

Thema: „Wie werden Aggregatfunktionen in MySQL eingesetzt?“

Unterrichtsfach:	IV: „Datenbanken“
Schulform:	Berufliches Gymnasium, Oberstufe
Klasse:	BGY W 14
Unterrichtsfach:	IV: „Datenbanken“

1. Phasen der Lernschleife

4. Produktive Erarbeitungsphase (9 Min.):

In der PAIR-Phase sollen die Schüler ihre erarbeiteten SQL-Anweisungen miteinander vergleichen und ergänzen evtl. die eigenen Lösungen.

3. Produktive Erarbeitungsphase (12 Min.):

Die Schüler formulieren in der THINK-Phase mithilfe des ausgeteilten Arbeits- und Informationsblattes die jeweiligen SQL-Anweisungen. Der Lehrer geht in dieser Phase durch die Reihen und bietet den einzelnen Schüler Hilfestellung an.

5. Präsentations- und Besprechungsphase (11 Min.):

Der Lehrer stellt den Schülern frei, wer seine Ergebnisse vor dem Plenum präsentieren bzw. vorstellen möchte. Falls es zu keiner Meldung kommen sollte, wird der Lehrer jeweils einen Schüler bestimmen.

Der Schüler erklärt seine SQL-Anweisungen und führt diese am Lehrerrechner mithilfe des Digitalprojektors vor.

Schülerfragen werden vom Präsentierenden beantwortet. Der Lehrer gibt in dieser Phase ggf. Hilfestellung. In dieser Zeit ergänzen bzw. verbessern die anderen Schüler ihre Ergebnisse.

1. Einstiegssituation (3 Min.):

Die Lernaufgabe (Fallbasierter Einstieg), wird den Schülern mithilfe des Digitalprojektors auf die Projektionsfläche übermittelt und der Lehrer spielt bzw. trägt eine praxisnahe Situation laut und deutlich vor. Hierbei handelt es sich um mehrere Problemstellungen, die mithilfe von SQL-Anweisungen gelöst werden sollen.

Die Lernaufgabe wirft folgende Frage auf: *Wie werden die Aggregatfunktionen in MySQL eingesetzt?*

2. Problemstrukturierung und Auftragsübergabe (6 Min.):

Der Lehrer wendet die Frage- und Impulstechnik an („Wer kann mit eigenen Worten die Problematik zusammenfassen?“, „Welche Gemeinsamkeit haben diese Aufträge?“). Die Schüler sollen die Lernaufgabe mit eigenen Worten erläutern und dadurch die Zielsetzung formulieren. In dieser Phase sollen die Schüler erkennen, dass sie mit ihrem bereits erlernten Wissen nicht weiterkommen, vorerst an ihre Grenzen stoßen und daher sich neues Wissen aneignen müssen. Anschließend wird der Lehrer die Schüler in Partner aufteilen und die Arbeitsblätter durch zwei Schüler austeilen lassen, die sie dann in Einzelarbeit an einem PC lösen sollen. Das Informationsblatt bekommen die Schüler über die Lernplattform „Moodle“ zur Verfügung gestellt.

6. Reflexion (4 Min.):

Am Ende der Stunde bekommen die Schüler eine Hausaufgabe. Anschließend sollen die Schüler mit der Frage „Wie praxisrelevant sie diesen Fall erachten und was sie gelernt haben?“ ein Feedback dem Lehrer geben. Zusätzlicher Hinweis auf die Unterrichtsmaterialien in der Lernplattform erfolgt durch den Lehrer.



2. Didaktische Analyse

2.1. Konzeptioneller Schwerpunkt

Die Schwerpunktsetzung hinsichtlich der zu erreichenden Kompetenzen liegt in der Unterrichtsstunde auf der in Kapitel 2.2 genannten Fachkompetenz. Darüber hinaus ist auch die Sozialkompetenz hinsichtlich einer Vorbereitung der Schüler auf das anstehende Abschlussprojekt (Kolloquium inkl. Präsentation) zu fördern. Hierbei ist es wichtig, dass die Schüler sich über gewisse Fachinhalte kommunikativ miteinander austauschen, die eigenen erarbeiteten Ergebnisse aus einer weiteren Perspektive beleuchten und ggf. diese überarbeiten.

2.2. Fachkompetenz

Die Schüler erweitern ihre **Fachkompetenz**, die verschiedenen Aggregatfunktionen anhand von Abfragen mit der Abfragesprache SQL anzuwenden, indem sie zuerst die SELECT-Abfrage strukturiert aufbauen sowie deren Umsetzung in der Datenbankumgebung MySQL vornehmen.

2.3. Sozialkompetenz

Die Schüler erweitern ihre **Sozialkompetenz**, sich mit einer weiteren Person über ihre SQL-Anweisungen auseinanderzusetzen, indem sie sich während des Think-Pair-Share-Verfahrens in der Pair-Phase (Partnerarbeit) ihre eigens erarbeiteten Ergebnisse vorstellen, begründen und möglicherweise anpassen.

II. Anhang

1. Titelbild (PowerPoint-Folie 1)

SQLSolution GmbH

2. Lernaufgabe (PowerPoint-Folie 2)

Klasse: BGY W 14

Fach: IV

Lehrer: Herr Kunz

Sie sollen für die Chefetage folgende Informationen aus der Datenbank auslesen:

- Wie hoch sind die jährlichen Kosten für den teuersten Dienstwagen?

→ Funktion: MAX()

3. Lernaufgabe (PowerPoint-Folie 3)

Klasse: BGY W 14

Fach: IV

Lehrer: Herr Kunz

Sie sollen für die Chefetage folgende Informationen aus der Datenbank auslesen:

- Ermitteln Sie den Wert mit den höchsten monatlichen Kosten im gesamten Fuhrpark.
- Ermitteln Sie den Wert mit den niedrigsten monatlichen Kosten im gesamten Fuhrpark.
- Wie hoch sind die monatlichen Kosten für den gesamten Fuhrpark?
- Wie viele Fahrzeuge sind im Fuhrpark vorhanden?
- Wie hoch sind die durchschnittlichen monatlichen Kosten im gesamten Fuhrpark?
- Wie hoch sind die jährlichen Kosten für den teuersten Dienstwagen?
- Wie hoch sind die jährlichen Kosten für den gesamten Fuhrpark?
- Welcher Dienstwagen ist der teuerste im gesamten Fuhrpark?
- Welcher Dienstwagen besitzt die geringsten monatlichen Kosten im gesamten Fuhrpark?

