

Paket-Zuordnung im Lager



Die CodeSolutions Neumarkt GmbH wurde beauftragt einen **Software-Prototypen** für die **Zuordnung** von **Paketen** zu **Fächern** in einem Lagersystem zu entwickeln.

Im Rahmen eines **Kundenworkshops** wurde in Zusammenarbeit mit dem Kunden eine **Technische Beschreibung** verfasst, welche die Grundlage für das Projektvorhaben darstellt.

Außerdem wurde während des Kundenworkshops ein rudimentäres **Funktionsmuster** entwickelt, welches die **Funktionsweise** und die **Interaktion** mit der Software grundsätzlich demonstriert.

Ihr Projektleiter überreicht Ihnen die Technische Beschreibung...

Das Projekt besteht aus
drei Arbeitspaketen.

Sie sollten bis zum
Mittwoch, 09.10.2024
mit der Bearbeitung des
Arbeitspakets 1 fertig
sein.

Beginnen Sie also mit
Arbeitspaket 1 (AP1).



Handlungsauftrag AP1: Informieren

Lesen Sie die folgenden Kapitel der **Technischen Beschreibung** durch:

- *1.1 Aufgabenstellung bis 1.5.1 Arbeitsinhalte von Arbeitspaket 1*

Machen Sie sich Gedanken zu folgenden **Fragestellungen**:

- Welche **Arbeitsinhalte** müssen für AP1 durchgeführt werden?
- Wie könnten Sie die **Zuordnungs-Regeln** im Programm umsetzen?



Eigenarbeit, 12 Minuten



Welche **Arbeitsinhalte** müssen für AP1 durchgeführt werden?

1. Entwurf des Programmablaufplans
2. Entwicklung der Software
3. Testen der Software
4. Dokumentation des Programmablaufs

Seite 2

1.5 ARBEITSINHALTE

1.5.1 AP1: Entwurf & Entwicklung einer Software zur einmaligen Zuordnung von Paketen

Die Bearbeitung von AP1 umfasst folgende Inhalte:

1. Entwurf des Programmablaufplans
2. Entwicklung der Software
3. Testen der Software
4. Dokumentation des Programmablaufs



Wie könnten Sie die Zuordnungs-Regeln im Programm umsetzen?

Nr.	Regel	Fach
R1	Die Paketnummer ist restlos durch 4 oder 5 teilbar.	1
R2	Die Paketnummer ist zusätzlich restlos durch 100 teilbar.	2
R3	Die Paketnummer ist zusätzlich restlos durch 400 teilbar.	3
R4	R1, R2 und R3 treffen nicht zu.	4



Logische Operatoren **and** und **or**

- Wahrheitstabelle

Bedingung A		Bedingung B	UND (and)	ODER (or)
0	or	0	0	0
0	or	1	0	1
1	or	0	0	1
1	or	1	1	1

0 = False 1 = True



Logische Operatoren and und or

- Wahrheitstabelle

Bedingung A			Bedingung B	UND (and)	ODER (or)
0	and	0	0	0	0
0	and	1	1	0	1
1	and	0	0	0	1
1	and	1	1	1	1

0 = False 1 = True



Logische Operatoren **and** und **or**

- Beispiele

```
paketnummer = 10
```

```
if paketnummer % 4 == 0 or paketnummer % 5 == 0:  
    # Wird der Anweisungsblock ausgeführt?  
    print("Der Anweisungsblock wird ausgeführt.")
```



Ja, wird ausgeführt, da linke Bedingung nicht erfüllt, aber rechte Bedingung erfüllt.

Bedingung A	Bedingung B	ODER
0	1	1



Logische Operatoren **and** und **or**

- Beispiele

```
paketnummer = 10
```

```
if paketnummer % 4 == 0 or paketnummer % 5 == 0:  
    # Wird der Anweisungsblock ausgeführt?  
    print("Der Anweisungsblock wird ausgeführt.")
```



Ja, wird ausgeführt, da linke Bedingung nicht erfüllt, aber rechte Bedingung erfüllt.

Bedingung A	Bedingung B	ODER (or)
0	1	1



Logische Operatoren **and** und **or**

- Beispiele

```
paketnummer = 7

if paketnummer % 4 == 0 or paketnummer % 5 == 0:
    # Wird der Anweisungsblock ausgeführt?
    print("Der Anweisungsblock wird ausgeführt.")
```



Nein, wird nicht ausgeführt, da beide Bedingungen nicht erfüllt.



Logische Operatoren **and** und **or**

- Beispiele

```
a = 15

if a > 10 and a < 20:
    # Wird der Anweisungsblock ausgeführt?
    print("Der Anweisungsblock wird ausgeführt.")
```



Ja, wird ausgeführt, da beide Bedingungen erfüllt.



Logische Operatoren **and** und **or**

- Beispiele

```
if False and True:  
    # Wird der Anweisungsblock ausgeführt?  
    print("Der Anweisungsblock wird ausgeführt.")
```



Nein, wird nicht ausgeführt, da linke Bedingung nicht erfüllt.



Wie könnten Sie die Zuordnungs-Regeln im Programm umsetzen?

Nr.	Regel	
R1	Die Paketnummer ist restlos durch 4 oder 5 teilbar.	
R2	Die Paketnummer ist zusätzlich restlos durch 100 teilbar.	2
R3	Die Paketnummer ist zusätzlich restlos durch 400 teilbar.	3
R4	R1, R2 und R3 treffen nicht zu.	4

Was bedeutet
zusätzlich?



Wie könnten Sie die Zuordnungs-Regeln im Programm umsetzen?

