

# Niveaubestimmende Aufgabe zum Fachlehrplan Informatik Fachgymnasium

„Ein Programm für eine Arztpraxis erstellen“

Schuljahrgang 12

Arbeitsstand: 28.04.2017

Niveaubestimmende Aufgaben sind Bestandteil des Lehrplankonzeptes für das Gymnasium und das Fachgymnasium. Die nachfolgende Aufgabe soll Grundlage unterrichtlicher Erprobung sein. Rückmeldungen, Hinweise, Anregungen und Vorschläge zur Weiterentwicklung der Aufgabe senden Sie bitte über die Eingabemaske (Bildungsserver) oder direkt an [andrea.neubauer@lisa.mb.sachsen-anhalt.de](mailto:andrea.neubauer@lisa.mb.sachsen-anhalt.de)

An der Erarbeitung der niveaubestimmenden Aufgabe haben mitgewirkt:

Hamann, Beate  
Krüger, Hartmut  
Rühlmann, Michael  
Schulze, Holger

Burg  
Stendal  
Quedlinburg  
Halle (Leitung der Fachgruppe)

Herausgeber im Auftrag des Ministeriums für Bildung des Landes Sachsen-Anhalt:  
Landesinstitut für Schulqualität und Lehrerbildung Sachsen-Anhalt  
Riebeckplatz 09  
06110 Halle



Die vorliegende Publikation, mit Ausnahme der Quellen Dritter, ist unter der „Creative Commons“-Lizenz veröffentlicht.

 CC BY-SA 3.0 DE

<http://creativecommons.org/licenses/by-sa/3.0/de/>

Sie dürfen das Material weiterverbreiten, bearbeiten, verändern und erweitern. Wenn Sie das Material oder Teile davon veröffentlichen, müssen Sie den Urheber nennen und kennzeichnen, welche Veränderungen Sie vorgenommen haben. Sie müssen das Material und Veränderungen unter den gleichen Lizenzbedingungen weitergeben.

Die Rechte für Fotos, Abbildungen und Zitate für Quellen Dritter bleiben bei den jeweiligen Rechteinhabern, diese Angaben können Sie den Quellen entnehmen. Der Herausgeber hat sich intensiv bemüht, alle Inhaber von Rechten zu benennen. Falls Sie uns weitere Urheber und Rechteinhaber benennen können, würden wir uns über Ihren Hinweis freuen.

# Ein Programm für eine Arztpraxis erstellen

## Qualifikationsphase

### Aufgabe 1

Für eine Arztpraxis soll ein kleines Programm erstellt werden. Die Datei **arztpraxis.xlsx** (**Anlage 1**) enthält bereits zwei UserFormen.

Die Patienten ziehen im Wartezimmer eine Nummer. Die Nummern 1 – 10 werden für das Sprechzimmer 1 und die Nummern 11 – 20 für das Sprechzimmer 2 vergeben. Nachdem die Arzthelferin die nächste Nummer eingegeben und den Button „Next“ betätigt hat, wird das jeweilige Sprechzimmer angezeigt (**Anlage 2**).

- 1.1 Informieren Sie sich über die verschiedenen Arten der Verzweigungen (**Anlage 3**).
- 1.2 Das Programm für die UserForm\_Wartezimmer arbeitet noch nicht ganz korrekt (siehe Ausgangssituation oben).
  - a) Finden Sie den Fehler im Quellcode. Tauschen Sie sich dazu mit Ihrem Banknachbarn über den vorgegebenen Quellcode aus (**Anlage 2 – UserForm\_Wartezimmer und Quellcode**).
  - b) Entwerfen Sie den neuen bzw. erweiterten Algorithmus und testen Sie Ihre Lösung am Computer auf Funktionsfähigkeit.

### Aufgabe 2

Der Patient wird vom Arzt untersucht. Anschließend beantwortet der Arzt im Programm (**Anlage 4 - UserForm\_Abfrage und Quellcode**) die Frage, ob eine Krankschreibung erfolgt, mit ja oder nein. Nachdem er auf den Button „Bestätigen“ geklickt hat, werden weitere Objekte angezeigt.

