
APV

Konzeption und Administration von IT-Systemen

AP – RAID – STORAGE – DATENSCHUTZ

AP

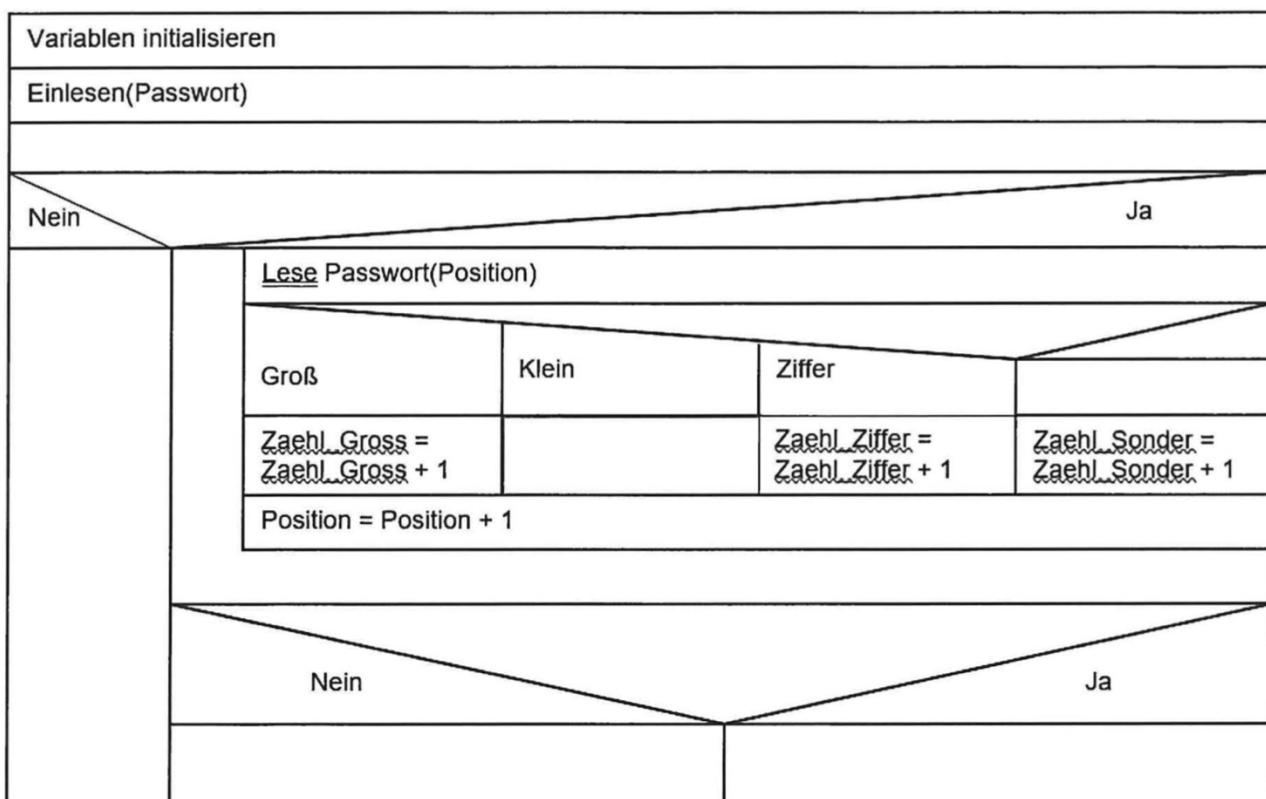
1.) Die Sicherheitsrichtlinie Ihres Unternehmens schreibt für Passwörter Folgendes vor:

- Ein Passwort muss aus mindestens zehn Zeichen bestehen.
- Ein Passwort muss den Komplexitätsbedingungen (Verwendung von Groß-, Kleinbuchstaben, Ziffern und Sonderzeichen) entsprechen.

Jedes neue Passwort soll daraufhin untersucht werden, ob es den oben angegebenen Richtlinien entspricht. Dazu wurde ein Struktogramm erstellt (siehe Seite 13), in dem einige Anweisungen fehlen.

Ergänzen Sie das Struktogramm um die fehlenden Anweisungen, indem Sie die folgenden Ziffern 1- 10 an den entsprechenden Stellen eintragen.

1. Länge von Passwort ≥ 10 ?
2. Länge von Passwort bestimmen
3. Zaehl_Klein = Zaehl_Klein + 1
4. Zaehl_Gross * Zaehl_Klein * Zaehl_Ziffer * Zaehl_Sonder > 0?
5. Zeichen= ?
6. Solange Position < Länge von Passwort
7. Sonder
8. Passwortlänge ungültig
9. Passwort unsicher
10. Passwort sicher



Notizen:

2.) Das Programm zur Überwachung der Arbeitsspeicher-Auslastung des Webservers speichert den maximalen Wert der Arbeitsspeicher-Auslastung einer jeden Stunde in einem eindimensionalen Array als ganzzahligen Prozentwert:

Darstellung der Werte im Array:

Stunde	0 bis 1	1 bis 2	2 bis 3	3 bis 4	4 bis 5	5 bis 6	20 bis 21	21 bis 22	22 bis 23	23 bis 0
Array-Index	0	1	2	3	4	5	20	21	22	23
Array-Wert	17	17	16	18	20	25	34	28	23	16

Eine Programmerweiterung soll den Mittelwert von diesen Werten berechnen. Überschreitet der Mittelwert 75 %, soll folgende Messagebox angezeigt werden:



Vervollständigen Sie den vorliegenden Programmentwurf entsprechend der Anforderung.

Orientieren Sie sich dabei an er [Anlage 1](#).

```
string message = "Wenig freier Speicher";           //Warnmeldung
string title = "Warnung";                           //Titel der Messagebox
int avgUsedRam = 0;                                //Mittelwert Tag
int sumTemp = 0;                                    //Summe der Stundenwerte

//Array mit Testdaten

int[] usedRAM = new int[24]
{17,17,16,18,20,25,33,44,40,52,60,56,33,44,40,52,60,56,33,44,34,28,23,16};

//Schleifenkopf

//Schleifenrumpf

//Berechnung Mittelwert

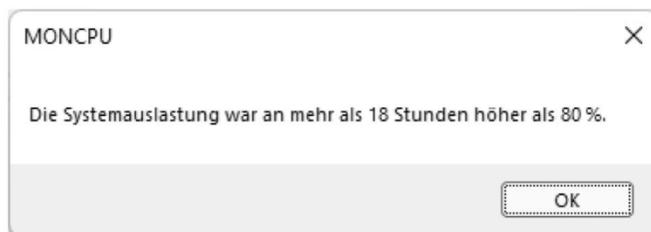
//Ausgabe Warnmeldung
```

3.) Im Rahmen der Serverüberwachung sollen Sie eine Programmerweiterung vornehmen.

