

## Versuchen Sie immer, Ihren Code zu kommentieren!

Erstellen Sie für jede Aufgabe ein Sequenzdiagramm.

### Aufgabe 1 Benutzerkonto

Der Leiter der Vertriebsabteilung teilt Ihnen telefonisch mit, dass Sie für einen neuen Mitarbeiter ein Benutzerkonto einrichten sollen. Sie legen das Benutzerkonto an. Danach rufen Sie den neuen Mitarbeiter an und bitten ihn, sich zum Test mit seinem neuen Account im Firmennetz anzumelden. Der neue Mitarbeiter meldet sich an. Sie warten die Bestätigung des Mitarbeiters, dass die Anmeldung mit dem neuen Account erfolgreich war, ab.

### Aufgabe 2 Onlineshop

Der Ablauf einer Bestellung in einem Onlineshop soll in einem UML-Sequenzdiagramm dargestellt werden:

- Der Kunde ruft die Startseite mit dem Angebot des Shops auf.
- Der Shop zeigt den Kunden die Startseite mit den zur Wahl stehenden Produktarten (Foto, Poster).
- Der Kunde teilt dem Shop die Produktwahl mit.
- Der Shop erstellt ein Produktvorlagen-Objekt: Durch den Aufruf der Funktion `Create()` der entsprechenden Produktvorlagen-Klasse.
- Der Shop zeigt dem Kunden auf einer HTML-Seite die leere Produktvorlage und die Mitteilung, dass das Bild hochgeladen werden kann, an.
- Der Kunde lädt ein Bild hoch.
- Der Shop übergibt das Bild an eine Produktvorlage: Durch den Aufruf der Funktion `Bild()` des Produktvorlagen-Objektes.
- Die Produktvorlage meldet: "Bild ist eingefügt".
- Der Shop zeigt dem Kunden eine HTML-Seite mit fertiger Produktvorlage an und fordert den Kunden zur Eingabe der Adress- und Zahlungsdaten auf.
- Der Kunde übermittelt die Daten.
- Der Shop ruft die eigene Funktion `Prüfung()` zur Prüfung der Daten auf.
- Die Funktion `Prüfung()` gibt "Daten ok" zurück. Der Fall "Daten nicht ok" wird nicht betrachtet.
- Der Shop teilt dem Kunden mit, dass er zur Annahme des Auftrags bereit ist.

- Der Kunde erteilt den Auftrag.
- Der Shop erstellt ein Auftragsbestätigungs-Objekt mit dem Aufruf der Funktion `Create()` der Klasse `Auftragsbestätigung`.
- Der Shop verschickt eine Auftragsbestätigung per E-Mail an den Kunden.

### Aufgabe 3 Buchungsverwaltung

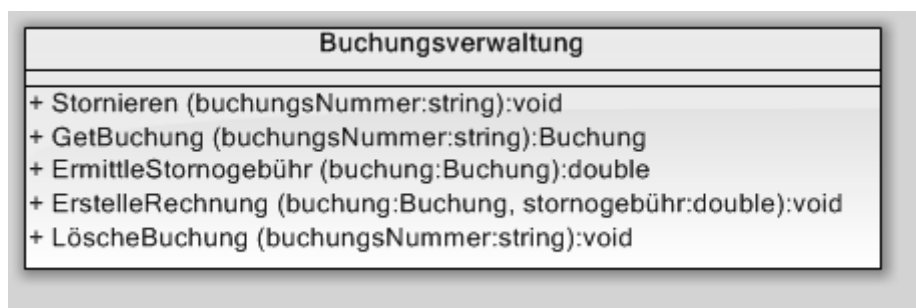
Zur Verwaltung der Gerätebuchungen ist bereits die Klasse „Buchungsverwaltung“ entsprechend dem gegebenen Klassendiagramm (siehe unten) erstellt worden.

Diese ermöglicht unter anderem die Durchführung von Buchungsstornierungen.

Der Stornierungsvorgang wird wie folgt beschrieben:

- Der Vorgang wird durch den Aufruf der Methode `Stornieren()` eingeleitet.
- Überprüfung, ob eine entsprechende Buchung vorliegt (Methode `GetBuchung()`).
- Bei vorliegender Buchung werden die Stornogebühren ermittelt (Methode `ErmittleStornogebühr()`).
- Die Rechnung wird erstellt (Methode `ErstelleRechnung()`).
- Die Buchung wird gelöscht und anschließend wird der Anwender über die Löschung informiert (Methode `LöscheBuchung()`).
- Sollte die Buchung nicht vorhanden sein, wird der Anwender auch darüber informiert.

Klassendiagramm Buchungsverwaltung:



Erstellen Sie für den Ablauf eines Löschvorgangs ein Sequenzdiagramm.

: Buchungsverwaltung

