

Name:	Studiengang: <input type="checkbox"/> B.A.   <input type="checkbox"/> M.A.
Vorname:	Matrikelnummer:
Studiengänge:	Fachsemester:

**Allgemeine Hinweise:**

1. Überprüfen Sie bitte, ob Sie alle Seiten der Klausurangabe vollständig erhalten haben  
(Gesamtzahl: 9)
2. **Bearbeitungszeit: 90 Minuten**, maximal erreichbare **Punktzahl: 77**. Die jeweils erreichbare Punktzahl ist bei jeder Frage angegeben. Bitte teilen Sie Ihre Arbeitszeit entsprechend ein.
3. Denken Sie daran, die Daten oben einzutragen, **bevor** Sie mit der Bearbeitung beginnen.
4. Verwenden Sie für die Beantwortung aller Fragen diese Klausurangabe. Sie können jederzeit auch die Rückseiten beschreiben, falls der Platz auf der Vorderseite nicht ausreichen sollte.  
Bitte geben Sie in jedem Fall an, auf welche Frage sich die Lösung jeweils bezieht. Bei Multiple-Choice-Fragen treffen Sie bitte die Auswahl Ihrer Antworten ebenfalls auf der Klausurangabe.
5. Benutzen Sie keine Bleistifte, keine rot schreibenden Stifte und kein TippEx, o.ä.
6. Zugelassene Hilfsmittel: **KEINE**
7. Mobiltelefone sowie Computer am Arbeitsplatz - auch ausgeschaltet - sind **nicht zugelassen**.
8. Bitte legen Sie Lichtbildausweis und Studierendenausweis gut sichtbar vor sich, damit Ihre Identität möglichst störungsfrei überprüft werden kann.
9. Geben Sie keine mehrdeutigen (oder mehrere) Lösungen an. In solchen Fällen wird stets die Lösung mit der geringeren Punktzahl gewertet. Eine richtige und eine falsche Lösung ergeben also 0 Punkte.
10. Wenden Sie sich bei Unklarheiten in den Aufgabenstellungen immer an die Aufsichtsführenden. Hinweise und Hilfestellungen werden dann, falls erforderlich, offiziell für alle Teilnehmer durchgegeben.

**Aufgabe 1 - Theoretische Grundlagen in Python** **(16 Punkte)****Wahr oder Falsch?**

Beurteilen Sie für jede der folgenden Aussagen, ob diese Wahr oder Falsch ist. Bitte beachten Sie:  
Falsch angekreuzte Antworten führen zu Punktabzug, wobei eine Gesamtpunktzahl kleiner als Null  
nicht erreicht werden kann.

<b>Wahr</b>	<b>Falsch</b>
<input type="checkbox"/>	<input type="checkbox"/>
Variablennamen in Python dürfen mit einer Zahl beginnen.	
<input type="checkbox"/>	<input type="checkbox"/>
Mit <b>items()</b> lassen sich Dictionaries mit gleichzeitigem Zugriff auf key und value iterieren.	
<input type="checkbox"/>	<input type="checkbox"/>
Die Typ-Prüfung erfolgt zur Laufzeit des Programms.	
<input type="checkbox"/>	<input type="checkbox"/>
Funktionen müssen immer einen Rückgabewert haben.	
<input type="checkbox"/>	<input type="checkbox"/>
Tupel sind veränderlich (mutable), Listen sind unveränderlich (immutable).	
<input type="checkbox"/>	<input type="checkbox"/>
Mittels <b>**</b> können Listen bei der Übergabe an Funktionen entpackt werden, um ihre Werte als einzelne Parameter an die Funktion zu übergeben.	
<input type="checkbox"/>	<input type="checkbox"/>
while-Schleifen in List Comprehensions sind möglich.	
<input type="checkbox"/>	<input type="checkbox"/>
Über den Konstruktor lässt sich definieren, welche Parameter beim Anlegen einer Instanz erwartet werden.	

**Aufgabe 2 - Fehlersuche** **(10 Punkte)**

Im unten abgebildeten Codeausschnitt haben sich **fünf Fehler** eingeschlichen. Nutzen Sie Ihre Python-Kenntnisse, um die Fehler zu finden, sodass der Code im Anschluss fehlerfrei durchläuft.

```
def remove_number_from_string(number='0', my_string);  
    try:  
        my_string = my_string.replace(number, '')  
    else:  
        print('an error has occurred')  
    return my_string  
  
my_string='0test'  
my_string = remove_numbers_from_string(my_string=my_string)  
print(my_string)  
  
for char in my_string:  
    if char in ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9']:  
        print(f'char {char} in {my_string}')
```

Markieren Sie die Fehler im Code und erläutern Sie kurz, was der Fehler ist und was geändert werden müsste, um den Fehler zu beheben.