Das Programm „MONCPU“ dient der täglichen Überwachung der CPU-Last des Datenbank-Servers. Es speichert für jede zurückliegende volle Stunde den Mittelwert der CPU-Last. Die Mittelwerte werden als ganzzahliger Prozentwert in einem eindimensionalen Array gespeichert.

Zeitraum	0 Uhr bis 1 Uhr	1 Uhr bis 2 Uhr	2 Uhr bis 3 Uhr	3 Uhr bis 4 Uhr	4 Uhr bis 5 Uhr	5 Uhr bis 6 Uhr	20 Uhr bis 21 Uhr	21 Uhr bis 22 Uhr	22 Uhr bis 23 Uhr	23 Uhr bis 0 Uhr
Array-Index	0	1	2	3	4	5	20	21	22	23
Array-Wert	33	44	40	52	60	56	40	52	60	56

Sie sollen das vorhandene Programm so erweitern, dass die folgende Meldung ausgegeben wird, sofern die CPU-Last am vergangenen Tag an mehr als 18 Stunden den Wert von 80 % überschritten hat.



Vervollständigen Sie den vorliegenden Programmentwurf entsprechend der Anforderung.

Orientieren Sie sich dabei an er Anlage 1.

```

string message = "Die Systemauslastung war an mehr als 18 Stunden höher als
80 %."                                     //Text Messagebox
string title = "MONCPU";                      //Titel Messagebox
int cpuartil = 0;                             //Systemauslastung
int cpulimit = 0;                            //Grenzwert der Systemauslastung
int sumstd = 0;                               //Summe Stunden
                                                //Array mit Testdaten

int[] usedCPU = new int[24]
{33,44,40,52,60,56,33,44,40,52,60,56,33,44,40,52,60,56,33,44,40,52,60,56};

```

4.) Aufgabe:

Für die Produktion von Wellpappen ist die vorhandene Datenbank zu erweitern. Die Firma hat sich für ein SQL-fähiges relationales Datenbanksystem entschieden, in der die nachfolgenden Bedingungen berücksichtigt werden sollen. Die Speicherung der Datenbank wird auf dem Hostrechner „Steuerungs-PC“ realisiert. In einer ersten Unterredung werden die zu speichernden Informationen definiert.

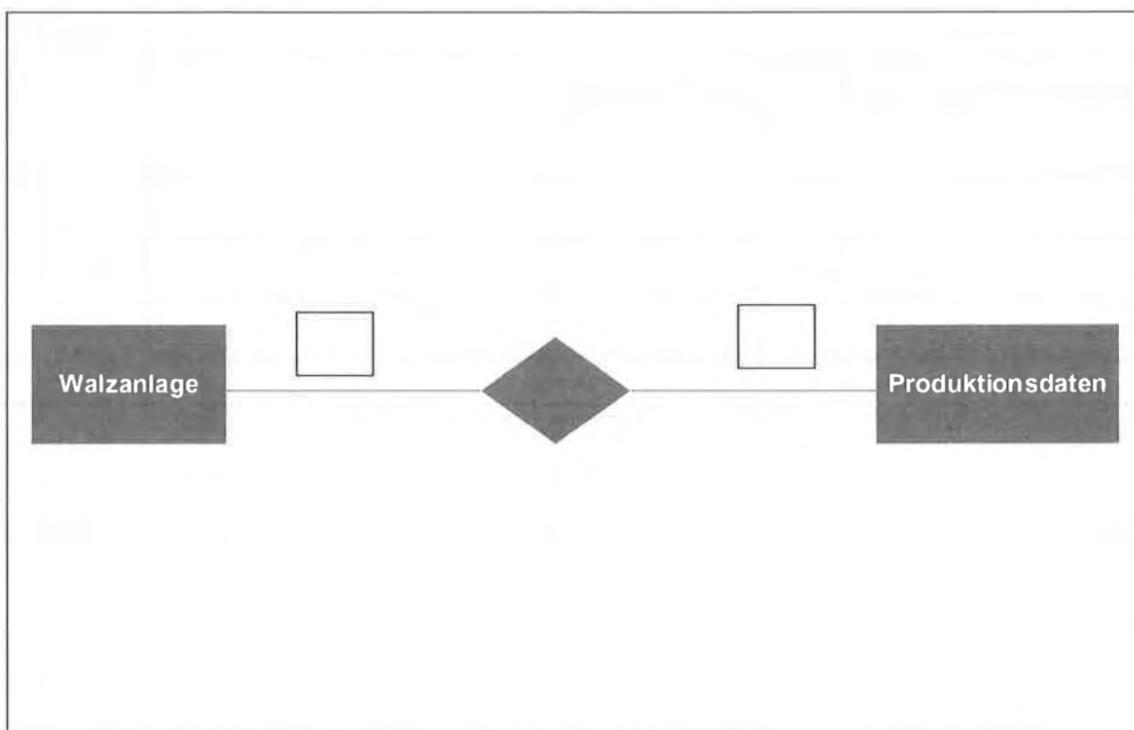
In dieser Datenbank sollen nur die Zusammenhänge zwischen den Walzanlagen, den Produktionsdaten abgebildet werden.

In der Produktionshalle sind mehrere Walzanlagen vorhanden. Diese jeweiligen Walzanlagen können Wellpappen mit unterschiedlichen Dicken (z. B. kleiner 4 mm, 4-8 mm, 8-12 mm) herstellen. In der Datenbank soll gespeichert werden, welche Walzanlage für welche Dicken (Spezifikation) verwendet werden kann. Außerdem soll das Baujahr, die Bezeichnung und eine eindeutige Maschinennummer gespeichert werden.

Für jede Walzanlage sollen die entsprechenden Produktionsdaten (Breite, Länge, Dicke und Anzahl) mit dem jeweiligen Zeitstempel abgespeichert werden.

Vervollständigen Sie das vorgegebene Entity-Relationship-Modell (kurz: ERM) für diese Datenbank mit allen erforderlichen Attributen und Kardinalitäten.