- 2.1 Analysieren Sie gemeinsam mit Ihrem Banknachbarn die UserForm\_Abfrage und den dazugehörigen Quellcode. Beachten Sie dabei insbesondere die Eigenschaften der Objekte (z. B. Name).
- 2.2 Interpretieren Sie jede Zeile schriftlich.
- 2.3 Erstellen Sie zwei neue UserFormen (UserForm\_KS, UserForm\_Rezept) nach vorliegendem Muster (**Anlage 5**). Beachten Sie hierbei die Auswahl der geeigneten Objekte und die korrekte Beschriftung.

### Aufgabe 3

In der UserForm\_Abfrage sind auf die Fragen nach der Krankschreibung und dem Rezept durch die Eingabe vom Arzt verschiedene Antwortkombinationen möglich (**Anlage 6**). Nach dem Klick auf den Bestätigungsbutton für das Rezept soll sich je nach Antwortkombination die UserForm\_Rezept oder die UserForm\_Krankenschein öffnen.

3.1 Entwerfen Sie dazu den Algorithmus. Beachten Sie folgende Hinweise:

- Deklarieren und verwenden Sie Variablen,
- Der Cursor sollte beim Programmstart immer im ersten Feld, welches auszufüllen ist, stehen,
- Alle Button (außer Drucken) sollen funktionieren (**Anlage 5**).

3.2 Testen Sie anschließend die Funktionsfähigkeit Ihres Programms und korrigieren Sie gegebenenfalls Fehler.

ERPROBUNG

## Materialien

- Anlage 1: arztpraxis.xlsm
- Anlage 2: UserForm\_Wartezimmer (Skizze mit Beispieleingabe und Quellcode)

Wartezimmer ×

### Der Nächste bitte!

**Nummer:**

**Sprechzimmer:**

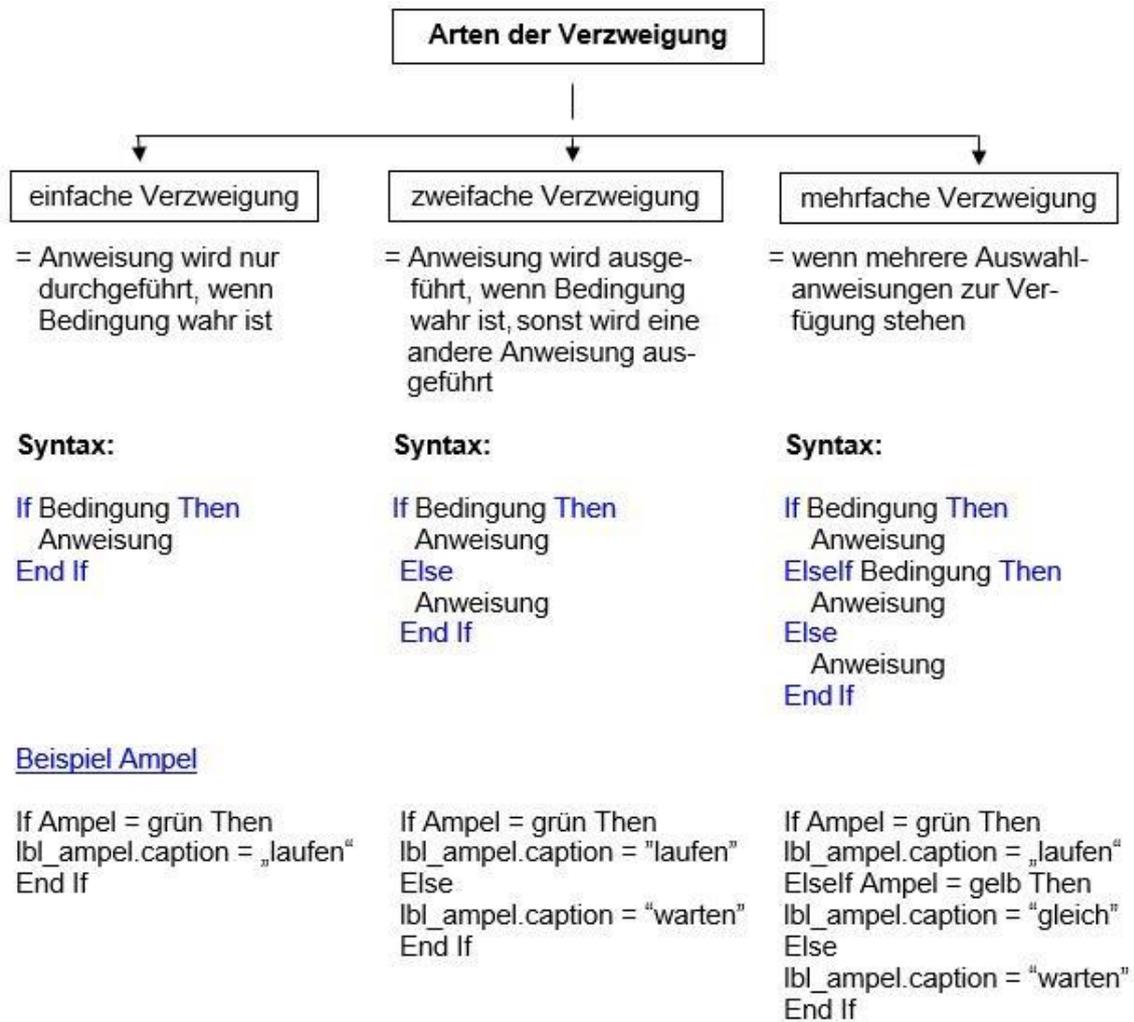
Quellcode:

```
Private Sub cmd_abfrage_Click()  
    UF_Abfrage.Show  
End Sub
```

```
Private Sub cmd_next_Click()  
    Dim nr As Integer  
  
    nr = txt_nr.value  
  
    If nr <= 10 Then  
        lbl_spz.caption = 1  
    End If
```

```
End Sub
```

