Unterlagen für die Lehrkraft

Abiturprüfung 2011

Informatik, Leistungskurs

1. Aufgabenart

Aufgabenart	Datenbankstrukturen – Relationale Datenbanken
Syntaxvariante	–

2. Aufgabenstellung[1]

siehe Prüfungsaufgabe

3. Materialgrundlage

- entfällt

4. Bezüge zu den Vorgaben 2011

1. *Inhaltliche Schwerpunkte*
   Relationale Datenbanken
   - Normalisierung: Überführung einer Datenbank in die 1. bis 3. Normalform
   - Realisierung in einem Datenbanksystem
   - Relationenalgebra (Selektion, Projektion, Join)
   - SQL-Abfragen über eine und mehrere verknüpfte Tabellen

2. *Medien/Materialien*
   - entfällt

5. Zugelassene Hilfsmittel

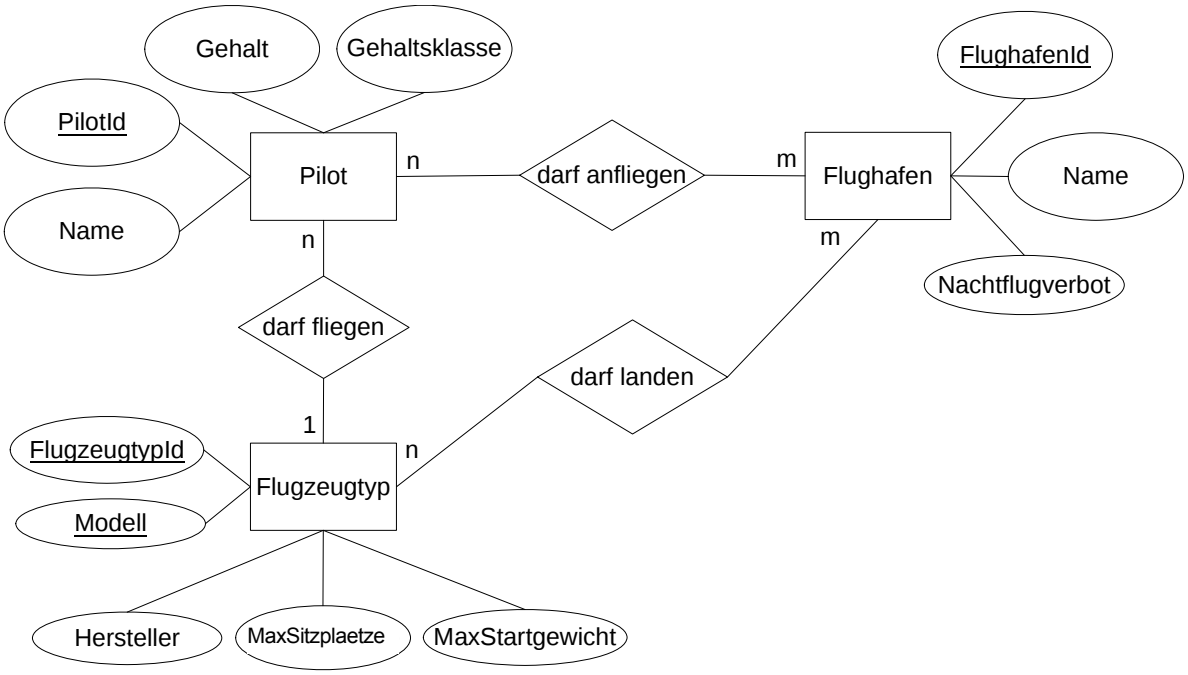
- Wörterbuch zur deutschen Rechtschreibung
- Taschenrechner

[1] Die Aufgabenstellung deckt inhaltlich alle drei Anforderungsbereiche ab.

6. Vorgaben für die Bewertung der Schülerleistungen

6.1 Modelllösungen

Teilaufgabe a)



Da ein Pilot laut Vorgabe nur einen Flugzeugtyp fliegen darf, ein Flugzeugtyp aber durchaus von mehreren Piloten geflogen werden könnte, handelt es sich hier um eine 1:n Beziehung.

Bei den Beziehungen zwischen Pilot und Flughafen bzw. Flugzeugtyp und Flughafen ist jeweils von einer n:m Beziehung auszugehen. Auf einem Flughafen dürfen mehrere Piloten landen, genauso wie ein Pilot mehrere Flughäfen anfliegen darf. Ebenso verhält es sich für den Flugzeugtyp.

