

```
5 echo "Nachname: $_POST['Nachname']<br>";
if (!$_POST['Ort'])
    echo "Wohnort: nicht angegeben</p>";
else
    echo "Wohnort: $_POST['Ort']</p>";
}
```

- 1 Bevor Sie versuchen, Variablen auszulesen, sollten Sie testen, ob eine bestimmte Variable einen Wert enthält. Dies erreichen Sie über den Befehl `isset()`. Hiermit kontrollieren Sie, ob die über die Methode POST übergebene Variable `$abgeschickt` einen Wert besitzt.
- 2 Wurde die Variable gesetzt, so kontrollieren Sie, ob diese den Wert 1 besitzt. Dieser Wert wird in der HTML-Seite über das versteckte Feld gesetzt.
- 3 Nun beginnen Sie, die einzelnen Daten auszulesen und anzuzeigen. Als Erstes kontrollieren Sie, ob der Nutzer das Eingabefeld Vorname ausgefüllt hat. Daraufhin wird dementsprechend ein Hinweis oder der Wert der Variable `$Vorname` ausgegeben.
- 4 Die Abfrage geschieht ebenfalls mit dem Wert des Eingabefeldes Nachname.
- 5 Zu guter Letzt wird der Wert der Variable `$Ort` überprüft und ausgegeben.

9.3 Formulare erzeugen

Per PHP können Sie auch Formulare erzeugen und die Felder mit bereits eingegebenen Werten füllen. Dies ist beispielsweise nützlich, wenn Sie die Eingaben eines HTML-Formulars überprüfen und feststellen, dass nicht alle Pflichtangaben ausgefüllt worden sind.



Zuerst erstellen Sie die HTML-Seite, in der sich das Formular befindet. In diesem Formular sollen die Nutzer Angaben zu Ihrer Person hinterlassen.

Für dieses Beispiel können Sie die bereits vorhandene Datei `form_post.html` nutzen. Ersetzen Sie in dieser Datei die Zeile

```
<form action="form_post.php" method="POST">
```

durch den Aufruf des PHP-Skripts `form_fill.php` mit der GET-Methode.

```
<form action="form_fill.php" method="GET">
```

Die Methode GET soll Ihnen die übergebenen Werte in der Adresszeile des Browsers sichtbar machen. Speichern Sie die HTML-Datei unter dem Namen `form_fill.html`.