Hinweis: Die eventuell benötigten Fremdschlüsse müssen nicht in diesem Entwurf eingetragen werden. Die Kardinalität zwischen den beiden Tabellen soll auf die entsprechenden Beziehungslinien eingetragen werden.



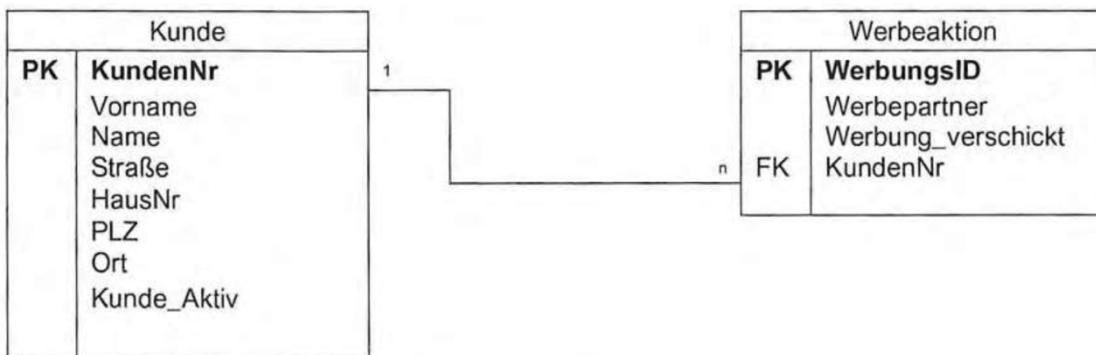
5.) Tragen Sie geeignete SQL-Datentypen ein:

	Musterwerte	SQL-Datentyp
WerbungsID	1	int
Werbepartner	Jubiläum	varchar(50)
Werbung_verschickt	01.01.2022	date
KundenNr	12345	
Kunde_Aktiv	true	
PLZ	01234	
Telefonnummer	0123/112233	
Erfasst am	11.05.2022	
IBAN	DE12123456780123456789	

6.) Erstellen Sie aus den folgenden Informationen ein ER-Modell, welches nur die Entitätstypen, die Beziehungen zwischen den Entitätstypen und die Kardinalitäten enthält:

- An einem Projekt arbeiten mehrere Mitarbeiter aus verschiedenen Fachbereichen und ein Mitarbeiter kann auch gleichzeitig in mehreren Projekten eingesetzt sein.
- Jeder Mitarbeiter kann genau einem Fachbereich zugeordnet werden.
- Jedes Projekt beinhaltet ein oder mehrere Lernfelder.
- Die Inhalte der Lernfelder können auch durch mehrere Projekte erstellt werden.
- Zu jedem Projekt werden mehrere Fachbücher zugeordnet. Dabei kann ein Fachbuch auch in verschiedenen Projekten Verwendung finden.

Sie werden mit Aufgaben im Zusammenhang mit dem Datenbanksystem beauftragt. Zur Speicherung der erfassten Kundendaten soll eine Datenbank zum Einsatz kommen. Es sind folgende Tabellen vorgesehen.



Orientieren Sie sich dabei an der [Anlage 3 \(SQL-Syntax\)](#).

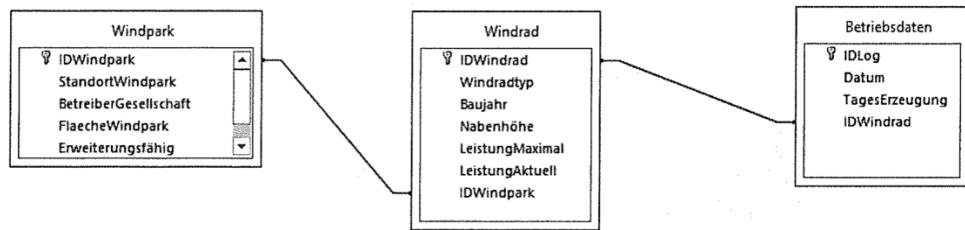
7.) Erstellen Sie eine Abfrage, um zu ermitteln, wie viele Kunden in Augsburg aktiv sind.

8.) Erstellen Sie einen SQL-Befehl, der die Tabelle *Werbeaktion* erstellt.

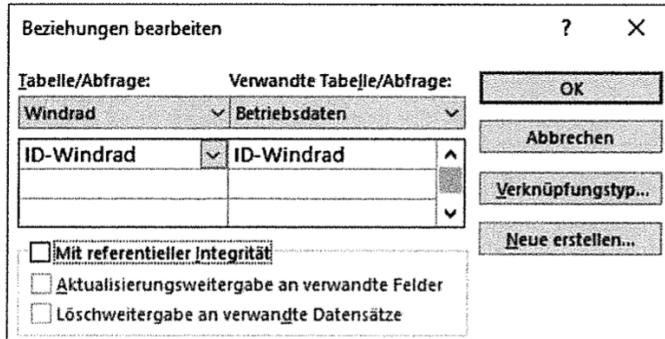
9.) Erstellen Sie eine SQL-Abfrage, um zu ermitteln, wie viele Werbungen vor dem 31.12.2021 an Kunden versendet wurden, deren PLZ mit der Ziffer 8 beginnt.

10.) Nennen Sie zwei nicht-relationale Datenbanken.

Folgender vereinfachter Datenbankentwurf liegt vor:



11.) Ergänzen Sie im Diagramm zu den beiden Beziehungen die Kardinalitäten.



12.) Erläutern Sie, was die referentielle Integrität ist und warum diese aktiviert werden sollte.

13.) Erstellen Sie eine SQL-Anweisung, die die Anzahl der Windräder im Windpark „Kahler Asten“ (ID Windpark 102) liefert.

14.) Erstellen Sie eine SQL-Anweisung, die die IDs der drei Windräder im Windpark „Kahler Asten“ (ID Windpark 102) mit der höchsten Tagesleistung am 03. Mai 2022 liefert.

15.) Zeichnen Sie ein *UML-Aktivitätsdiagramm* für ein CPS. Das *Aktivitätsdiagramm* soll den Prozess der Temperaturüberwachung in einem Gewächshaus grafisch darstellen:

- Im Programm werden die notwendigen Überwachungs- und Messparameter festgelegt.
- Die aktuelle Lufttemperatur wird durch einen Mess-Sensor gemessen und in der Variablen Temperatur gespeichert.
- Die Temperaturüberprüfung findet innerhalb einer (sich wiederholenden) Schleife statt.
- Wenn der Messwert (Temperatur) unter 30 Grad Celsius liegt oder diesem gleich, soll eine Textnachricht ausgegeben werden und danach eine Leuchtdiode (LED) angehen.
- Falls die Temperatur den Wert von 30 Grad Celsius übersteigt, wird die Methode `exit(0)` ausgeführt, um einen Programmabbruch (Prozessende) zu verursachen.