4. Arbeitsblatt mit Erwartungshorizont

Klasse: HBF IT 12

Fach: BBU – LB 10

Lehrer: Herr Kunz

Arbeitsblatt zum Thema: „Wie werden Aggregatfunktionen in MySQL eingesetzt?“



1. Arbeitsauftrag	SQL-Anweisungen formulieren	Bearbeitungszeit: 12 Minuten	
-------------------	-----------------------------	---------------------------------	--

THINK: Führen Sie folgende Tätigkeiten in Einzelarbeit durch:

1. Nehmen Sie das Informationsblatt „Aggregatfunktionen in MySQL“ zur Hand und lesen Sie dieses genau durch.
2. Formulieren Sie zu den folgenden Aufträgen die SQL-Anweisungen und notieren Sie diese auf diesem Blatt:
 - a) Ermitteln Sie den Wert mit den höchsten monatlichen Kosten im gesamten Fuhrpark.
`SELECT MAX(KostenproMonat) AS "maximale Kosten pro Monat" FROM dienstwagen;`
 - b) Ermitteln Sie den Wert mit den niedrigsten monatlichen Kosten im gesamten Fuhrpark.
`SELECT MIN(KostenproMonat) AS "minimale Kosten pro Monat" FROM dienstwagen;`
 - c) Wie hoch sind die monatlichen Kosten für den gesamten Fuhrpark?
`SELECT SUM(KostenproMonat) AS "Gesamtkosten pro Monat" FROM dienstwagen;`
 - d) Wie viele Fahrzeuge sind im Fuhrpark vorhanden?
`SELECT COUNT(Fahrzeugmodell) AS "Anzahl Fahrzeugmodelle" FROM dienstwagen;`
 - e) Wie hoch sind die durchschnittlichen monatlichen Kosten im gesamten Fuhrpark?
`SELECT AVG(KostenproMonat) AS "durchschnittliche Monatskosten" FROM dienstwagen;`
`SELECT ROUND(AVG(KostenproMonat),2) AS "durchschnittliche Monatskosten" FROM dienstwagen;`
 - f) Wie hoch sind die jährlichen Kosten für den teuersten Dienstwagen?
`SELECT MAX(KostenproMonat)*12 AS "Gesamtkosten pro Jahr" FROM dienstwagen;`
 - g) Wie hoch sind die jährlichen Kosten für den gesamten Fuhrpark?
`SELECT SUM(KostenproMonat)*12 AS "Gesamtkosten im Jahr" FROM dienstwagen;`
 - h) Welcher Dienstwagen ist der teuerste im gesamten Fuhrpark?
`SELECT Fahrzeugmodell, Kennzeichen, MAX(KostenproMonat) AS "höchste Kosten" FROM dienstwagen WHERE KostenproMonat = (SELECT MAX(KostenproMonat) FROM dienstwagen);`
 - i) Welcher Dienstwagen besitzt die geringsten monatlichen Kosten im gesamten Fuhrpark?
`SELECT Fahrzeugmodell, Kennzeichen, MIN(KostenproMonat) AS "niedrigste Kosten" FROM dienstwagen WHERE KostenproMonat = (SELECT MIN(KostenproMonat) FROM dienstwagen);`

2. Arbeitsauftrag	SQL-Anweisungen vergleichen	Bearbeitungszeit: 9 Minuten	
-------------------	-----------------------------	--------------------------------	--

PAIR: Gehen Sie nun in die Partnerarbeit:

1. Vergleichen Sie Ihre SQL-Anweisungen miteinander.
2. Überarbeiten Sie eventuell diese SQL-Anweisungen.

3. Arbeitsauftrag	SQL-Anweisungen vorstellen	Bearbeitungszeit: 11 Minuten	
-------------------	----------------------------	---------------------------------	--

SHARE: Der Lehrer wählt die Personen aus der Klasse aus!

➔ Der Schüler präsentiert seine Ergebnisse.

5. Informationsblatt

Klasse: HBF IT 12

Fach: BBU – LB 10

Lehrer: Herr Kunz



Informationsblatt zum Thema: „Wie werden Aggregatfunktionen in MySQL eingesetzt?“

Mithilfe von Aggregat- oder Gruppenfunktionen können Berechnungen an Gruppen von Daten durchgeführt und Berechnungsergebnisse angezeigt werden.

Folgende Funktionen werden vorgestellt: MAX (), MIN (), SUM (), COUNT (), AVG ()

Die Funktionen beziehen sich auf die folgende Tabelle „Speisen“:

```
mysql> SELECT * FROM Speisen;
```

id	Art	Gericht	Preis
101	Pizza	Pizza Diabolo	5.50
102	Pizza	Pizza Vulkano	6.00
103	Pizza	Pizza Feuro	6.50
104	Nudeln	Lasagno	6.00
105	Salat	Eskimo	4.50
106	Nudeln	Spaghetti Vongole	7.90
107	Nudeln	Tortellini Carbonara	7.90
108	Nudeln	Tagliatelle Lachs	9.50
109	Salat	HOT	6.50
110	Salat	EUROPE	5.00