Nr.	Regel	Fach
R1	Die Paketnummer ist restlos durch 4 oder 5 teilbar.	
R2	Die Paketnummer ist zusätzlich restlos durch 100 teilbar.	
R3	Die Paketnummer ist zusätzlich restlos durch 400 teilbar.	3
R4	R1, R2 und R3 treffen nicht zu.	4

Was bedeutet
zusätzlich?



Handlungsauftrag AP1: Planen

- Erstellen Sie einen **PAP** mit geeigneten Variablen, welcher die beschriebenen Anforderungen von **AP1** mit folgenden **Regeln** erfüllt:

Nr.	Regel	Fach
R1	Die Paketnummer ist restlos durch 4 oder 5 teilbar.	1
R2	Die Paketnummer ist zusätzlich restlos durch 100 teilbar.	2
R3	Die Paketnummer ist zusätzlich restlos durch 400 teilbar.	3
R4	R1, R2 und R3 treffen nicht zu.	4

- HINWEIS: UND-Verknüpfungen sind hier nicht zwangsläufig notwendig.



Zusammenarbeit mit Nachbar, 12 Minuten



Handlungsauftrag AP1: Entwickeln

- **Schreiben** Sie das entsprechende **Python-Skript** in einer neuen Code-Datei „Paketzuordnung_AP1.py“.
- **Testen** Sie Ihr Programm während und nach der Entwicklung mit den **Testfällen** aus der Technischen Beschreibung.
- **Laden** Sie Ihre **Codedatei** (.py) mit Ihrem Nachnamen in MS Teams unter *Dateien > Schülerlösungen > Paket-Zuordnung AP1 hoch*.



Zusammenarbeit mit Nachbar, 15 Minuten



Welche Arbeitsinhalte mussten für AP1 durchgeführt werden?

1. Entwurf des Programmablaufplans
2. Entwicklung der Software
3. Testen der Software
4. Dokumentation des Programmablaufs

Haben wir alles erledigt?



Ihr Projektleiter ist zufrieden mit Ihrer Bearbeitung von AP1



Sie haben **gute Arbeit** geleistet in AP1.

Der **Kunde** wird sehr zufrieden sein.

Bearbeiten Sie als nächstes bitte die Arbeitsinhalte aus AP2.



Handlungsauftrag AP2: Informieren

Lesen Sie die folgenden Kapitel der **Technischen Beschreibung** durch:

➤ **1.5.2 Arbeitsinhalte von Arbeitspaket 2**

Beantworten Sie dabei für sich folgende **Fragestellungen**:

- Wie **unterscheiden** sich die **Anforderungen** von AP2 gegenüber AP1?
- Welches **Konstrukt** benötigen Sie für die Programmierung?



Wie unterscheiden sich die Anforderungen von AP2 gegenüber AP1?

AP1: Seite 3

Die Software nimmt **einmalig** eine Paketnummer eines Pakets in Form einer Konsoleneingabe als Zeichenkette entgegen.

AP2: Seite 5

Die Software nimmt **kontinuierlich** eine reguläre Paketnummer eines Pakets oder den Befehl zum Programmabbruch in Form einer Konsoleneingabe als Zeichenkette entgegen.

Wenn es sich bei der Konsoleneingabe um eine reguläre Paketnummer handelt, ordnet die Software die Paketnummer gemäß der Regeln im Kapitel „Zuordnung“ dem entsprechenden Fach zu. Die Software gibt die Nummer des Fachs, welchem das Paket zugeordnet wird, über eine Konsolenausgabe aus.

Nach der Ausgabe des Fachs fährt die Software mit einer erneuten Eingabeaufforderung fort.

Wenn es sich bei der Konsoleneingabe um die **Eingabe der Paketnummer 0** handelt, erfolgt keine Zuordnung und die Software wird **unmittelbar beendet**.



Handlungsauftrag AP2: Informieren

Informieren Sie sich zur **while-Schleife** mithilfe der folgenden **Website** oder des **Fachbuchs**:

- Website: *python-kurs.eu/schleifen.php*
- *Kapitel 8.4 im Fachbuch: „while“-Schleife*

Beantworten Sie die folgende **Fragestellung**:

- Welche Funktion hat das Schlüsselwort **break**?



Zusatzaufgabe: Bringen Sie mit einer Internetrecherche in Erfahrung, welche Funktion das Schlüsselwort **continue** hat.



Eigenarbeit, 10 Minuten



Handlungsauftrag AP2: Entwickeln

- **Schreiben** Sie das entsprechende **Python-Skript** in einer neuen Code-Datei „Paketzuordnung_AP2.py“.
- **Testen** Sie Ihr Programm während und nach der Entwicklung mit den **Testfällen** aus der Technischen Beschreibung.
- **Laden** Sie Ihre **Codedatei** (.py) mit Ihrem Nachnamen in MS Teams unter *Dateien > Schülerlösungen > Paket-Zuordnung AP2 hoch*.



Handlungsauftrag AP2: Entwickeln

- **Schreiben** Sie das entsprechende **Python-Skript** in einer neuen Code-Datei „Paketzuordnung_AP2.py“.
- **Testen** Sie Ihr Programm während und nach der Entwicklung mit den **Testfällen** aus der Technischen Beschreibung.
- **Laden** Sie Ihre **Codedatei** (.py) mit Ihrem Nachnamen in MS Teams unter *Dateien > Schülerlösungen > Paket-Zuordnung AP2 hoch*.



Welche Arbeitsinhalte mussten für AP2 durchgeführt werden?

1. Entwurf des Struktogramms
2. Entwicklung der Software
3. Testen der Software
4. Dokumentation des Programmablaufs

Haben wir die Deadline eingehalten?



Ihr Projektleiter ist zufrieden mit Ihrer Bearbeitung von AP2



Danke für die
gewissenhafte
Bearbeitung von AP2.

Der **Kunde** wird sehr
zufrieden sein.