Die Beziehung zwischen Pilot und Flugzeugtyp wird dadurch realisiert, dass das Attribut **`FlugzeugtypId`** als Fremdschlüssel in das Relationsschema **`Pilot`** übernommen wird.

`Pilot``(PilotId, Name, Gehalt, Gehaltsklasse, ↑FlugzeugtypId)`

Die beiden anderen werden durch die Einführung zweier weiterer Relationenschemata realisiert, indem die entsprechenden Primärschlüssel als Fremdschlüssel aufgenommen werden.

`darf_anfliegen``(↑PilotId, ↑FlughafenId)`

`darf_landen``(↑FlugzeugtypId, ↑FlughafenId)`

Teilaufgabe b)

1. Normalform
Eine Tabelle liegt in der ersten Normalform (1NF) vor, wenn jeder Attributwert eine atomare, nicht weiter zerlegbare Dateneinheit ist. Eine Tabelle ist nicht in der 1NF, wenn Attribute mehrfach oder komplex in einer Spalte auftreten.

2. Normalform
Eine Tabelle liegt in der zweiten Normalform (2NF) vor, wenn sie in der 1NF ist und jedes Nichtschlüsselattribut voll funktional abhängig vom Primärschlüssel ist. Eine Tabelle ist nicht in der 2NF, wenn Attribute von einem Teil des Schlüssels eindeutig identifiziert werden.

3. Normalform
Eine Tabelle liegt in der dritten Normalform (3NF) vor, wenn sie sich in der 2NF befindet und jedes Nichtschlüsselattribut nicht transitiv abhängig vom Primärschlüssel ist. Eine Tabelle ist nicht in der 3NF, wenn Attribute von anderen Nichtschlüsselattributen identifiziert werden.

Im Schema **`Pilot`** ist das Attribut `Name` nicht atomar, da es sich in Vorname und Nachname aufspalten lässt. Im Schema **`Flughafen`** ist das Attribut `Nachtflugverbot` nicht atomar, da es sich in den Beginn und das Ende des Nachtflugverbotes zerlegen lässt. Somit ist die Bedingung für 1NF verletzt.

Im Schema besteht der Primärschlüssel aus den beiden Attributen `FlugzeugtypId` und `Modell`. Das Nichtschlüsselattribut Hersteller hängt aber bereits vom Attribut `FlugzeugtypId` ab. Somit ist die Bedingung für 2NF verletzt.

Im Schema **`Pilot`** hängt das Attribut `Gehalt` vom Nichtschlüsselattribut `Gehaltsklasse` ab. Somit ist die Bedingung für 3NF verletzt.

Da das Gehalt in Wirklichkeit von weiteren Faktoren abhängig sein kann, ist es auch zu akzeptieren, wenn der Prüfling hier keine Abhängigkeit erkennt.

Um die Relationenschemata in 3NF zu überführen, werden die beiden nichtatomaren Attribute `Name` und `StartLandezeiten` jeweils in zwei atomare Attribute zerlegt.

Es wird eine neue Relation **`Hersteller`** eingeführt, in der das Attribut Flugzeugtyp als Primärschlüssel fungiert. Einziges weiteres Attribut ist `Hersteller`. In der Relation **`Flugzeugtyp`** tritt das Attribut `FlugzeugtypId` nunmehr als Fremdschlüssel auf.

Es wird ein neues Schema **`Gehaltsklasse`** eingeführt, um die transitive Abhängigkeit des Attributs `Gehalt` vom Primärschlüssel PilotId aufzulösen. Der Primärschlüssel `Gehaltsklasse` wird in das Schema **`Pilot`** als Fremdschlüssel eingefügt.

Pilot(<u>PilotId</u>, Vorname, Nachname, ↑GehaltsklasseId,
  ↑FlugzeugtypId)
Hersteller(<u>HerstellerId</u>, ↑FlugzeugtypId)
Flugzeugtyp(<u>FlugzeugtypId</u>, <u>Modell</u>, MaxSitzplaetze, MaxStartgewicht)
Flughafen(<u>FlughafenId</u>, Name, FlugverbotBeginn, FlugverbotEnde)
Gehaltsklasse(<u>GehaltsklasseId</u>, Gehalt)
darf_anfliegen(<u>↑PilotId</u>, <u>↑FlughafenId</u>)
darf_landen(<u>↑FlugzeugtypId</u>, <u>↑FlughafenId</u>)

Teilaufgabe c)

1. Es ist eine Liste aller Flugzeuge zusammen mit dem Flugzeugtyp und dem Baujahr zu erstellen. Dabei sollen nur die Flugzeuge erfasst werden, die vor dem Jahr 2000 gebaut wurden.

```
select FlugzeugId, FlugzeugtypId, Baujahr
from Flugzeug
where Baujahr < 2000
```