Die Funktion MIN () gibt den kleinsten Wert dieses Attributs zurück.	<pre>mysql> SELECT MIN(Preis) FROM Speisen;</pre> <table><tr><th>MIN(Preis)</th></tr><tr><td>4.50</td></tr></table>	MIN(Preis)	4.50
MIN(Preis)			
4.50			
Die Funktion MAX () gibt den größten Wert dieses Attributs zurück.	<pre>mysql> SELECT MAX(Preis) FROM Speisen;</pre> <table><tr><th>MAX(Preis)</th></tr><tr><td>9.50</td></tr></table>	MAX(Preis)	9.50
MAX(Preis)			
9.50			
Die Funktion SUM () summiert alle Werte in dieser Spalte auf.	<pre>mysql> SELECT SUM(Preis) FROM Speisen;</pre> <table><tr><th>SUM(Preis)</th></tr><tr><td>65.30</td></tr></table>	SUM(Preis)	65.30
SUM(Preis)			
65.30			
Die Funktion COUNT () zählt die unterschiedlichen Datensätze zu diesem Attribut.	<pre>mysql> SELECT COUNT(Preis) FROM Speisen;</pre> <table><tr><th>COUNT(Preis)</th></tr><tr><td>10</td></tr></table>	COUNT(Preis)	10
COUNT(Preis)			
10			
Die Funktion AVG () bildet den Mittelwert dieses Attributs.	<pre>mysql> SELECT AVG(Preis) FROM Speisen;</pre> <table><tr><th>AVG(Preis)</th></tr><tr><td>6.530000</td></tr></table>	AVG(Preis)	6.530000
AVG(Preis)			
6.530000			

Zusatzinformation:

Die numerische Funktion ROUND () rundet eine Spalte, einen Ausdruck oder einen Wert auf n Dezimalstellen. Ist kein Wert für n angegeben, wird auf einen ganzzahligen Wert gerundet.	<pre>mysql> SELECT ROUND(53.923,2), ROUND(45.923,0) FROM DUAL;</pre> <table><thead><tr><th>ROUND(53.923,2)</th><th>ROUND(45.923,0)</th></tr></thead><tbody><tr><td>53.92</td><td>46</td></tr></tbody></table>	ROUND(53.923,2)	ROUND(45.923,0)	53.92	46
ROUND(53.923,2)	ROUND(45.923,0)				
53.92	46				

6. Hausaufgabe mit Erwartungshorizont

Klasse: HBF IT 12

Fach: BBU – LB 10

Lehrer: Herr Kunz

Hausaufgabenblatt zum Thema: „Wie werden Aggregatfunktionen in MySQL eingesetzt?“



Formulieren Sie zu den folgenden Auftrag die SQL-Anweisung:

- 1) Welches ist das jüngste Fahrzeug im gesamten Fuhrpark?

```
SELECT Fahrzeugmodell, Kennzeichen, MAX(Erstzulassung) AS "jüngste Fahrzeug" FROM dienstwagen  
WHERE Erstzulassung = (SELECT MAX(Erstzulassung) FROM dienstwagen);
```

- 2) Welches ist das älteste Fahrzeug im Unternehmen?

```
SELECT Fahrzeugmodell, Kennzeichen, MIN(Erstzulassung) AS "älteste Fahrzeug" FROM dienstwagen;
```

- 3) Eine Liste mit den höchsten monatlichen Kosten für jedes Dienstwagenmodell.

```
SELECT Fahrzeugtyp, MAX(KostenproMonat) AS "höchsten Kosten" FROM dienstwagen GROUP BY  
Fahrzeugtyp;
```

- 4) Eine Liste mit den geringsten monatlichen Kosten für jedes Dienstwagenmodell.

```
SELECT Fahrzeugtyp, MIN(KostenproMonat) AS "niedrigsten Kosten" FROM dienstwagen GROUP BY  
Fahrzeugtyp;
```

- 5) Eine Liste der Fahrzeugtypen, welche die Anzahl der jeweiligen Fahrzeugmodelle anzeigt.

```
SELECT Fahrzeugtyp, COUNT(Fahrzeugmodell) AS "Anzahl Fahrzeugmodelle" FROM dienstwagen GROUP BY  
Fahrzeugtyp;
```

- 6) Eine Liste der Fahrzeugtypen mit deren aufsummierte monatlichen Kosten.

```
SELECT Fahrzeugtyp, SUM(KostenproMonat) AS "Summe der monatlichen Kosten" FROM dienstwagen  
GROUP BY Fahrzeugtyp;
```

- 7) Eine Liste mit den ermittelten durchschnittlichen Monatskosten für jeden Fahrzeugtyp.

```
SELECT Fahrzeugtyp, AVG(KostenproMonat) AS "durchschnittliche Monatskosten" FROM dienstwagen  
GROUP BY Fahrzeugtyp;  
  
SELECT Fahrzeugtyp, ROUND(AVG(KostenproMonat),2) AS "durchschnittliche Monatskosten" FROM  
dienstwagen GROUP BY Fahrzeugtyp;
```

- 8) Eine Auflistung der monatlichen und jährlichen Kosten für jeden Fahrzeugtyp.

```
SELECT Fahrzeugtyp, SUM(KostenproMonat) AS "Monatskosten", SUM(KostenproMonat * 12) AS  
"Jahreskosten" FROM dienstwagen GROUP BY Fahrzeugtyp;
```

- 9) Eine Auflistung der monatlichen und jährlichen Kosten für jedes Fahrzeugmodell.

```
SELECT Fahrzeugtyp, Fahrzeugmodell, SUM(KostenproMonat) AS "Monatskosten", SUM(KostenproMonat *  
12) AS "Jahreskosten" FROM dienstwagen GROUP BY Fahrzeugmodell;
```


III. Literaturverzeichnis

DuBois P.: MySQL Kochbuch, O'Reilly Verlag, 1. Auflage 2003.

Landesakademie für Fortbildung und Personalentwicklung an Schulen, Think-Pair-Share:
<http://lehrerfortbildung-bw.de/bs/bsueb/if/unterrichtsgestaltung/methodenblaetter/thinkpairshare.html>, abgerufen am 02.11.2015.

Mattes Wolfgang: Methoden für den Unterricht, Schöningh Verlag, 2009.

Kersken Sascha: Praktischer Einstieg in MySQL mit PHP, O'Reilly Verlag, Köln, 2007, 2. Auflage, http://examples.oreilly.de/openbooks/pdf_einmysql2ger.pdf, abgerufen am 02.11.2015.