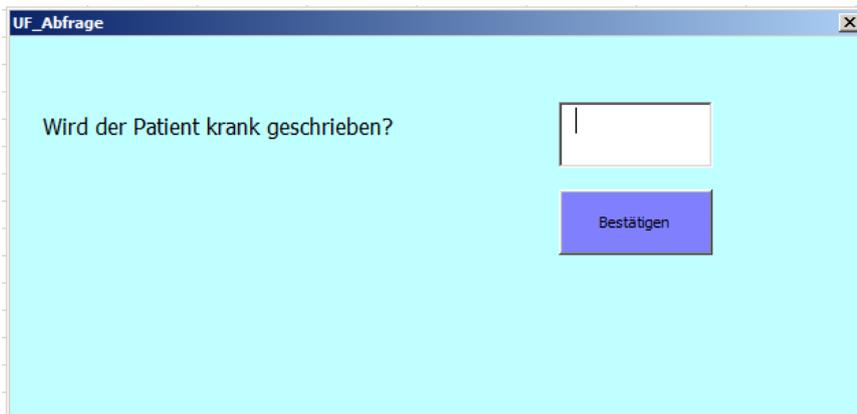
– Anlage 3: Informationstext – Arten der Verzweigungen



EN

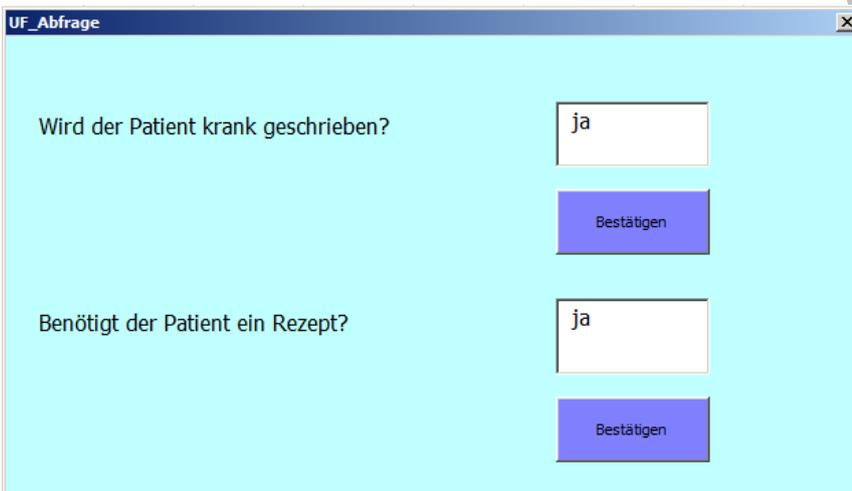
- Anlage 4: UserForm\_Abfrage (Skizze und Quellcode)

UserForm vor der Eingabe



The screenshot shows a window titled "UF\_Abfrage" with a light blue background. It contains a single question: "Wird der Patient krank geschrieben?". To the right of the question is an empty text input field. Below the input field is a blue button labeled "Bestätigen".