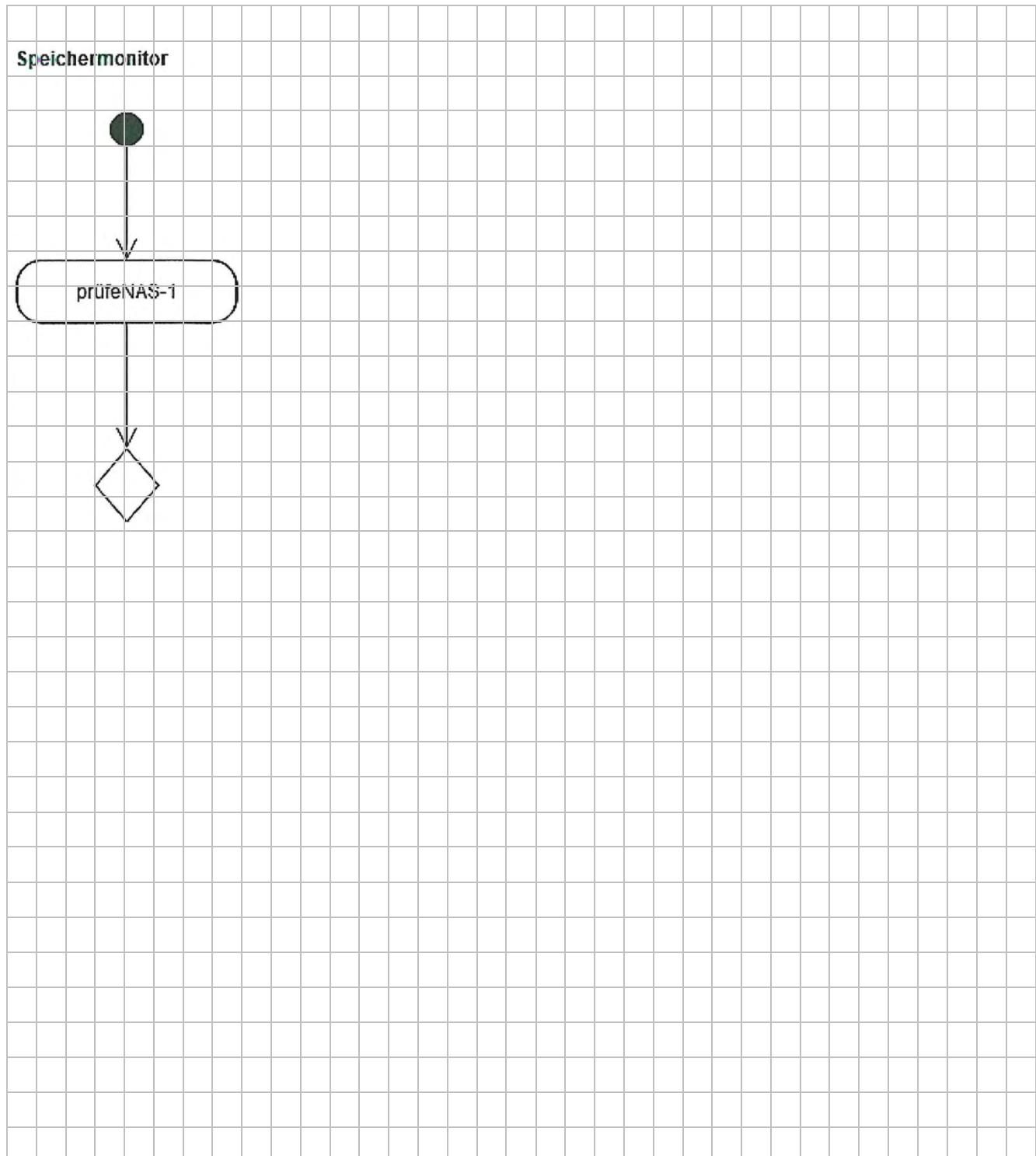
Orientieren Sie sich dabei an er [Anlage 2](#).



16.) Zeichnen Sie ein UML-Aktivitätsdiagramm für ein CPS. Das Aktivitätsdiagramm soll den Prozess der Systemverwaltung mit den folgenden Aufgaben darstellen:

Die Datenhaltung erfolgt auf den Speichersystemen NAS-1 und NAS-2. NAS-1 hat eine Kapazität von 10 TiB und NAS-2 hat 1 PiB. Damit das schnellere NAS-1 optimal genutzt wird, gelten die folgenden Regeln:

- Wenn die Speicherbelegung von NAS-1 größer als 40 % und kleiner als 60 % beträgt, dann sollen alle Dateien, auf die seit 120 Tagen nicht mehr zugegriffen wurde, von NAS-1 auf NAS-2 verschoben werden.
- Wenn die Speicherbelegung von NAS-1 größer als 60 % und kleiner als 80 % beträgt, dann sollen alle Dateien, auf die seit mindestens 60 Tagen nicht mehr zugegriffen wurde, von NAS-1 auf NAS-2 verschoben werden.
- Wenn die Speicherbelegung von NAS-1 mehr als 80 % beträgt, dann sollen alle Dateien, auf die seit sieben Tagen nicht mehr zugegriffen wurde und die größer als 100 MiB sind, von NAS-1 auf NAS-2 verschoben werden.



17.) Nennen Sie jeweils drei verschiedene UML-Diagramme für die statische und dynamische Sicht der Softwaremodellierung.

18.) Erstellen Sie ein UML-Klassendiagramms für die Schadensregulierung erstellt. Für die Versicherungsfälle KFZ und Immobilie sollen eigene Klassen entworfen werden.