Stundenablaufplanung zum Thema: „Wie werden Aggregatfunktionen in MySQL eingesetzt?“

Instruktionen	Monitor	Zeit	Uhrzeit	Phase
<ul style="list-style-type: none"> • Begrüßung aller Teilnehmer 	aus	1	08:36	Lernaufgabe
<ul style="list-style-type: none"> • Die Lernaufgabe wird vorgelesen. Diese wird mit dem Digitalprojektor an die Wand projiziert und liegt in ausgedruckter Form zur Verfügung. Die Schüler passen auf. 	aus	2	08:38	
<ul style="list-style-type: none"> • Frage- und Impulstechnik • „Wer kann mit eigenen Worten die Problematik zusammenfassen?“ • Ein Schüler fasst die Problemstellung zusammen. 	aus	3	08:41	Problemstrukturierung
<ul style="list-style-type: none"> • Welche Gemeinsamkeit haben diese Aufträge? • ⇒ Berechnung an Gruppen von Daten • Thema angeben und an die Tafel schreiben: „Wie werden Aggregatfunktionen in MySQL eingesetzt?“ • „Was benötigen Sie (Welche Informationen?), um diese Aufgaben lösen zu können?“ ⇒ Informationen bzgl. Aggregat- oder Gruppenfunktionen • Festlegen der benötigten Informationen für die Aufstellung der Datenbankabfragen. 	aus	2	08:43	
<ul style="list-style-type: none"> • Einteilung für die THINK-PAIR-SHARE-Methode • Anmeldung am System • Austeilen des Informationsblattes und des Arbeitsblattes, das die einzelnen Arbeitsaufträge beinhaltet. 	an	1	08:44	
<ul style="list-style-type: none"> • THINK: 1. Arbeitsauftrag. • Einzelarbeit: <i>SQL-Anweisungen formulieren</i> • Der Lehrer gibt in dieser Zeit Hilfestellung. 	an	12	08:56	Erarbeitung
<ul style="list-style-type: none"> • PAIR: 2. Arbeitsauftrag • Partnerarbeit: <i>SQL-Anweisungen vergleichen</i> • Die Schüler vergleichen ihre ausgearbeiteten SQL-Anweisungen miteinander und ergänzen evtl. die eigenen Lösungen 	an	9	09:05	
<ul style="list-style-type: none"> • SHARE: 3. Arbeitsauftrag • Der Schüler erklärt seine SQL-Anweisungen und führt diese am Lehrerrechner mithilfe des Digitalprojektors vor. • Schülerfragen werden vom Präsentierenden beantwortet. • Der Lehrer gibt ggf. Hilfestellung. • In dieser Zeit ergänzen/verbessern die anderen Schüler ihre Ergebnisse. 	an	11	09:16	Präsentation und Besprechung
<ul style="list-style-type: none"> • Feedback zur Methode „Think-Pair-Share“ der Schüler einholen („Wo haben die Schüler Probleme gehabt?“, „Sind die Schüler in der Methode auf Probleme gestoßen?“) • Hausaufgabe in Moodle freischalten 	an	4	09:20	Reflexion

Aggregatfunktionen erlaubt. Das sind Funktionen, die die Zusammenfassung unterstützen, etwa für das Zählen von Datensätzen, Bilden von Summen oder Errechnen von Durchschnittswerten.

Lernaufgabe: Zu den Funktionen:

Steuerungsfunktion und Motivationsfunktion: Die Azubis kommen demnächst in die Prüfung. Der Chef spielt mit dem Gedanken, sie nach der Ausbildung zu übernehmen.

Er kommt noch einmal über den Abteilungsleiter auf sie zu und gibt Ihnen ein paar Fragestellung, die zum weiteren Handeln bzw. Planung in dem Unternehmen benötigt werden.

Anschauungsfunktion und Informationsfunktion: Es wird mit der bekannten Datenbank gearbeitet.

Ebenfalls wird ein Informationsblatt (digital und in ausgedruckter Form), dass das Fachwissen beinhaltet, zur Verfügung gestellt. Der Arbeitsauftrag rundet die Anschauungsfunktion ab.

Zielfunktion: Das didaktische Konzept trägt die Zielfunktion und richtet sich nach der Zielorientierung des Schülers.

Allgemeine Begründung:

- Die Problemstellung wurde gelöst.
- Die Motivation war durch den Fachbezug gegeben.
- Information erfolgte durch entsprechend aufbereitete Materialien.
- Anschauung wurde durch Zeigen am PC in MySQL hergestellt.
- Das Ziel war klar formuliert.
- Anschluss an vorangegangene Stunden wurde durch die verwendete Datentabelle hergestellt.
- Den Schülern wurde die Wissenslücke aufgezeigt.

Motivierung:

- Die Motivation erfolgte durch den Fachinhalt, was bei der Lerngruppe gelang.
- Die Schüler waren bis zum Ende der Stunde motiviert.
- Motive wurden ebenfalls durch die laufenden Projektarbeiten geliefert.

Kommunizieren und Interagieren:

- Meine Stimmmodulation habe ich optimiert bzw. verbessert.
- Feedback und Reflexion waren ausgeprägt.
- Es wurden mehrere Alternativlösungen bedacht.


Aktivierung:

- Während der gesamten Stunde war eine hohe Schüleraktivität zu beobachten.
- Die Balance Think/Pair wurde besser beachtet.
- Selbstkontrolle wurde durch die Schüler genutzt.
- Handlungsspielräume wurden ebenfalls durch die Lerngruppe genutzt.

Kompetenz- und Wirkungsorientierung:

- Die angestrebte Fach- und Sozialkompetenz wurde im angestrebten Rahmen entwickelt.
- Die Arbeitsphase wurde mit Blick auf die Validierung der Sozialkompetenzentwicklung reflektiert.

