

Termin: Montag, 20. November 2006

Außerdem, der entsprechende große Berücksichtigungsgestalt Zusammengesetzter, abgeschlossen.
Die Vervielfältigung, Verbreitung und öffentliche Wiedergabe der Prüfungsaufgaben und Lösungen ist nicht gestattet. Zuwiderhandlungen werden zivil- und strafrechtlich (§§ 97 ff., 106 ff. UrhG) verfolgt. – © ZPA Köln 2006 – Alle Rechte vorbehalten!

Die EASY-Travel GmbH ist ein Reiseveranstalter, der seine Geschäftsprozesse durch EDV-Anwendungen besonders effizient gestalten will.

1. Analyse und Modellierung von Geschäftsprozessen
2. Erstellung einer Entscheidungstabelle
3. Erstellung eines UML-Anwendungsfalldiagramms und von SQL-Abfragen
4. Erstellung von Tabellen an Hand eines vorgegebenen ER-Modells
5. Erstellung einer Methode (Programmlogik)
6. Erstellung von Testdaten für einen Algorithmus

Die EASY-Travel GmbH will ihre Geschäftsprozesse nach dem Konzept zur computerunterstützten Modellierung und Dokumentation von Geschäftsprozessen dokumentieren und modellieren.

(4 Punkte)

(6 Punkte)

[illegible]

c) Für die Funktion „Abwicklung eines Buchungsauftrags“ wurden folgende Teil- und Elementarfunktionen ermittelt.

Erstellen Sie einen Funktionshierarchiebaum.

(10 Punkte)

Korrekturrand

Abwicklung eines Buchungsauftrags

- Buchungsbestätigung
- Reservierung
- Flugreservierung
- Buchungsbearbeitung
- Hotelreservierung
- Fakturierung
- Buchungserfassung

Korrekturrand

Die Fahrpreise sind wie folgt festgelegt:

- Erstellen Sie aus folgender Tabelle eine Matrix, aus der die Fahrpreise ablesbar sind. Sie sollte konsolidiert sein.

[illegible]This image shows a full page of blank graph paper. The grid consists of small, equal-sized squares formed by thin black lines. There are no margins, text, or other markings on the page.

3. Handlungsschritt (20 Punkte)

Korrekturrand

a) Die EASY-Travel GmbH bereitet die Einführung von Online-Buchungen durch Kunden vor.

Folgendes Szenario soll betrachtet werden:

- Der Kunde kann sich für eine Reise alle erforderlichen Informationen anzeigen lassen.
- Der Kunde muss bei der Buchung einer Reise eingeben:
 - Reisedaten
 - Persönliche Daten
 - Zahlungsart (bei Bankeinzug ergänzend die erforderlichen Bankdaten, bei Kreditkarte die erforderlichen Daten der Kreditkarte)

Erstellen Sie ein Anwendungsfalldiagramm.

(10 Punkte)

Fortsetzung 3. Handlungsschritt →

Auszug aus dem Flugplan der EASY-Travel-Air, Dezember 2006

[illegible]

Für das neue Online-Buchungssystem wurden die beiden folgenden Tabellen Flug und Sitze erstellt.

Sitze

	Daten
FlugID	ET400
Startflughafen	FRA
Zielflughafen	JFK
Abflugzeit	10:00
Ankunftszeit	12:10

	Daten
<u>FlugID</u>	ET400
<u>Datum</u>	07.12.2006
<u>SitzNr</u>	112
FensterMitteGang	F
Belegt	JA
Bemerkung (dieses Feld kann leer sein)	

Erstellen Sie eine entsprechende SQL-Anweisung

(5 Punkte)

Beispiel für Ausgabe

Datum	FlugID	FensterMitteGang	FreiePlaetze
07.12.2006	ET400	F	33
07.12.2006	ET400	M	61
07.12.2006	ET400	G	20

This image shows a single sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

bb) Der Flug ET406 soll nun auch am 02.12.2006 angeboten werden.

Erstellen Sie eine SQL-Anweisung mit der beispielhaft ein Datensatz der entsprechenden Tabelle hinzugefügt wird. (5 Punkte)