2. Es ist die nach dem Namen absteigend sortierte Liste aller Piloten zu erstellen, die überdurchschnittlich viel verdienen.

```
select Name
from Pilot
where Gehalt > (
  select sum(Gehalt)/count(*)
  from Pilot
)
order by Name desc
```

3. Es ist eine Liste aller Flugzeuge zu erstellen, die sowohl in Stuttgart-Echterdingen als auch in Berlin-Schönefeld landen dürfen.

```
select Flugzeug.FlugzeugId
from Flugzeug, (
  select dl1.FlugzeugtypId
  from darf_landen as dl1, darf_landen as dl2
  where dl1.FlugzeugtypId = dl2.FlugzeugtypId
  and dl1.FlughafenId = "STR"
  and dl2.FlughafenId = "SXF") as tmp
where Flugzeug.FlugzeugtypId = tmp.FlugzeugtypId
```

Teilaufgabe d)

```
select Flugzeug.FlugzeugId, Flugzeugtyp.FlugzeugtypId
from Flugzeug, Flugzeugtyp
     where Flugzeug.FlugzeugtypId = Flugzeugtyp.FlugzeugtypId
     and Flugzeug.Sitzplaetze < Flugzeugtyp.MaxSitzplaetze
```

Im Verbund der beiden Tabellen `Flugzeug` und `Flugzeugtyp` werden diejenigen Datensätze ausgewählt, die im Attribut `FlugzeugtypId` übereinstimmen, und bei denen das betrachtete Flugzeug weniger Sitzplätze aufweist, als es für den Flugzeugtyp möglich ist.

Abschließend wird auf die beiden Attribute `FlugzeugId` und `FlugzeugtypId` projiziert. Es werden also alle Flugzeuge zusammen mit ihrer Typbezeichnung selektiert, die weniger Sitzplätze aufweisen, als es für den Typ möglich ist.

```
select Pilot.Name, darf_fliegen.FlugzeugtypId
from Pilot, darf_fliegen, Flugzeugtyp
     where Pilot.PilotId = darf_fliegen.PilotId
     and darf_fliegen.FlugzeugtypId = Flugzeugtyp.FlugzeugtypId
     and MaxSitzplaetze > 400
order by Pilot.Name asc
```

Im Verbund der drei Tabellen `Pilot`, `darf_fliegen` und `Flugzeugtyp` werden diejenigen Datensätze ausgewählt, die sowohl bezüglich der Tabellen `Pilot` und `darf_fliegen` im Attribut `PilotId`, als auch bezüglich der Tabellen `darf_fliegen` und `Flugzeugtyp` im Attribut `FlugzeugtypId` übereinstimmen, und bei denen der Flugzeugtyp mehr als 400 Sitzplätze aufweist.

Diese werden auf den Namen des Piloten und auf die Typbezeichnung projiziert und nach dem Namen des Piloten aufsteigend sortiert ausgegeben.

Es wird also eine nach den Namen der Piloten alphabetisch aufsteigend sortierte Liste erstellt, in der Piloten zusammen mit den von ihnen geflogenen Flugzeugtypen stehen, wenn der Flugzeugtyp mehr als 400 Sitzplätze aufweist.

6.2 Teilleistungen – Kriterien

Teilaufgabe a)

	Anforderungen	**maximal erreichbare Punktzahl**
	Der Prüfling	
1	stellt die Entitäten und Beziehungen in einem ER-Diagramm dar.	3
2	bestimmt die Kardinalitäten.	2
3	begründet seine Entscheidung bezüglich der Kardinalitäten.	3
Der gewählte Lösungsansatz und -weg muss nicht identisch mit dem der Modelllösung sein. Sachlich richtige Alternativen werden an dieser Stelle mit entsprechender Punktzahl bewertet.		

Teilaufgabe b)

	Anforderungen **Der Prüfling**	**maximal erreichbare Punktzahl**
1	gibt die Bedingungen für die erste, zweite und dritte Normalform an.	6
2	erläutert, dass die Bedingungen für die Normalformen verletzt sind.	3
3	überführt das Relationenschema in die 3. Normalform und begründet das Vorgehen.	3
Der gewählte Lösungsansatz und -weg muss nicht identisch mit dem der Modelllösung sein. Sachlich richtige Alternativen werden an dieser Stelle mit entsprechender Punktzahl bewertet.		