Arbeitsblatt zum Thema: „Wie werden Aggregatfunktionen in MySQL eingesetzt?“

1. Arbeitsauftrag	<i>SQL-Anweisungen formulieren</i>	Bearbeitungszeit: 12 Minuten	
--------------------------	------------------------------------	--	---

THINK: Führen Sie folgende Tätigkeiten in **Einzelarbeit** durch:

1. Nehmen Sie das Informationsblatt „Aggregatfunktionen in MySQL“ zur Hand und lesen Sie dieses genau durch.
2. Formulieren Sie zu den folgenden Aufträgen die SQL-Anweisungen und notieren Sie diese auf diesem Blatt:

a) Ermitteln Sie den Wert mit den höchsten monatlichen Kosten im gesamten Fuhrpark.

b) Ermitteln Sie den Wert mit den niedrigsten monatlichen Kosten im gesamten Fuhrpark.

c) Wie hoch sind die monatlichen Kosten für den gesamten Fuhrpark?

d) Wie viele Fahrzeuge sind im Fuhrpark vorhanden?

e) Wie hoch sind die durchschnittlichen monatlichen Kosten im gesamten Fuhrpark?


f) Wie hoch sind die jährlichen Kosten für den teuersten Dienstwagen?

g) Wie hoch sind die jährlichen Kosten für den gesamten Fuhrpark?

h) Welcher Dienstwagen ist der teuerste im gesamten Fuhrpark?


i) Welcher Dienstwagen besitzt die geringsten monatlichen Kosten im gesamten Fuhrpark?

Arbeitsblatt zum Thema: „Wie werden Aggregatfunktionen in MySQL eingesetzt?“

2. Arbeitsauftrag	<i>SQL-Anweisungen vergleichen</i>	Bearbeitungszeit: 9 Minuten	
--------------------------	------------------------------------	---------------------------------------	---

PAIR: Gehen Sie nun in die **Partnerarbeit:**


1. Vergleichen Sie Ihre SQL-Anweisungen miteinander.
2. Überarbeiten Sie eventuell diese SQL-Anweisungen.

3. Arbeitsauftrag	<i>SQL-Anweisungen vorstellen</i>	Bearbeitungszeit: 11 Minuten	
--------------------------	-----------------------------------	--	---

SHARE: Der Lehrer wählt **die Personen** aus der Klasse aus!


- ➔ Der Schüler präsentiert seine Ergebnisse.

Arbeitsblatt zum Thema: „Wie werden Aggregatfunktionen in MySQL eingesetzt?“

1. Arbeitsauftrag	<i>SQL-Anweisungen formulieren</i>	Bearbeitungszeit: 12 Minuten	
--------------------------	------------------------------------	--	---


THINK: Führen Sie folgende Tätigkeiten in **Einzelarbeit** durch:

1. Nehmen Sie das Informationsblatt „Aggregatfunktionen in MySQL“ zur Hand und lesen Sie dieses genau durch.
2. Formulieren Sie zu den folgenden Aufträgen die SQL-Anweisungen und notieren Sie diese auf diesem Blatt:
 - a) Ermitteln Sie den Wert mit den höchsten monatlichen Kosten im gesamten Fuhrpark.
SELECT MAX(KostenproMonat) AS "maximale Kosten pro Monat" FROM dienstwagen;
 - b) Ermitteln Sie den Wert mit den niedrigsten monatlichen Kosten im gesamten Fuhrpark.
SELECT MIN(KostenproMonat) AS "minimale Kosten pro Monat" FROM dienstwagen;
 - c) Wie hoch sind die monatlichen Kosten für den gesamten Fuhrpark?
SELECT SUM(KostenproMonat) AS "Gesamtkosten pro Monat" FROM dienstwagen;
 - d) Wie viele Fahrzeuge sind im Fuhrpark vorhanden?
SELECT COUNT(Fahrzeugmodell) AS "Anzahl Fahrzeugmodelle" FROM dienstwagen;
 - e) Wie hoch sind die durchschnittlichen monatlichen Kosten im gesamten Fuhrpark?
SELECT AVG(KostenproMonat) AS "durchschnittliche Monatskosten" FROM dienstwagen;
SELECT ROUND(AVG(KostenproMonat),2) AS "durchschnittliche Monatskosten" FROM dienstwagen;
 - f) Wie hoch sind die jährlichen Kosten für den teuersten Dienstwagen?
SELECT MAX(KostenproMonat)*12 AS "Gesamtkosten pro Jahr" FROM dienstwagen;
 - g) Wie hoch sind die jährlichen Kosten für den gesamten Fuhrpark?
SELECT SUM(KostenproMonat)*12 AS "Gesamtkosten im Jahr" FROM dienstwagen;
 - h) Welcher Dienstwagen ist der teuerste im gesamten Fuhrpark?
SELECT Fahrzeugmodell, Kennzeichen, MAX(KostenproMonat) AS "höchste Kosten" FROM dienstwagen WHERE KostenproMonat = (SELECT MAX(KostenproMonat) FROM dienstwagen);
 - i) Welcher Dienstwagen besitzt die geringsten monatlichen Kosten im gesamten Fuhrpark?
SELECT Fahrzeugmodell, Kennzeichen, MIN(KostenproMonat) AS "niedrigste Kosten" FROM dienstwagen WHERE KostenproMonat = (SELECT MIN(KostenproMonat) FROM dienstwagen);

2. Arbeitsauftrag	<i>SQL-Anweisungen vergleichen</i>	Bearbeitungszeit: 9 Minuten	
--------------------------	------------------------------------	---------------------------------------	---

PAIR: Gehen Sie nun in die **Partnerarbeit**:

1. Vergleichen Sie Ihre SQL-Anweisungen miteinander.
2. Überarbeiten Sie eventuell diese SQL-Anweisungen.

3. Arbeitsauftrag	<i>SQL-Anweisungen vorstellen</i>	Bearbeitungszeit: 11 Minuten	
--------------------------	-----------------------------------	--	---

SHARE: Der Lehrer wählt **die Personen** aus der Klasse aus!

➔ Der Schüler präsentiert seine Ergebnisse.

Informationsblatt zum Thema: „Wie werden Aggregatfunktionen in MySQL eingesetzt?“

Mithilfe von Aggregat- oder Gruppenfunktionen können Berechnungen an Gruppen von Daten durchgeführt und Berechnungsergebnisse angezeigt werden.