UserForm nach einer Beispieleingabe und dem Betätigen des 1. Bestätigungsbuttons



The screenshot shows the same "UF\_Abfrage" window after two interactions. The first question, "Wird der Patient krank geschrieben?", now has the text "ja" entered in the input field. Below it is the "Bestätigen" button. A second question, "Benötigt der Patient ein Rezept?", is now visible below the first one. It also has "ja" entered in its input field and a "Bestätigen" button below it.

Quellcode: für die Frage nach der Krankschreibung (cmd\_krank)

Dim krank, rezept As String

**Private Sub cmd\_krank\_Click()**

krank = txt\_krank.Text

If krank = "ja" Then

    lbl\_rezept.Visible = True

    txt\_rezept.Visible = True

    cmd\_rezept.Visible = True

    txt\_rezept.SetFocus

Elseif krank = "nein" Then

    lbl\_rezept.Visible = True

    txt\_rezept.Visible = True

    cmd\_rezept.Visible = True

    txt\_rezept.SetFocus

Else

    MsgBox("Geben Sie bitte ja oder nein ein")

End If

**End Sub**

- Anlage 5: Muster Krankenschein, Rezept

UserForm\_KS

UF\_KS

### Arbeitsunfähigkeitsbescheinigung

Anschrift

Name:  Vorname:

Arbeitsunfähig seit:

Voraussichtlich bis:

Festgestellt am:

Drucken Ende

UserForm\_Rezept

UF\_Rezept

### Rezept

Menge	Bezeichnung	Preis
<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>

Drucken Krankenschein

Ende

- Anlage 6: Mögliche Antwortkombinationen auf die Fragen nach der Krankschreibung und dem Rezept

<b>Mögliche Antwortkombinationen auf die Fragen nach der Krankschreibung und dem Rezept</b>		
<b>Krankschreibung?</b>	<b>Rezept?</b>	<b>Ereignisse (bei Klick auf den Button)</b>
Nein	Ja	Wenn bei der Krankschreibung „Nein“ und beim Rezept „Ja“ steht, dann soll sich die UserForm_Rezept öffnen ...
Ja	Nein	... sonst, wenn bei der Krankschreibung „Ja“ und beim Rezept „Nein“ steht, dann soll sich die UserForm_KS öffnen ...
Ja	Ja	... sonst, wenn bei der Krankschreibung „Ja“ und beim Rezept „Ja“ steht, dann soll sich die UserForm_Rezept öffnen ...
		... sonst soll sich das Formular schließen.

## Einordnung in den Fachlehrplan

### Kompetenzschwerpunkt:

- Abläufe und Daten strukturiert modellieren und implementieren

### zu entwickelnde Schlüsselkompetenzen:

- einen Fachtext erschließen, Fachsprache verwenden
- Algorithmen entwerfen, gemeinsam Lösungen finden und diskutieren

### zu entwickelnde fachspezifische Kompetenzen:

- Problemstellungen erfassen und mittels Algorithmen Programmabläufe formulieren
- Programmstrukturen zur Umsetzung von Algorithmen ermitteln und vergleichen
- Algorithmen mithilfe einer aktuellen Programmiersprache in geeignete Programmstrukturen umsetzen
- Unterschiedliche Lösungsvarianten gegenüberstellen und unter Beachtung des Sicherheitsbedürfnisses diskutieren
- Fehler in der Programmierung durch Einsatz geeigneter Testszenarien erkennen und beheben

### Bezug zu grundlegenden Wissensbeständen:

- Variablenkonzept
- Algorithmische Grundbausteine und ihre Implementierung (z. B. Verzweigungen)

## **Anregungen und Hinweise zum unterrichtlichen Einsatz**

Die Lösung der Aufgabe erfolgt am Computer mit einer eingeführten Programmiersprache. Diese Aufgabe wurde mit VBA gelöst.

Für die Aufgabe bietet sich Partnerarbeit an.

Die Schülerinnen und Schüler besitzen bereits erste Erfahrungen im Programmieren. Sie sollten bereits die Variablentypen kennen. Zum Unterrichtsbeginn sollten den Schülerinnen und Schülern nochmals Hinweise zur Speicherung (Speicherort, Dateiname, Dateityp, z. B. mit Makros ...) gegeben werden.

