**-Objekten verwaltet. Die Klasse sollte folgende Funktionen unterstützen:

1. **Hinzufügen eines Studenten:** Eine Methode, die einen neuen Studenten zum System hinzufügt.
2. **Entfernen eines Studenten:** Eine Methode, die einen Studenten aus dem System entfernt.
3. **Anzeigen aller Studenten:** Eine Methode, die alle Studenten im System mit ihren Kursverläufen ausgibt.

Implementiere eine Schleife in der **main**-Funktion, die es dem Benutzer ermöglicht, Studenten hinzuzufügen, Kurse hinzuzufügen oder zu entfernen und alle Studenten im System anzuzeigen. Die Schleife sollte so lange laufen, wie der Benutzer das Programm nicht beendet. Verwende dazu beispielsweise eine einfache Benutzereingabe, um Aktionen auszuwählen.

**Aufgabe: Buch-Klasse**

Erstelle eine C++-Klasse namens "Book" für die Repräsentation von Büchern. Jedes Buch soll die folgenden Eigenschaften haben:

1. **title** (String): Der Titel des Buches.
2. **author** (String): Der Autor des Buches.
3. **publicationYear** (int): Das Veröffentlichungsjahr des Buches.

Die Klasse sollte einen Konstruktor haben, der die oben genannten Attribute initialisiert. Zusätzlich soll die Klasse eine Funktion namens **displayInfo** haben, die die Informationen über das Buch auf der Konsole ausgibt.

1. **isSameAuthor(const Book& other) -> bool**: Diese Methode soll überprüfen, ob das aktuelle Buch vom gleichen Autor wie ein anderes Buch (**other**) stammt. Sie gibt **true** zurück, wenn die Autoren gleich sind, andernfalls **false**.
2. **isPublishedAfter(const Book& other) -> bool**: Diese Methode soll überprüfen, ob das aktuelle Buch nach einem anderen Buch (**other**) veröffentlicht wurde. Sie gibt **true** zurück, wenn das aktuelle Buch später veröffentlicht wurde, andernfalls **false**.
3. **updatePublicationYear(int newYear)**: Diese Methode ermöglicht es, das Veröffentlichungsjahr des Buches zu aktualisieren.
4. **displayInfoWithAge() const**: Diese Methode soll die Informationen über das Buch auf der Konsole ausgeben und zusätzlich das Alter des Buches basierend auf dem aktuellen Jahr anzeigen.

Erstelle dann in der **main**-Funktion zwei Objekte der Klasse "Book", weise ihnen verschiedene Bücher zu und rufe die **displayInfo**-Funktion für jedes Objekt auf, um die Buchinformationen auszugeben.