Die Klasse KFZ hat die Eigenschaften:

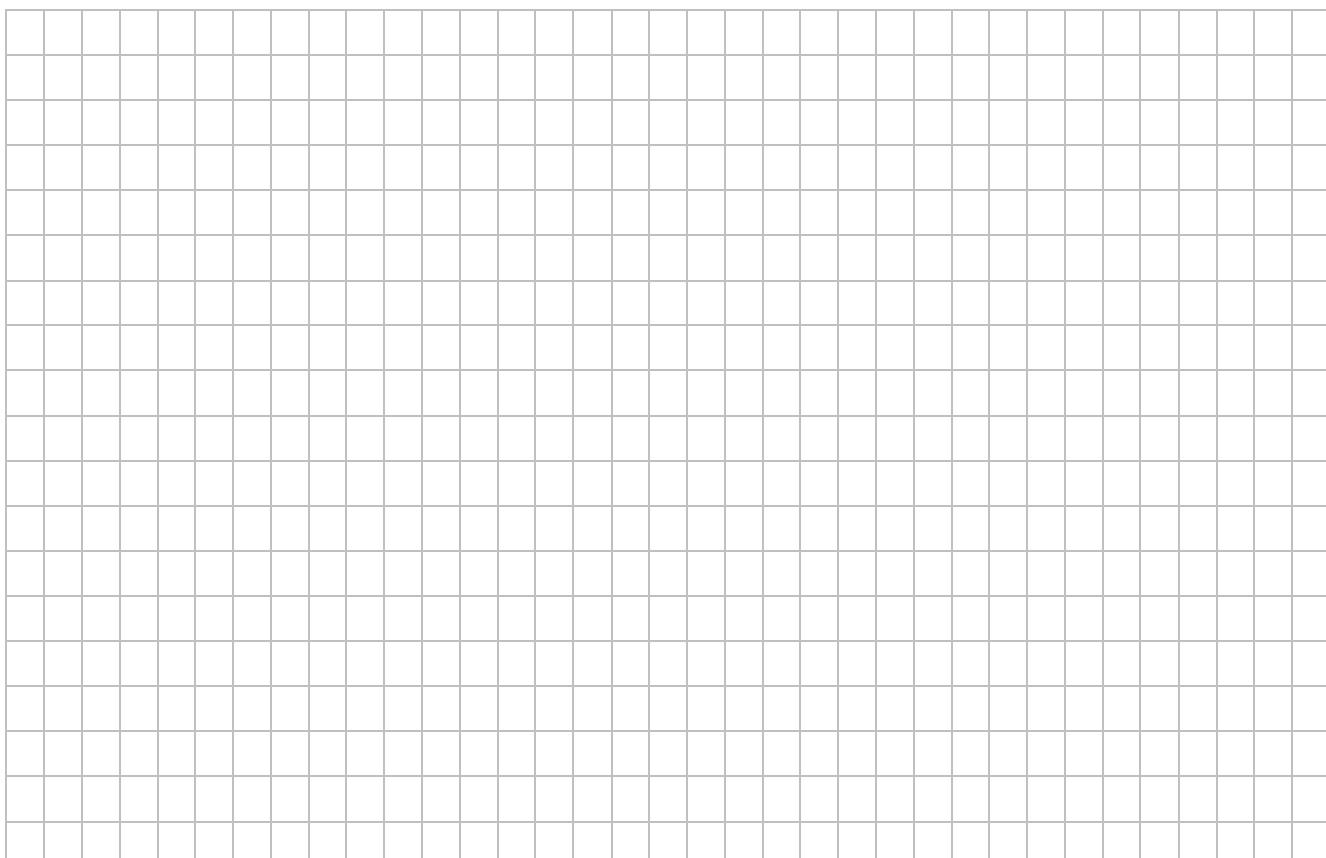
- Hersteller
 - Typschlüssel
 - Neupreis
 - Baujahr
 - Laufleistung
 - Schadenshöhe
- und Methoden:
- restwertBerechnen()
 - auszahlen()

Die Klasse Immobilie hat die Eigenschaften:

- Neupreis
 - Baujahr
 - Wohnfläche
 - Lagebewertung – Schadenshöhe
- und Methoden:
- restwertBerechnen()
 - getLagebewertung()
 - auszahlen()

Erstellen Sie eine sinnvolle Klassenhierarchie mit einer gemeinsamen Oberklasse „Versicherungsobjekt“.

Hinweis: Die Restwertberechnung erfolgt beim KFZ und bei der Immobilie unterschiedlich.
Zusätzliche Methoden, Konstruktoren und Zugriffsmodifikatoren sind nicht erforderlich.



19.) Nennen Sie zwei nicht-relationale Datenbanktypen.

20.) Erläutern Sie ausführlich die folgenden Konzepte der Softwareentwicklung.

Datenkapselung:

Vererbung:

Polymorphie:

Aggregation:

Komposition:

Kopplung:

RAID

21.) Ergänzen Sie die RAID-Tabelle.

RAID-Level	Linear	RAID-0	RAID-1	RAID-5	RAID-6	RAID-10	RAID-50	RAID-55
Min. Anzahl von Platten					4	4		9
wieviele Platten dürfen ausfallen bei min. Anzahl	0	0	1	1	2	1	1	3
wieviele Platten dürfen ausfallen bei n Platten	0	0	n-1	1	2	abhängig von der Anzahl der Untergruppen (legs)	abhängig von der Anzahl der Untergruppen (legs)	abhängig von der Anzahl der Untergruppen (legs)
Schreibleistung *	1	n	1	> n/4 *	> n/6 *	n/2 *	n *	n *
Leseleistung *	1	n	1	n-1 *	n-2 *	n/2 *	n *	n *
Nettokapazität in % bei min. Plattenanzahl								44 %
Nettokapazität in % bei 3 Platten					X	X	X	X
Nettokapazität in % bei 4 Platten							X	X
Nettokapazität in % bei 5 Platten						X	X	X
Nettokapazität in % bei 6 Platten						abhängig von der Anzahl der Untergruppen (legs)		X
Nettokapazität in % bei n Platten						abhängig von der Anzahl der Untergruppen (legs)	abhängig von der Anzahl der Untergruppen (legs)	abhängig von der Anzahl der Untergruppen (legs)

22.) Vervollständigen Sie die Übersichtstabelle um zwei weitere leistungsbestimmende Merkmale:

Das Speichersystem soll insgesamt aus zwei RAID-Verbünden bestehen. Zum einen aus einem möglichst schnellen RAID-Verbund mit einer Kapazität von ca. 20 TiB und zum anderen aus einem kostengünstigen RAID-Verbund mit einer Kapazität von ca. 200 TiB. Bei der Auswahl der RAID-Controller sind verschiedene technische Merkmale der Controller zu berücksichtigen.

Technisches Merkmal	Möglicher Wert
Unterstützte RAID-Level	0, 1, 10, 5, 6, 50, 60

23.) Geben Sie den fehlertoleranten RAID-Level **an**, der die höchstmögliche Datenrate bei Schreiboperationen bietet.