Für die Lösung der Aufgabe 2.2 empfiehlt es sich, dass die Schülerinnen und Schüler in der Anlage 4 die Zeilen des Quellcodes nummerieren. So kann die zeilenweise Interpretation den entsprechenden Nummerierungen zugeordnet werden.

Die Arbeitszeit für die Lösung der niveaubestimmenden Aufgabe beträgt ca. drei Unterrichtsstunden.

## Variationsmöglichkeiten

Die Umsetzung der Aufgabe kann mit einer anderen Programmiersprache erfolgen.

Die Aufgabe kann zu einem Gruppenpuzzle erweitert werden. So können andere Gruppen nach diesem Muster weitere Themen bearbeiten, z. B. Schleifen, Zufallszahl ...

Die Ergebnisse können von den Gruppen in eine geeignete Präsentationsform gebracht bzw. für die Mitschülerinnen und Mitschüler Handouts entwickelt werden.

Es empfiehlt sich eine anschließende Übungsphase, so dass alle Schülerinnen und Schüler die Verzweigungen, Schleifen und Zufallszahl anwenden können. Dadurch würde sich der Zeitanteil erhöhen.

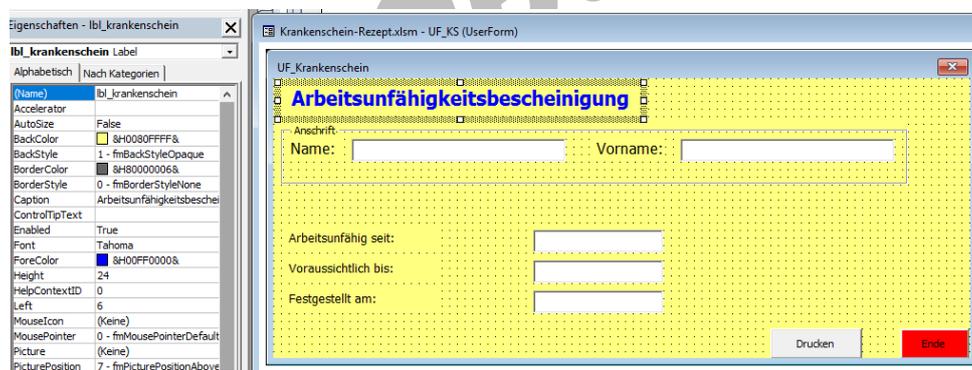
Die Aufgabe 2.2 kann auch als Leistungskontrolle durchgeführt werden.