Korrekturrand

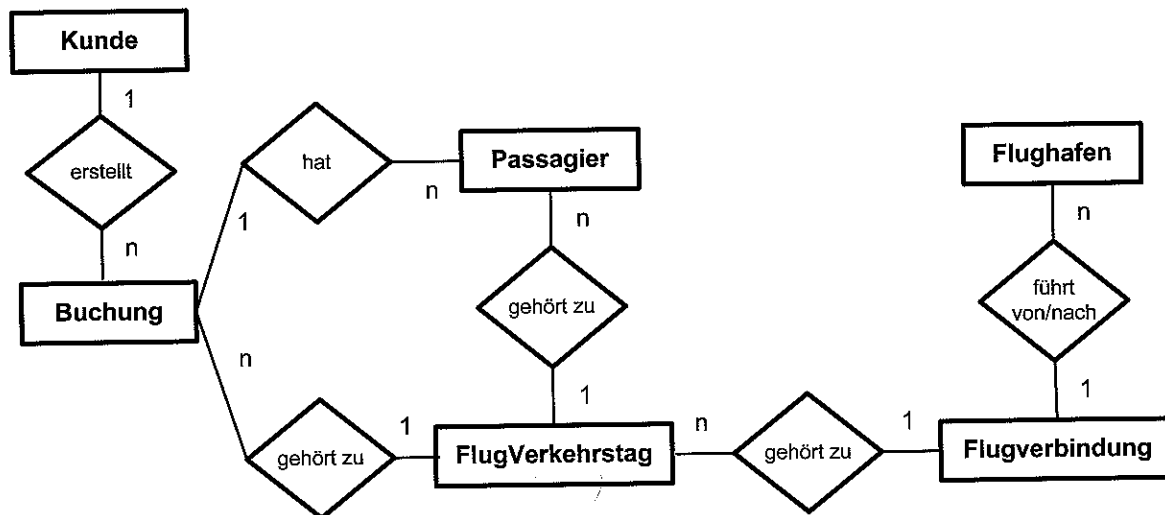
4. Handlungsschritt (20 Punkte)

Korrekturrand

Die Kunden der EASY-Travel GmbH sollen für gebuchte Flüge eine Buchungsbestätigung erhalten. Die dafür erforderlichen Daten sollen in einer Datenbank gespeichert werden.

Erstellen Sie unter Beachtung der nachstehenden Vorgaben (ER-Modell, Buchungsbestätigung, Flugplan) die entsprechenden Tabellen in der dritten Normalform auf der folgenden Seite. Geben Sie alle erforderlichen Attribute an und kennzeichnen Sie Primärschlüssel mit PK und Fremdschlüssel mit FK.

ER-Modell



Buchungsbestätigung

Herr Marcel Pott
Sandweg 2
12345 Berlin

Kundennummer: 00456
Buchungsnummer: 3897212633

Sehr geehrter Herr Pott,

wir bestätigen folgende Buchung.

Hinflug:	ET9250 Köln (CGN) - Palma de Mallorca (PMI)	02.12.06	11:20	13:30
Rückflug:	ET9251 Palma de Mallorca (PMI) - Köln (CGN)	06.12.06	20:20	22:40

Passagiere:

Vorname	Nachname	Geburtsdatum
Petra	Pott	23.07.1968
Inge	Pott	31.10.1988

Preis je Flug und Person: 150,00 €
Gesamtpreis: 600,00 €

Flugplan

Abflug	Ankunft	Verkehrstage							Flug-ID
CGN*	PMI*	Fr 01.12.	Sa 02.12.	So 03.12.	Mo 04.12.	Di 05.12.	Mi 06.12.	Do 07.12.	
11:30	13:30	→	→	→	→	→	→	→	ET9250
13:40	15:40	→	→	→	→	→	→	→	ET9252
17:00	19:00	→	→	→	→	→	→	→	ET9254

*CGN = Flughafen Köln-Bonn, PMI = Flughafen Palma de Mallorca

Fortsetzung 4. Handlungsschritt →

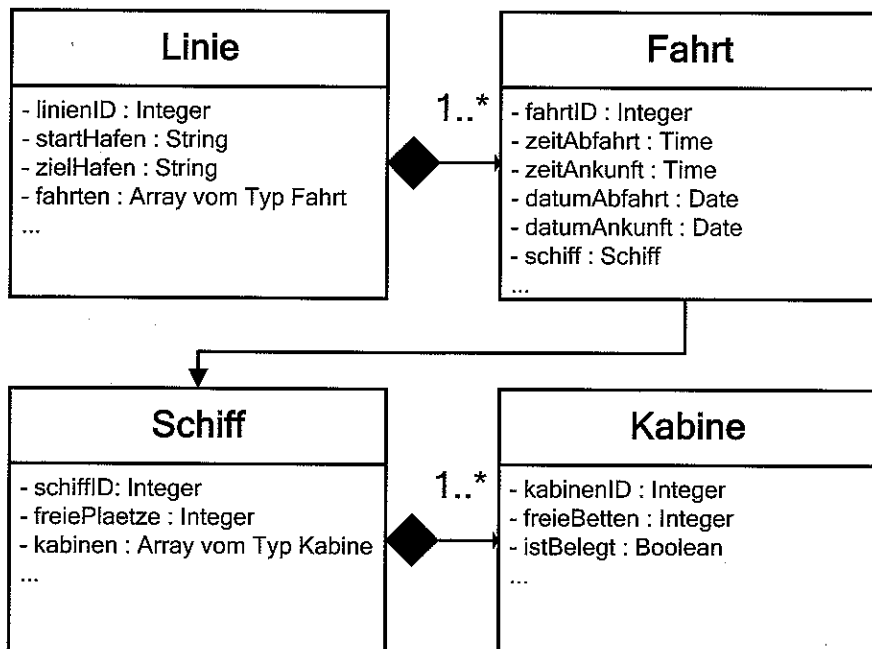
5. Handlungsschritt (20 Punkte)

Korrekturrand

Die EASY-Travel GmbH will Online-Buchungen auch für Fährverbindungen einführen.