24.) Geben Sie den fehlertoleranten RAID-Level **an**, bei dem die verfügbare Speicherkapazität gegenüber der gesamten Speicherkapazität des RAID-Verbunds möglichst groß ist.

Als Datenspeicherlösung stehen zwei Storage Arrays mit je maximal acht Festplatten zu je 2 TiB zur Verfügung. Die Datenbank ist 5 TiB groß. Zusätzlich werden 3 TiB für Nutzdaten und 5 TiB als Puffer benötigt. Um die geforderte Verfügbarkeit sicherzustellen, entscheiden Sie sich unter anderem zur Nutzung von **RAID 6** mit Hot-Spare-Festplatte.

25.) Berechnen Sie, wie viele Festplatten benötigt werden, um den Speicherbedarf zu decken und die Architekturanforderungen zu erfüllen.

Einige Zulieferer erstellen weiterhin Lieferscheine, Angebote und Rechnungen in Papierform. Diese sollen digital archiviert werden. Pro Monat werden 30 GiB Dokumentendaten erzeugt. Diese werden mit einer Komprimierungsrate von 60 % gepackt und auf ein Archivsystem kopiert, auf dem 1 TiB Speicher für die Daten reserviert ist. Dort sollen Sie für zehn Jahre gespeichert werden.

26.) Berechnen Sie den Speicherplatz, der für die Speicherung der Daten für zehn Jahre benötigt wird.

27.) Berechnen Sie, wie lange der reservierte Speicher reicht. Runden Sie auf ganze Jahre ab.

28.) Zusätzlich soll im SAN ein RAID-6-Verbund mit einer Hot-Spare-Festplatte betrieben werden.
Erläutern Sie die Funktion einer Hot-Spare-Festplatte.

STORAGE

29.) Erläutern Sie die genannten Backupverfahren unter Berücksichtigung des benötigten Speicher-volumens und den Auswirkungen auf das Archivbit.

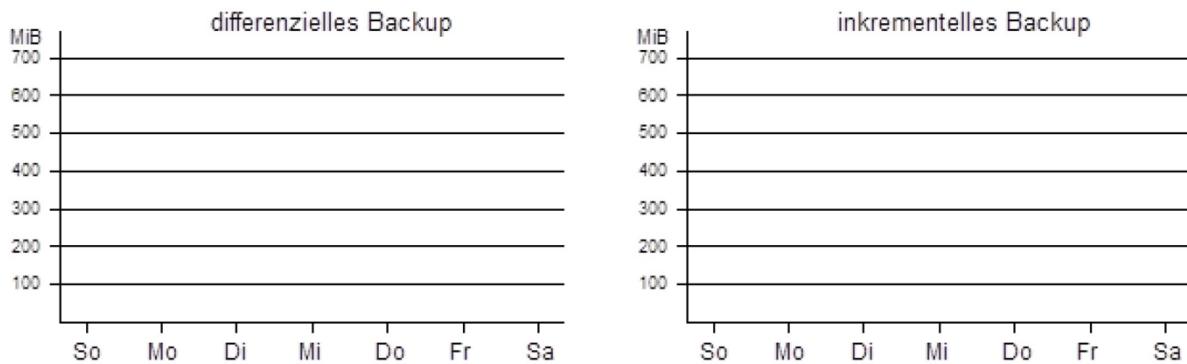
Backupverfahren	Erläuterung
Vollbackup	
Differenzielles Backup	
Inkrementelles Backup	
Klonen	

Die Unternehmensdaten befinden sich in einer Cloud. An Werktagen kommen neue Daten hinzu oder werden geändert. Am Sonntag wird ein Vollbackup durchgeführt - von Montag bis Freitag wird ein differenzielles sowie ein inkrementelles Backup erstellt. Die Daten werden auf ein SCSI-Bandlaufwerk mit 90 GiB Speicherkapazität und 12 MiB/s Schreibgeschwindigkeit gesichert. Das gesamte Datenvolumen auf dem Fileserver beträgt zurzeit 10,5 GiB. Täglich werden 80 MiB Daten erstellt und 100 MiB Daten geändert.

30.) Ermitteln Sie für ein 1.) differenzielles oder 2.) inkrementelles Backup die entsprechende Datenmenge sowie die wöchentliche Sicherungszeit. Verwenden sie dazu die folgende Tabelle.

Wochentag	1.) Datenmenge	1.) Dauer in Sekunden	2.) Datenmenge	2.) Dauer in Sekunden
Sonntag				
Montag				
Dienstag				
Mittwoch				
Donnerstag				
Freitag				
Zeit pro Woche				

31.) Stellen Sie die täglich anfallenden Datenmengen als Balkendiagramme **dar** (Vollbackup nur andeuten).



32.) Im Rahmen der Firmendaten-Sicherheit soll eine neue Backup-Lösung implementiert werden. Es wird diskutiert, ob das „Backup as a Service“ (BaaS) eine geeignete Lösung darstellt.

33.) Erläutern Sie kurz die folgenden drei Begriffe: SaaS, PaaS und IaaS.

Im Rahmen der Neustrukturierung der IT-Dienste sollen verschiedene Anwendungen in die Private Cloud migriert und als Software as a Service (SaaS) bereitgestellt werden.

34.) Erläutern Sie drei Aspekte, die für eine Bereitstellung von Anwendungen in Form von SaaS sprechen.

Im Zuge der Digitalisierungsinitiative ein neuer Datenbankserver aufgebaut werden, da mit einem erhöhten Datenaufkommen zu rechnen ist. Sie werden mit der Planung und Durchführung der Migration beauftragt.

35.) Nennen Sie je 2 Vorteile für ein Hosting On Premises und die Nutzung einer Public oder Private Cloud.

In den Vertragsbedingungen Ihres Cloud-Providers steht unter dem Punkt „Abrechnung“ der Begriff „Nutzungsgerechte Abrechnung (Measured Service)“.

36.) Erläutern Sie anhand eines Beispiels, was unter „Nutzungsgerechte Abrechnung“ zu verstehen ist.

37.) Erläutern Sie, inwiefern durch eine Cloud-Lösung folgende Vorteile erzielt werden können: Zeit-einsparung, Skalierbarkeit, Kostenreduktion.

Von den IT-Systemen wird ein hohes Maß an Zuverlässigkeit (Reliability) und Verfügbarkeit (Availability) gefordert.

