

Klassenarbeit

(Variante F)

66/70 90% 1-
26.5.24

Name: Flus Hendricks
Datum: 6.6.2024
Klasse: FISI-32

Aufgabe 1

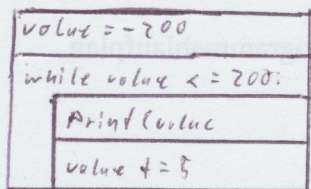
NA: 21

- a) Schreiben Sie den Python-Code für eine Schleife, die in 5-er Schritten von -200 bis + 200 jeweils einschließlich zählt und die jeweilige Zahl ausgibt.

```
value = -200
while value <= 200:
    print(value)
    value += 5
```

4

- b) Visualisieren Sie die Situation aus (a) mit einem Struktogramm.



4

- c) Eine Liste namens `namensliste` enthält fünf unterschiedliche Namen Ihrer Wahl. Anschließend werden der Liste zwei weitere Namen hinzugefügt und die Liste wird sortiert. Schreiben Sie den entsprechenden Python-Code.

```
namensliste = ["Samuel", "Anna", "Mannuel", "Jürglen", "Michaela"]
namensliste.append("Lena", "Hinna")
```

|||
7

2

- d) Schreiben Sie ein Python-Skript, welches den Nettopreis eines Produkts einliest und die Umsatzsteuer und den Bruttopreis des Produkts ausgibt. Der Umsatzsteuersatz von 19% soll als Konstante festgelegt werden. Umsatzsteuer und Bruttopreis sollen auf zwei Nachkommastellen gerundet sein.

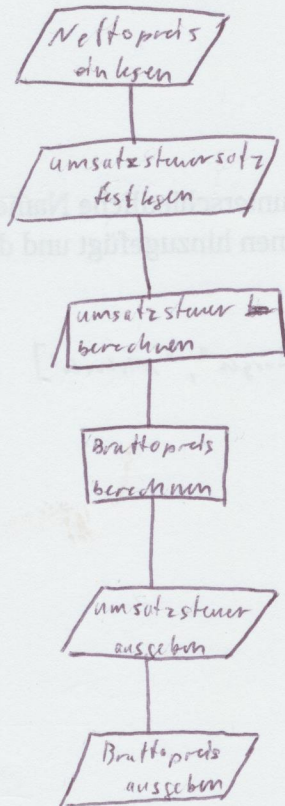
```

input Floatl
nettopreis = input("Nettopreis eingeben: ")
umsatzsteuersatz = 0.19

umsatzsteuer = round(nettopreis * umsatzsteuersatz, 2)
bruttopreis = round(nettopreis + umsatzsteuer, 2)

print("Umsatzsteuer:", umsatzsteuer)
print("Bruttopreis:", bruttopreis)
    
```

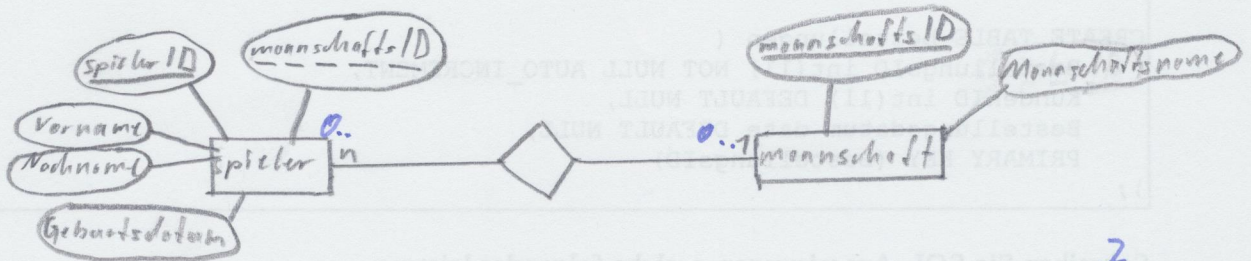
- e) Visualisieren Sie das Skript aus Aufgabe (d) mit einem Programmablaufplan.



Aufgabe 2

B2: 13

- a) Stellen Sie den folgenden Sachverhalt als Entity-Relationship-Diagramm mit Attributen dar (Chen-Notation). dar. Eine Mannschaft besteht aus Spielern. Spieler sind Personen. Eine Mannschaft kann aus mehreren Spielern bestehen. Ein Spieler kann Teil von maximal einer Mannschaften sein. Es gibt Spieler die noch keiner Mannschaft zugeordnet sind. Es gibt Mannschaften, die noch keine Spieler haben. Für jeden Spieler werden Nachname, Vorname, Geburtsdatum gespeichert. Jede Mannschaft besitzt einen Mannschaftsnamen. Jede Entity soll zusätzlich einen Identifier als Primärschlüssel besitzen.



- b) Erstellen Sie ein Relationen-Schemata für die in (a) beschriebene Situation.

~~Spieler =~~

Spieler (Spieler ID, mannschafts ID, Vorname, Nachname, Geburtsdatum)

Mannschaft (mannschafts ID, Mannschaftsname)

Aufgabe 3

A3:14

Durch die folgenden SQL-Code wird eine Datenbank angelegt.

```
CREATE DATABASE beispiel_ls4_joins DEFAULT CHARACTER SET utf8;

CREATE TABLE kunden (
  KundenID int(11) NOT NULL AUTO_INCREMENT,
  KundenName varchar(255) DEFAULT NULL,
  PRIMARY KEY (KundenID)
);

CREATE TABLE bestellungen (
  BestellungsID int(11) NOT NULL AUTO_INCREMENT,
  KundenID int(11) DEFAULT NULL,
  Bestelldatum date DEFAULT NULL,
  PRIMARY KEY (BestellungsID)
);
```

Schreiben Sie SQL-Anweisungen, welche folgendes leisten:
(SQL Schlüsselwörter sind in Großbuchstaben zu schreiben)

- a) Alle Daten aller kunden ausgeben.

`SELECT * FROM kunden;`

2

- b) Einen neuen Kunden namens Hannibal Lector mit der KundenID 111 einfügen.

`INSERT INTO kunden (KundenID, KundenName) VALUES (111, "Hannibal Lector");`

2

- c) Die Anzahl der Bestellungen des Kunden mit der KundenID 55 ausgeben.

`SELECT COUNT(*) FROM bestellungen WHERE KundenID = 55;`
(kundenID LIKE 55)

4

- d) Alle Bestellungen des Kunden mit der KundenID 55 löschen.

`DELETE * FROM bestellungen WHERE kundenID = 55;`

3

- e) Alle Bestelungsdaten und den Kundennamen des Kunden mit der KundenID 111 ausgeben.

`SELECT bestellungen.BestellungsID, bestellungen.Bestelldatum, kunden.KundenName FROM bestellungen
LEFT JOIN kunden ON bestellungen.kundenID = kunden.kundenID`

3

verbesserte
version
↳

`SELECT bestellungen.BestellungsID, bestellungen.Bestelldatum, bestellungen.kundenID,
bestellungen.kundenName
FROM bestellungen LEFT JOIN kunden ON bestellungen.kundenID = kunden.kundenID`