Die Software für dieses Buchungssystem soll in einer objektorientierten Programmiersprache realisiert werden. Es sind bereits folgende Klassen erstellt worden:

Klassendiagramm



Erläuterungen:

Eine Kabine kann als belegt gekennzeichnet werden, auch wenn nicht alle Betten belegt sind (z. B. eine Doppelbettkabine wird als Einzelbettkabine genutzt).

Für die Eigenschaften der Klassen gibt es öffentliche get-Methoden, z. B.: getStartHafen() in der Klasse Linie.

Mit einer Methode der Klasse *Linie* soll geprüft werden, ob an einem bestimmten Tag die vom Kunden gewünschte Anzahl Plätze und Betten auf den Schiffen einer Linie verfügbar sind.

Übergabe- und Ausgabedaten der Methode

Übergabedaten:

- Datum der Abfahrt
- Anzahl der reisenden Personen
- Anzahl der gewünschten Kabinenbetten

Ausgabedaten:

- Fahrt-ID
- Abfahrtsdatum
- Ankunftsdatum
- Abfahrtszeit
- Ankunftszeit
- Kabinenbetten verfügbar Ja/Nein
- Plätze verfügbar Ja/Nein

Beispiel einer Ausgabe

ID028 01.11.06 15:00 / 02.11.06 08:30 Plätze nicht verfügbar
ID029 01.11.06 22:00 / 03.11.06 15:30 Plätze verfügbar, Betten nicht verfügbar

Erstellen Sie für die Klasse *Linie* die entsprechende Methode *zeigeVerfügbarkeit()*.

Hinweis: Stellen Sie die Logik in Code (Pseudocode oder an eine Programmiersprache angelehnten Notation) dar.

Lined area for corrections, consisting of 30 horizontal lines.

6. Handlungsschritt (20 Punkte)

Korrekturrand

Die EASY-Travel GmbH legt Wert auf gut getestete Software.

Mit unten stehendem Algorithmus soll die Plausibilität von Datumsangaben für den Monat Februar überprüft werden. Das Datum wird in den Variablen tag, mon und jahr gespeichert.

Mit dem Algorithmus soll ein Whitebox-Test durchgeführt werden. Die Testdaten sollen eine vollständige Anweisungsüberdeckung und Zweigüberdeckung gewährleisten. Einige Testdaten wurden bereits erstellt.

Hinweis: Der Algorithmus soll nur mit Datumsangaben aus dem Monat Februar getestet werden.

a) Testdaten für Anweisungsüberdeckung (Tabelle A)

Die Anweisungen 1 bis 7 müssen mindestens einmal durchlaufen werden.

aa) Geben Sie für jedes vorgegebene Testdatum an, welche der Anweisungen 1 bis 7 durchlaufen werden. Markieren Sie in der Tabelle A die entsprechenden Felder mit X. (4 Punkte)

ab) Ergänzen Sie in der Tabelle A ein weiteres Testdatum, mit dem die Anweisungsüberdeckung erreicht wird und markieren Sie die Nummern der Anweisungen mit X, die mit diesem Datum durchlaufen werden. (4 Punkte)

b) Testdaten für Zweigüberdeckung (Tabelle B)

Jede in der Tabelle B angegebene Bedingung muss mindestens einmal als wahr und mindestens einmal als falsch bewertet werden.

ba) Prüfen Sie je Testdatum, welche der Bedingungen a bis f zutreffen oder nicht zutreffen. Tragen Sie in der Tabelle B true ein, wenn die Bedingung zutrifft und false ein, wenn die Bedingung nicht zutrifft. (6 Punkte)

bb) Ergänzen Sie in der Tabelle B ein weiteres Testdatum, mit dem die Zweigüberdeckung erreicht wird und geben Sie an, ob die angegebenen Bedingungen zutreffen (true) oder nicht zutreffen (false). (6 Punkte)

Algorithmus zur Überprüfung von Datumsangaben aus dem Monat Februar

Anweisungs-Nr	
1	maxtag = 0
2	wenn jahr modulo 4 = 0 und jahr modulo 100 \neq 0 oder jahr modulo 400 = 0 dann
3	maxtag = 29
	sonst
4	maxtag = 28
	ende wenn
5	wenn maxtag > 0 und tag <= maxtag und tag > 0 dann
6	datum_ok = true
	sonst
7	datum_ok = false
	ende wenn

Tabelle A: Testdaten für Anweisungsüberdeckung

Testdatum			durchlaufene Anweisung (Nummern der Anweisungen)						
Tag	Monat	Jahr	1	2	3	4	5	6	7
29	2	2000							
28	2	2001							

Tabelle B: Testdaten für Zweigüberdeckung

Korrekturrand

Bedingung		Testdatum		
		29.2.2004	0.2.2000	
a	jahr modulo 4 = 0			
b	jahr modulo 100 \neq 0			
c	jahr modulo 400 = 0			
d	maxtag > 0			
e	tag <= maxtag			
f	tag > 0			

+ true, – false