**Aufgabe 3 - Datenverarbeitung mit pandas****(10 Punkte)**Gegeben ist folgender DataFrame `regensburg_bars_df`:

	name	rating	year	isDancePub
0	Heimat	5.0	1980	True
1	Flannigans	4.0	1990	False
2	Astra Kiez-Bar	4.5	2022	False
3	Blomberg	2.0	2000	True
4	Banane	3.0	1970	False

Schreiben Sie Code für jede der folgenden Teilaufgaben.

1. Iterieren Sie durch den Datensatz und geben Sie jede Zeile wie folgt formatiert auf die Konsole aus: "Die Bar <name> hat eine Bewertung von <rating> und wurde im Jahr <year> eröffnet."
2. Wählen Sie die Bars aus, in denen getanzt werden kann (`isDancePub`). Gewünschtes Ergebnis:

	name	rating	year	isDancePub
0	Heimat	5.0	1980	True
3	Blomberg	2.0	2000	True

3. Wählen Sie die ersten drei Einträge der Spalte **name** aus.

4. Wie werden Regensburger Bars im Schnitt bewertet? Berechnen Sie den Mittelwert (**mean**) der Ratings.

#### Aufgabe 4 - Visualisieren mit matplotlib (8 Punkte)

Der DataFrame **regensburg\_bars\_df** aus Aufgabe 3 wird mit folgendem Code visualisiert:

```
import matplotlib
import matplotlib.pyplot as plt

fig, ax = plt.subplots(figsize=(10, 5))
ax.boxplot(regensburg_bars_df['rating'],
            showmeans=True,
            meanline=True,
            meanprops={"color": "red", "linewidth": 1},
            )
ax.set_title('Regensburger Bar Ratings')
```

Beurteilen Sie für jede der folgenden Aussagen, ob diese Wahr oder Falsch ist. Falsch angekreuzte Antworten führen zu Punktabzug, wobei eine Punktzahl kleiner als Null nicht erreicht werden kann.

**Wahr**

**Falsch**

Wenn der Code ausgeführt wird, erhält man einen Boxplot mit dem Titel  
**“Regensburger Bar Ratings”**.

Der Boxplot enthält eine Linie für den Median und eine für das arithmetische  
Mittel.

Die Grafik hat eine Breite von 10 und eine Höhe von 5.

Der Boxplot besitzt eine rote Füllung.

**Aufgabe 5 - Funktionen schreiben****(10 Punkte)**

Schreiben Sie eine Funktion **calc\_odd\_numbers\_mean**, die als Parameter eine Liste mit Zahlen (integer) entgegennimmt. Die Funktion soll den **Mittelwert** der **ungeraden** Zahlen, die sich in der Liste befinden, zurückgeben.

Beispielaufruf: `calc_odd_numbers_mean([2, 3, 6, 7])` → Ergebnis: 2

**Aufgabe 6 - Klassen schreiben** **(15 Punkte)**

Legen Sie eine Klasse **University** an, mit der sich Universitäten als Objekte in Python repräsentieren lassen. Die Klasse soll über folgende Instanzvariablen verfügen:

1. Name der Universität
2. Gründungsjahr
3. Stadt, in der sich die Universität befindet
4. Anzahl der Studentinnen und Studenten
5. Anzahl der Fakultäten
6. Name der Präsidentin/des Präsidenten

Fügen Sie Ihrer Klasse eine Methode **get\_age\_of\_university()** hinzu, die einen Übergabeparameter **current\_year** erwartet. Die Rückgabe soll das Alter der Universität sein, wobei dieser Wert basierend auf der Differenz des übergebenen Parameters und der Instanzvariable des Gründungsjahres berechnet werden soll.

Schreiben Sie außerdem eine Methode **new\_president\_elected()**, die als Übergabeparameter **new\_president** erwartet und mit diesem Wert die Instanzvariable "Name der Präsidentin/des Präsidenten" überschreibt. Die Funktion hat keine Rückgabe.

Legen Sie eine Instanz Ihrer Klasse zur Repräsentation der Universität Regensburg an. Nutzen Sie dazu die folgenden Informationen. Name: *Universität Regensburg*; Gründungsjahr: *1962*; Stadt: *Regensburg*; Anzahl Studierende: *19749*; Anzahl Fakultäten: *12*; Präsident: *Udo Hebel*



**Aufgabe 7 - Codeverständnis: Was ist der Output? (8 Punkte)**

Gegeben sei folgende Funktion:

```
def separate_csv_values(csv_string):
    try:
        chars_of_words = []
        for word in csv_string.split(','):
            word = word.strip()
            if len(word) > 0:
                chars_of_words.append(word)
            else:
                print('one field empty')
                break
        print(chars_of_words)
    except:
        print('ERROR')
```

Geben Sie die Konsolenausgabe an, die entsprechend des angegebenen Übergabeparameters beim Funktionsaufruf erzeugt wird.

Übergabe	'my, name, is'	'some\tstring'	'my, \n, is'	True
Ausgabe				