38.) Erläutern Sie am Beispiel eines Kassensystems je eine Anforderung der beiden Größen.

DATENSCHUTZ

39.) Nennen Sie drei Kriterien, die aus Datenschutzsicht bei der Speicherung von Daten grundsätzlich zu beachten sind.

40.) Beschreiben Sie die Anonymisierung und die Pseudonymisierung.

41.) Erläutern Sie die drei folgenden Schutzziele der Informationssicherheit (Vertraulichkeit, Integrität und Verfügbarkeit).

Für die Videoüberwachung des Firmengeländes werden folgende vier verglichen:

Nr.	Eigenschaften	Neocam BVR	CamHD-Sec	SecretAV-NB	XoCam z1
1	Bildauflösung	1.920 x 1.080	HD	FHD	4K
2	Bilder pro Sekunde	30	60	30	30
3	Bewegungserkennung	ja	ja	nein	nein
4	PoE	nein	ja	nein	nein
5	SD-Kartenslot	bis 8 GiB	nein	bis 8 GiB	bis 4 GiB
6	Schutz vor Vandalismus	ja	nein	nein	nein
7	Zugriff passwortgeschützt	ja	ja	ja	nein
8	Verschlüsselungsmethode	WPA2	WPA3	WEP	WPA
9	Nachtsichtmodus	Infrarot	nein	nein	nein
10	Video-Komprimierung	H.264, MPEG-4, MJPEG	MPEG-4, MJPEG	H.264, MPEG-4, MJPEG	H.264
11	WLAN	ja	ja	ja	ja

42.) Geben Sie zu dem jeweiligen aufgelisteten Einsatzzweck die geeignete Webcam aus der oben-stehenden Tabelle **an**. **Begründen** Sie Ihre Entscheidung.

Einsatzzweck	Webcam	Begründung
24-stündige Überwachung des Außengeländes mit Publikumsverkehr		
Webkonferenzen bei höchster Bildqualität		
Überwachung des Serverraums bei sicherster Verschlüsselungsmethode		

43.) Ermitteln Sie anhand der folgenden Angaben die anfallende Datenmenge der zu speichernden Bilder pro Webcam pro Minute in MiB. Der Rechenweg ist anzugeben. Das Ergebnis ist auf eine ganze Zahl aufzurunden.

Überwachungszeit pro Tag: 00:00 bis 05:00 Uhr und 19:00 bis 24:00 Uhr

Anzahl der Webcams: 10

Bildauflösung pro Bild: 1.920 x 1.080 Pixel

Farbtiefe: 16 Bit

Videoübertragung: 10 Bilder/Sekunde

44.) Definieren Sie in diesem Zusammenhang die 3 Begriffe: Löschfristen, Persönlichkeitsrechte und Zweckbindung.

Anlage 1

Beispiele für verschiedene Schleifenarten:

```
while(number < 5)
{
    Console.WriteLine(number);
    number = number + 1
}

for(int i = 0; i < number; i++)
{
    Console.WriteLine(number);
}

do
{
    Console.WriteLine(number);
    number = number + 1;
} while(number < 5);
```

Beispiel für eine Auswahlanweisung:

```
int number = 20;
if (number < 18)
{
    Console.WriteLine("ok");
}
else
{
    Console.WriteLine("not ok");
}
```

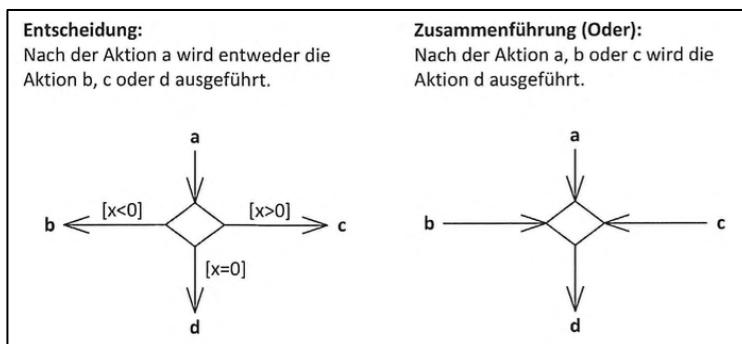
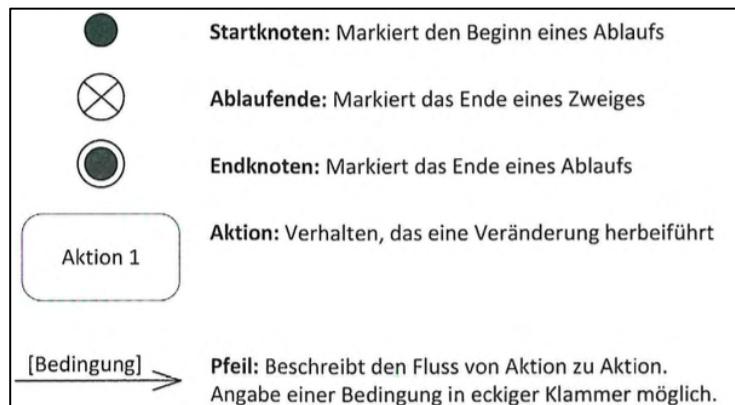
Kommentare

```
//Kommentar einzeilig
/* Kommentar mehrzeilig */
```

Beispiel Messagebox

```
string message = "Anzeigetext in der Messagebox";
string title = "Bezeichnung der Messagebox";
MessageBox.Show(message,title);
```