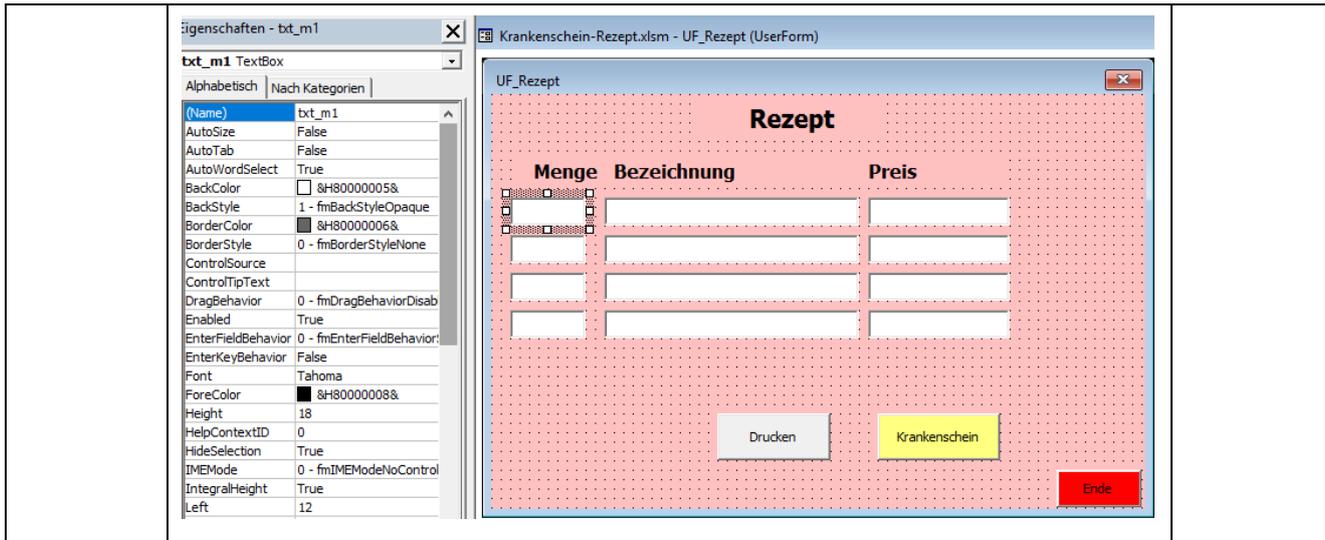
ERPROBUNG

## Erwarteter Stand der Kompetenzentwicklung

Aufgabe	erwartete Schülerleistung	prozent. Anteil
1.1 - 1.2	<p>Die Schülerinnen und Schüler können:</p> <ul style="list-style-type: none"> <li>– mithilfe des Fachtextes den Fehler im Quellcode finden und den Algorithmus entsprechend korrigieren bzw. ergänzen,</li> <li>– die Funktionsfähigkeit des Algorithmus testen.</li> </ul> <pre> <b>Private Sub cmd_next_Click()</b> Dim nr As Integer  nr = txt_nr.Value  If nr &lt;= 10 Then lbl_spz.Caption = 1 <b>Else</b> <b>lbl_spz.Caption = 2</b> End If  <b>End Sub</b>                     </pre>	15 %
2.1 - 2.3	<p>Die Schülerinnen und Schüler können:</p> <ul style="list-style-type: none"> <li>– den Quellcode und die Eigenschaften der Objekte in der UserForm_Abfrage analysieren und jede Zeile des Quellcodes interpretieren,</li> </ul> <p><b>Interpretation</b></p> <ol style="list-style-type: none"> <li>(1) Variablendeklaration – Variablen „krank“ und „rezept“ vom Datentyp „Zeichenfolge“</li> <li>(2) Start der Prozedur bei Klick auf den Button „cmd_krank“</li> <li>(3) Die Variable „krank“ ergibt sich aus dem Text des Textfeldes „txt_krank“</li> <li>(4) Wenn die Variable „krank“ das Wort „Ja“ enthält, dann</li> <li>(5) wird das Bezeichnungsfeld „lbl_rezept“ sichtbar</li> </ol>	60 %

- (6) wird das Textfeld „txt\_rezept“ sichtbar
  - (7) wird der Button „cmd\_rezept“ sichtbar
  - (8) steht der Cursor im Textfeld „txt\_rezept“
  - (9) sonst, wenn die Variable „krank“ das Wort „nein“ enthält, dann wird das Bezeichnungsfeld „lbl\_rezept“ sichtbar
  - (10) wird das Textfeld „txt\_rezept“ sichtbar
  - (11) wird der Button „cmd\_rezept“ sichtbar
  - (12) steht der Cursor im Textfeld „txt\_rezept“
  - (13) sonst
  - (14) wird ein Ausgabefenster mit dem Satz „Geben Sie bitte ja oder nein ein“ angezeigt
  - (15) Ende der mehrfachen Verzweigung
  - (16) Ende der Prozedur
- zwei neue UserFormen entsprechend dem vorgegebenen Muster erstellen und die Objekte korrekt auswählen und beschriften.





<p>3.1 - 3.2</p>	<p>Die Schülerinnen und Schüler können:</p> <ul style="list-style-type: none"> <li>– den Algorithmus für den Bestätigungsbutton Rezept entwerfen,</li> <li>– Variablen deklarieren und verwenden,</li> <li>– die Funktionsfähigkeit des Algorithmus testen und eventuelle Fehler korrigieren.</li> </ul> <p><b>Algorithmus für den Button „cmd_rezept“</b></p> <pre> Dim krank, rezept As String <b>Private Sub cmd_rezept_Click()</b> krank = txt_krank.Text rezept = txt_rezept.Text  If rezept = "ja" And krank = "nein" Then     UF_Rezept.Show Elseif rezept = "nein" And krank = "ja" Then     UF_KS.Show Elseif rezept = "ja" And krank = "ja" Then     UF_Rezept.Show Else     Unload Me End If  <b>End Sub</b> </pre>	<p>25 %</p>
------------------	--	-------------