Folgende Funktionen werden vorgestellt: MAX(), MIN(), SUM(), COUNT(), AVG()

Die Funktionen beziehen sich auf die folgende Tabelle „Speisen“:

```
mysql> SELECT * FROM Speisen;
```

id	Art	Gericht	Preis
101	Pizza	Pizza Diabolo	5.50
102	Pizza	Pizza Vulkano	6.00
103	Pizza	Pizza Feuro	6.50
104	Nudeln	Lasagno	6.00
105	Salat	Eskimo	4.50
106	Nudeln	Spaghetti Vongole	7.90
107	Nudeln	Tortellini Carbonara	7.90
108	Nudeln	Tagliatelle Lachs	9.50
109	Salat	HOT	6.50
110	Salat	EUROPE	5.00

Die Funktion <code>MIN()</code> gibt den kleinsten Wert dieses Attributs zurück.	<pre>mysql> SELECT MIN(Preis) FROM Speisen; +-----+ MIN(Preis) +-----+ 4.50 +-----+</pre>
Die Funktion <code>MAX()</code> gibt den größten Wert dieses Attributs zurück.	<pre>mysql> SELECT MAX(Preis) FROM Speisen; +-----+ MAX(Preis) +-----+ 9.50 +-----+</pre>
Die Funktion <code>SUM()</code> summiert alle Werte in dieser Spalte auf.	<pre>mysql> SELECT SUM(Preis) FROM Speisen; +-----+ SUM(Preis) +-----+ 65.30 +-----+</pre>
Die Funktion <code>COUNT()</code> zählt die unterschiedlichen Datensätze zu diesem Attribut.	<pre>mysql> SELECT COUNT(Preis) FROM Speisen; +-----+ COUNT(Preis) +-----+ 10 +-----+</pre>
Die Funktion <code>AVG()</code> bildet den Mittelwert dieses Attributs.	<pre>mysql> SELECT AVG(Preis) FROM Speisen; +-----+ AVG(Preis) +-----+ 6.530000 +-----+</pre>

Zusatzinformation:

Die numerische Funktion ROUND() rundet eine Spalte, einen Ausdruck oder einen Wert auf n Dezimalstellen. Ist kein Wert für n angegeben, wird auf einen ganzzahligen Wert gerundet.	<pre>mysql> SELECT ROUND(53.923,2), ROUND(45.923,0) FROM DUAL;</pre> <table> <tr> <th>ROUND(53.923,2)</th><th>ROUND(45.923,0)</th></tr> <tr> <td>53.92</td><td>46</td></tr> </table>	ROUND(53.923,2)	ROUND(45.923,0)	53.92	46
ROUND(53.923,2)	ROUND(45.923,0)				
53.92	46				

a) Ermitteln Sie den Wert mit den höchsten monatlichen Kosten im gesamten Fuhrpark.

```
SELECT MAX(KostenproMonat) AS "maximale Kosten pro Monat" FROM dienstwagen;
```

b) Ermitteln Sie den Wert mit den niedrigsten monatlichen Kosten im gesamten Fuhrpark.

```
SELECT MIN(KostenproMonat) AS "minimale Kosten pro Monat" FROM dienstwagen;
```

c) Wie hoch sind die monatlichen Kosten pro Monat für den gesamten Fuhrpark?

```
SELECT SUM(KostenproMonat) AS "Gesamtkosten pro Monat" FROM dienstwagen;
```

d) Wie viele Fahrzeuge sind im Fuhrpark vorhanden?

```
SELECT COUNT(Fahrzeugmodell) AS "Anzahl Fahrzeugmodelle" FROM dienstwagen;
```

e) Wie hoch sind die durchschnittlichen monatlichen Kosten im gesamten Fuhrpark?

```
SELECT AVG(KostenproMonat) AS "durchschnittliche Monatskosten" FROM dienstwagen;
```

```
SELECT ROUND(AVG(KostenproMonat),2) AS "durchschnittliche Monatskosten" FROM dienstwagen;
```

f) Welcher Dienstwagen ist der teuerste im gesamten Fuhrpark?

```
SELECT Fahrzeugmodell, Kennzeichen, MAX(KostenproMonat) AS "höchsten Kosten" FROM dienstwagen WHERE  
KostenproMonat = (SELECT MAX(KostenproMonat) FROM dienstwagen);
```

g) Welcher Dienstwagen besitzt die geringsten monatlichen Kosten im gesamten Fuhrpark?

```
SELECT Fahrzeugmodell, Kennzeichen, MIN(KostenproMonat) AS "niedrigsten Kosten" FROM dienstwagen WHERE  
KostenproMonat = (SELECT MIN(KostenproMonat) FROM dienstwagen);
```

h) Welches Fahrzeug ist das Jüngste im gesamten Fuhrpark?

```
SELECT Fahrzeugmodell, Kennzeichen, MAX(Erstzulassung) AS "jüngste Fahrzeug" FROM dienstwagen WHERE  
Erstzulassung = (SELECT MAX(Erstzulassung) FROM dienstwagen);
```

i) Welches Fahrzeug ist das Älteste im Unternehmen?

```
SELECT Fahrzeugmodell, Kennzeichen, MIN(Erstzulassung) AS "älteste Fahrzeug" FROM dienstwagen;
```

j) Eine Liste mit den höchsten monatlichen Kosten für jedes Dienstwagenmodell.

```
SELECT Fahrzeugtyp, MAX(KostenproMonat) AS "höchsten Kosten" FROM dienstwagen GROUP BY Fahrzeugtyp;
```

k) Eine Liste mit den geringsten monatlichen Kosten für jedes Dienstwagenmodell.

```
SELECT Fahrzeugtyp, MIN(KostenproMonat) AS "niedrigsten Kosten" FROM dienstwagen GROUP BY Fahrzeugtyp;
```