Teilaufgabe c)

	Anforderungen **Der Prüfling**	**maximal erreichbare Punktzahl**
1	entwickelt eine SQL-Anweisung zur ersten Abfrage.	5
2	entwickelt eine SQL-Anweisung zur zweiten Abfrage.	6
3	entwickelt eine SQL-Anweisung zur dritten Abfrage.	7
Der gewählte Lösungsansatz und -weg muss nicht identisch mit dem der Modelllösung sein. Sachlich richtige Alternativen werden an dieser Stelle mit entsprechender Punktzahl bewertet.		

Teilaufgabe d)

	Anforderungen **Der Prüfling**	**maximal erreichbare Punktzahl**
1	erläutert die SQL-Anweisung.	6
2	erläutert die SQL-Anweisung.	6
Der gewählte Lösungsansatz und -weg muss nicht identisch mit dem der Modelllösung sein. Sachlich richtige Alternativen werden an dieser Stelle mit entsprechender Punktzahl bewertet.		

7. Bewertungsbogen zur Prüfungsarbeit

Name des Prüflings: _____ Kursbezeichnung: _____

Schule: _____

Teilaufgabe a)

	Anforderungen	**Lösungsqualität**			
	Der Prüfling	**maximal erreichbare Punktzahl**	**EK**[2]	**ZK**	**DK**
1	stellt die Entitäten …	3			
2	bestimmt die Kardinalitäten.	2			
3	begründet seine Entscheidung …	3			
	Summe Teilaufgabe a)	**8**			

Teilaufgabe b)

	Anforderungen	**Lösungsqualität**			
	Der Prüfling	**maximal erreichbare Punktzahl**	**EK**	**ZK**	**DK**
1	beschreibt die Bedingungen …	6			
2	erläutert, dass die …	3			
3	überführt das Relationenschema …	3			
	Summe Teilaufgabe b)	**12**			

Teilaufgabe c)

	Anforderungen	**Lösungsqualität**			
	Der Prüfling	**maximal erreichbare Punktzahl**	**EK**	**ZK**	**DK**
1	entwickelt eine SQL-Anweisung …	5			
2	entwickelt eine SQL-Anweisung …	6			
3	entwickelt eine SQL-Anweisung …	7			
	Summe Teilaufgabe c)	**18**			

[2] EK = Erstkorrektur; ZK = Zweitkorrektur; DK = Drittkorrektur

Teilaufgabe d)

	Anforderungen	**Lösungsqualität**			
	Der Prüfling	**maximal erreichbare Punktzahl**	**EK**	**ZK**	**DK**
1	erläutert die SQL-Anweisung.	6			
2	erläutert die SQL-Anweisung.	6			
	Summe Teilaufgabe d)	**12**			

	Summe insgesamt	**50**			

Festlegung der Gesamtnote (Bitte nur bei der letzten bearbeiteten Aufgabe ausfüllen.)

	Lösungsqualität			
	maximal erreichbare Punktzahl	**EK**	**ZK**	**DK**
Übertrag der Punktsumme aus der ersten bearbeiteten Aufgabe	**50**			
Übertrag der Punktsumme aus der zweiten bearbeiteten Aufgabe	**50**			
Übertrag der Punktsumme aus der dritten bearbeiteten Aufgabe	**50**			
Punktzahl der gesamten Prüfungsleistung	**150**			
aus der Punktsumme resultierende Note				
Note ggf. unter Absenkung um ein bis zwei Notenpunkte gemäß § 13 Abs. 2 APO-GOSt				
Paraphe				

ggf. arithmetisches Mittel der Punktsummen aus EK und ZK: _____

ggf. arithmetisches Mittel der Notenurteile aus EK und ZK: _____

Die Klausur wird abschließend mit der Note: _____ (____ Punkte) bewertet.

Unterschrift, Datum:

Grundsätze für die Bewertung (Notenfindung)

Für die Zuordnung der Notenstufen zu den Punktzahlen ist folgende Tabelle zu verwenden:

Note	**Punkte**	**Erreichte Punktzahl**
sehr gut plus	15	150 – 143
sehr gut	14	142 – 135
sehr gut minus	13	134 – 128
gut plus	12	127 – 120
gut	11	119 – 113
gut minus	10	112 – 105
befriedigend plus	9	104 – 98
befriedigend	8	97 – 90
befriedigend minus	7	89 – 83
ausreichend plus	6	82 – 75
ausreichend	5	74 – 68
ausreichend minus	4	67 – 58
mangelhaft plus	3	57 – 49
mangelhaft	2	48 – 40
mangelhaft minus	1	39 – 30
ungenügend	0	29 – 0