Anlage 2



Anlage 3

Syntax	Beschreibung
Tabelle	
CREATE TABLE Tabellenname(Spaltenname < DATENTYP >, Primärschlüssel, Fremdschlüssel)	Erzeugt eine neue leere Tabelle mit der beschriebenen Struktur
ALTER TABLE Tabellenname ADD COLUMN Spaltenname Datentyp DROP COLUMN Spaltenname Datentyp ADD FOREIGN KEY (Spaltenname) REFERENCES Tabellenname(Primärschlüsselspaltenname)	Änderungen an einer Tabelle: Hinzufügen einer Spalte Entfernen einer Spalte Definiert eine Spalte als Fremdschlüssel
CHARACTER	Textdatentyp
DECIMAL	Numerischer Datentyp (Festkommazahl)
DOUBLE	Numerischer Datentyp (Doppelte Präzision)
INTEGER	Numerischer Datentyp (Ganzzahl)
DATE	Datum (Format DD.MM.YYYY)
PRIMARY KEY (Spaltenname)	Erstellung eines Primärschlüssels
FOREIGN KEY (Spaltenname) REFERENCES Tabellenname(Primärschlüsselspaltenname)	Erstellung einer Fremdschlüssel-Beziehung
DROP TABLE Tabellenname	Löscht eine Tabelle
Befehle, Klauseln, Attribute	
SELECT * Spaltenname1 [, Spaltenname2, ...]	Wählt die Spalten einer oder mehrerer Tabellen, deren Inhalte in die Liste aufgenommen werden sollen; alle Spalten (*) oder die namentlich aufgeführten
FROM	Name der Tabelle oder Namen der Tabellen, aus denen die Daten der Ausgabe stammen sollen
SELECT ... FROM ... (SELECT ... FROM ... WHERE ...) AS tbl WHERE ...	Unterabfrage (subquery), die in eine äußere Abfrage eingebettet ist. Das Ergebnis der Unterabfrage wird wie eine Tabelle – hier mit Namen "tbl" – behandelt.
SELECT DISTINCT	Eliminiert Redundanzen, die in einer Tabellen auftreten können, Werte werden jeweils nur einmal angezeigt.
JOIN / INNER JOIN	Liefert nur die Datensätze zweier Tabellen, die gleiche Datenwerte enthalten
LEFT JOIN / LEFT OUTER JOIN	Liefert von der erstgenannten (linken) Tabelle alle Datensätze und von der zweiten Tabelle jene, deren Datenwerte mit denen der ersten Tabelle übereinstimmen
RIGHT JOIN / RIGHT OUTER JOIN	Liefert von der zweiten (rechten) Tabelle alle Datensätze und von der ersten Tabelle jene, deren Datenwerte mit denen der zweiten Tabelle übereinstimmen
WHERE	Bedingung, nach der Datensätze ausgewählt werden sollen
WHERE EXISTS (subquery)	Die Bedingungen EXISTS prüft, ob die Suchbedingung einer Unterabfrage mindestens eine Zeile zurück liefert. NOT EXIST negiert die Bedingung.
WHERE NOT EXISTS (subquery)	
WHERE ... IN (subquery)	Der Wert des Datenfelds ist in der auswählten Menge vorhanden.
WHERE NOT... IN (subquery)	Der Wert des Datenfelds ist in der auswählten Menge nicht vorhanden.
GROUP BY Spaltenname1 [,Spaltenname2, ...]	Gruppierung (Aggregation) nach Inhalt des genannten Feldes
ORDER BY Spaltenname1 [,Spaltenname2, ...] ASC DESC	Sortierung nach Inhalt des genannten Feldes oder der genannten Felder ASC: aufsteigend; DESC: absteigend

Syntax	Beschreibung
Datenmanipulation	
DELETE FROM Tabellenname	Löschen von Datensätzen in der genannten Tabelle
UPDATE Tabellenname SET	Aktualisiert Daten in Feldern einer Tabelle
INSERT INTO Tabellenname[(spalte1, spalte2, ...)] VALUES (Wert für Spalte 1 [, Wert für Spalte 2, ...]) oder SELECT ... FROM ... WHERE	Fügt Datensätze in die genannte Tabelle, die entweder mit festen Werten belegt oder Ergebnis eines SELECT-Befehls sind
Berechtigungen kontrollieren	
CREATE Benutzer Rolle IDENTIFIED BY 'Passwort'	Erzeugt einen neuen Benutzer oder eine neue Rolle mit einem Passwort
GRANT Recht Rolle ON *.* Datenbank.* Datenbank.Objekt	Weist einem Benutzer oder einer Rolle ein Recht auf ein bestimmtes Datenbank-Objekt zu
TO Benutzer Rolle [WITH GRANT OPTION]	Weist einem Benutzer eine Rolle zu
REVOKE Rechte Rollen ON *.* Datenbank.* Datenbank.Objekt	Entzieht einem Benutzer oder einer Rolle ein Recht auf ein bestimmtes Datenbank-Objekt
FROM Benutzer Rolle	Entzieht einem Benutzer eine Rolle
Aggregatfunktionen	
AVG(Spaltenname)	Ermittelt das arithmetische Mittel aller Werte im angegebenen Feld
COUNT(Spaltenname *)	Ermittelt die Anzahl der Datensätze mit Nicht-NULL-Werten im angegebenen Feld oder alle Datensätze der Tabelle (dann mit Operator *)
SUM(Spaltenname Formel)	Ermittelt die Summe aller Werte im angegebenen Feld oder der Formelergebnisse
MIN(Spaltenname Formel)	Ermittelt den kleinsten aller Werte im angegebenen Feld
MAX (Spaltenname Formel)	Ermittelt den größten aller Werte im angegebenen Feld
Funktionen	
LEFT(Zeichenkette, Anzahlzeichen)	Liefert Anzahlzeichen der Zeichenkette von links.
RIGHT(Zeichenkette, Anzahlzeichen)	Liefert Anzahlzeichen der Zeichenkette von rechts.
CURRENT	Liefert das aktuelle Datum mit der aktuellen Uhrzeit
CONVERT(time,[DatumZeit])	Liefert die Uhrzeit aus einer DatumZeit-Angabe
DATE(Wert)	Wandelt einen Wert in ein Datum um
DAY(Datum)	Liefert den Tag des Monats aus dem angegebenen Datum
MONTH(Datum)	Liefert den Monat aus dem angegebenen Datum
TODAY	Liefert das aktuelle Datum
WEEKDAY(Datum)	Liefert den Tag der Woche aus dem angegebenen Datum
YEAR(Datum)	Liefert das Jahr aus dem angegebenen Datum
DATEADD(Datumsteil, Intervall, Datum)	Fügt einem Datum ein Intervall (ausgedrückt in den unter Datumsteil angegebenen Einheiten) hinzu
DATEDIFF(Datumsteil, Anfangsdatum, Enddatum) Datumsteile: DAY, MONTH, YEAR	Liefert Enddatum-Startdatum (ausgedrückt in den unter Datumsteil angegebenen Einheiten)
Operatoren	
AND	Logisches UND
LIKE	Überprüfung von Text auf Gleichheit wenn Platzhalter ("regular expressions") eingesetzt werden.
NOT	Logische Negation
OR	Logisches ODER
IS NULL	Überprüfung auf NULL
=	Test auf Gleichheit
>, >=, <, <=, < >	Test auf Ungleichheit
*	Multiplikation
/	Division
+	Addition, positives Vorzeichen
-	Subtraktion, negatives Vorzeichen