l) Eine Liste der Fahrzeugtypen, welche die Anzahl der jeweiligen Fahrzeugmodelle anzeigt.

```
SELECT Fahrzeugtyp, COUNT(Fahrzeugmodell) AS "Anzahl Fahrzeugmodelle" FROM dienstwagen GROUP BY Fahrzeugtyp;
```

m) Eine Liste der Fahrzeugtypen mit deren aufsummierte monatlichen Kosten.

```
SELECT Fahrzeugtyp, SUM(KostenproMonat) AS "Summe der monatlichen Kosten" FROM dienstwagen GROUP BY Fahrzeugtyp;
```

n) Eine Liste mit den ermittelten durchschnittlichen Monatskosten für jeden Fahrzeugtyp.

```
SELECT Fahrzeugtyp, AVG(KostenproMonat) AS "durchschnittliche Monatskosten" FROM dienstwagen GROUP BY Fahrzeugtyp;
```

```
SELECT Fahrzeugtyp, ROUND(AVG(KostenproMonat),2) AS "durchschnittliche Monatskosten" FROM dienstwagen GROUP BY  
Fahrzeugtyp;
```

o) Eine Auflistung der monatlichen und jährlichen Kosten für jeden Fahrzeugtyp.

```
SELECT Fahrzeugtyp, SUM(KostenproMonat) AS "Monatskosten", SUM(KostenproMonat * 12) AS "Jahreskosten" FROM  
dienstwagen GROUP BY Fahrzeugtyp;
```

p) Eine Auflistung der monatlichen und jährlichen Kosten für jedes Fahrzeugmodell.

```
SELECT Fahrzeugtyp, Fahrzeugmodell, SUM(KostenproMonat) AS "Monatskosten", SUM(KostenproMonat * 12) AS  
"Jahreskosten" FROM dienstwagen GROUP BY Fahrzeugmodell;
```


MAX()

1. Welcher Dienstwagen (Fahrzeugmodell) ist der teuerste im gesamten Fuhrpark?

SELECT Fahrzeugmodell, MAX(KostenproMonat) AS "teuerste Fahrzeug" FROM dienstwagen WHERE KostenproMonat = (SELECT MAX(KostenproMonat) FROM dienstwagen);

2. Welches Fahrzeug ist das Jüngste im Unternehmen?

SELECT Fahrzeugmodell, MAX(Erstzulassung) AS "Jüngste Fahrzeug" FROM dienstwagen;

MIN()

3. Welche Dienstwagenmodelle der einzelnen Fahrzeugtypen besitzen die geringsten monatlichen Kosten?

SELECT Fahrzeugtyp, Fahrzeugmodell, MIN(KostenproMonat) FROM dienstwagen GROUP BY Fahrzeugtyp;

noch offen!!!!

SELECT Fahrzeugtyp, Fahrzeugmodell, KostenproMonat FROM dienstwagen WHERE KostenproMonat = (SELECT MIN(KostenproMonat) FROM dienstwagen) GROUP BY Fahrzeugtyp;

noch offen!!!!

4. Welcher Dienstwagen (Fahrzeugmodell) besitzt die geringsten monatlichen Kosten?

SELECT Fahrzeugtyp, Fahrzeugmodell, KostenproMonat FROM dienstwagen WHERE KostenproMonat = (SELECT MIN(KostenproMonat) FROM dienstwagen);

5. Eine Liste der einzelnen Fahrzeugtypen mit dem geringsten monatlichen Kosten.

SELECT Fahrzeugtyp, MIN(KostenproMonat) AS "geringste monatliche Kosten" FROM dienstwagen GROUP BY Fahrzeugtyp;

6. Welches Fahrzeug ist das Älteste im Unternehmen?

SELECT Fahrzeugmodell, MIN(Erstzulassung) AS "Älteste Fahrzeug" FROM dienstwagen;

SUM()

7. Eine List der Fahrzeugtypen mit deren aufsummierten monatlichen Kosten.

SELECT Fahrzeugtyp, SUM(KostenproMonat) FROM dienstwagen GROUP BY Fahrzeugtyp;

ok

8. Eine Auflistung der Fahrzeugtypen, deren aufsummierte monatlichen Kosten aller Modelle mehr als 1300 € betragen.

SELECT Fahrzeugtyp, SUM(KostenproMonat)>1300 AS "Kosten über 1300 €" FROM dienstwagen GROUP BY Fahrzeugtyp;

SELECT Fahrzeugtyp, KostenproMonat FROM dienstwagen WHERE SUM(KostenproMonat)>1300;

SELECT Fahrzeugtyp, KostenproMonat FROM dienstwagen;

SELECT Fahrzeugtyp, KostenproMonat, SUM(KostenproMonat) AS "Summierte Kosten pro F.-typ" FROM dienstwagen GROUP BY Fahrzeugtyp;

SELECT Fahrzeugtyp, KostenproMonat, SUM(KostenproMonat) AS "Summierte Kosten pro F.-typ" FROM dienstwagen WHERE Fahrzeugtyp = "AUDI" GROUP BY Fahrzeugtyp;

SELECT Fahrzeugtyp, KostenproMonat, SUM(KostenproMonat) AS "Summierte Kosten pro F.-typ" FROM dienstwagen WHERE SUM(KostenproMonat)>1300;

SELECT Fahrzeugtyp, KostenproMonat, SUM(KostenproMonat) AS "Summierte Kosten pro F.-typ" FROM dienstwagen WHERE SUM(KostenproMonat)>1300 GROUP BY Fahrzeugtyp;

??????????????

9. Eine Auflistung der monatlichen und jährlichen Kosten für jeden Fahrzeugtyp.

```
SELECT Fahrzeugtyp, SUM(KostenproMonat) AS "Monatskosten", SUM(KostenproMonat * 12) AS "Jahreskosten" FROM dienstwagen GROUP BY Fahrzeugtyp;
```

ok

10. Eine Auflistung der monatlichen und jährlichen Kosten für jedes Fahrzeugmodell.

```
SELECT Fahrzeugtyp, Fahrzeugmodell, SUM(KostenproMonat) AS "Monatskosten", SUM(KostenproMonat * 12) AS "Jahreskosten" FROM dienstwagen GROUP BY Fahrzeugmodell;
```

ok, wie kann ich diese Liste nach den Jahreskosten sortieren?

11. Eine Auflistung der monatlichen und jährlichen Kosten für jedes Fahrzeug. Die Liste soll aufsteigen nach den Jahreskosten sortiert werden!

```
SELECT Fahrzeugtyp, Fahrzeugmodell, SUM(KostenproMonat) AS "Monatskosten", SUM(KostenproMonat * 12) AS "Jahreskosten" FROM dienstwagen GROUP BY (KostenproMonat * 12);
```

COUNT()

12) Wieviel Fahrzeuge sind im Fuhrpark vorhanden?

```
SELECT COUNT(Fahrzeugmodell) AS "Anzahl Fahrzeugmodelle" FROM dienstwagen;
```

ok

13) Eine Liste der Fahrzeugtypen, welche die Anzahl der jeweiligen Fahrzeugmodelle anzeigt.

```
SELECT Fahrzeugtyp, COUNT(Fahrzeugmodell) AS "Anzahl Fahrzeugmodelle" FROM dienstwagen GROUP BY Fahrzeugtyp;
```

ok

AVG()

14. Eine Liste mit den ermittelten durchschnittlichen Monatskosten für jeden Fahrzeugtyp.

```
SELECT Fahrzeugtyp, AVG(KostenproMonat) AS "durchschnittliche Monatskosten" FROM dienstwagen GROUP BY Fahrzeugtyp;
```

ok

15. SELECT Fahrzeugtyp, ROUND(AVG(KostenproMonat),2) AS "durchschnittliche Monatskosten" FROM dienstwagen GROUP BY Fahrzeugtyp;

ok

Hausaufgabenblatt zum Thema: „Wie werden Aggregatfunktionen in MySQL eingesetzt?“

Formulieren Sie zu den folgenden Auftrag die SQL-Anweisung:

- 1) Welches ist das jüngste Fahrzeug im gesamten Fuhrpark?
- 2) Welches ist das älteste Fahrzeug im Unternehmen?
- 3) Eine Liste mit den höchsten monatlichen Kosten für jedes Dienstwagenmodell.
- 4) Eine Liste mit den geringsten monatlichen Kosten für jedes Dienstwagenmodell.
- 5) Eine Liste der Fahrzeugtypen, welche die Anzahl der jeweiligen Fahrzeugmodelle anzeigt.
- 6) Eine Liste der Fahrzeugtypen mit deren aufsummierte monatlichen Kosten.
- 7) Eine Liste mit den ermittelten durchschnittlichen Monatskosten für jeden Fahrzeugtyp.
- 8) Eine Auflistung der monatlichen und jährlichen Kosten für jeden Fahrzeugtyp.
- 9) Eine Auflistung der monatlichen und jährlichen Kosten für jedes Fahrzeugmodell.

Hausaufgabenblatt zum Thema: „Wie werden Aggregatfunktionen in MySQL eingesetzt?“

Formulieren Sie zu den folgenden Auftrag die SQL-Anweisung:

- 1) Welches ist das jüngste Fahrzeug im gesamten Fuhrpark?
`SELECT Fahrzeugmodell, Kennzeichen, MAX(Erstzulassung) AS "jüngste Fahrzeug" FROM dienstwagen
WHERE Erstzulassung = (SELECT MAX(Erstzulassung) FROM dienstwagen);`
- 2) Welches ist das älteste Fahrzeug im Unternehmen?
`SELECT Fahrzeugmodell, Kennzeichen, MIN(Erstzulassung) AS "älteste Fahrzeug" FROM dienstwagen;`
- 3) Eine Liste mit den höchsten monatlichen Kosten für jedes Dienstwagenmodell.
`SELECT Fahrzeugtyp, MAX(KostenproMonat) AS "höchsten Kosten" FROM dienstwagen GROUP BY
Fahrzeugtyp;`
- 4) Eine Liste mit den geringsten monatlichen Kosten für jedes Dienstwagenmodell.
`SELECT Fahrzeugtyp, MIN(KostenproMonat) AS "niedrigsten Kosten" FROM dienstwagen GROUP BY
Fahrzeugtyp;`
- 5) Eine Liste der Fahrzeugtypen, welche die Anzahl der jeweiligen Fahrzeugmodelle anzeigt.
`SELECT Fahrzeugtyp, COUNT(Fahrzeugmodell) AS "Anzahl Fahrzeugmodelle" FROM dienstwagen GROUP BY
Fahrzeugtyp;`
- 6) Eine Liste der Fahrzeugtypen mit deren aufsummierte monatlichen Kosten.
`SELECT Fahrzeugtyp, SUM(KostenproMonat) AS "Summe der monatlichen Kosten" FROM dienstwagen
GROUP BY Fahrzeugtyp;`
- 7) Eine Liste mit den ermittelten durchschnittlichen Monatskosten für jeden Fahrzeugtyp.
`SELECT Fahrzeugtyp, AVG(KostenproMonat) AS "durchschnittliche Monatskosten" FROM dienstwagen
GROUP BY Fahrzeugtyp;`
`SELECT Fahrzeugtyp, ROUND(AVG(KostenproMonat),2) AS "durchschnittliche Monatskosten" FROM
dienstwagen GROUP BY Fahrzeugtyp;`
- 8) Eine Auflistung der monatlichen und jährlichen Kosten für jeden Fahrzeugtyp.
`SELECT Fahrzeugtyp, SUM(KostenproMonat) AS "Monatskosten", SUM(KostenproMonat * 12) AS
"Jahreskosten" FROM dienstwagen GROUP BY Fahrzeugtyp;`
- 9) Eine Auflistung der monatlichen und jährlichen Kosten für jedes Fahrzeugmodell.
`SELECT Fahrzeugtyp, Fahrzeugmodell, SUM(KostenproMonat) AS "Monatskosten", SUM(KostenproMonat *
12) AS "Jahreskosten" FROM dienstwagen GROUP BY Fahrzeugmodell;`