

cbs-tipps 06/2023

Arbeitszeit dezimal

Arbeitszeit dezimal					
Beginn	Ende	AZ Soll	Wunschpausenzeit	max. Pausenzeit	geleistet [h]
07:30	17:00	8.80	00:30	00:42	9.00

Sie haben täglich eine Arbeitszeit von 8.8 Stunden zu leisten, damit bei Ihrer Lohnabrechnung keine Minusstunden auftauchen. Sie beginnen am Morgen um 07:30 Uhr und möchten um 17:10 Uhr Feierabend machen. Nun möchten Sie schnell wissen, wie lange Sie Mittagspause machen können, um auf Ihre 8.8 h zu kommen. Da stellt sich nun das Problem, dass wir einerseits Uhrzeitangaben haben und andererseits das Soll in Dezimalstunden.

Mathematisches Grundproblem

- a) Erst einmal geht es darum, die Differenz der Zeit von 07:30 Uhr bis 17:10 Uhr zu berechnen. In der Volksschule zählt das Kind erst einmal die vollen Stunden. Also 08:00 bis 17:00 Uhr und kommt dabei auf 9 Stunden. Dazu kommen noch die 30 Minuten von 07:30 – 08:00 Uhr plus die 10 Minuten nach 17:00 Uhr, demnach 40 Minuten. Sie würden also ohne Mittagspause 9 h 40 min (oder 9:40 h) arbeiten.

Excel rechnet, wenn die Zellen als Uhrzeit formatiert sind, ganz normal grösserer Wert minus kleinerer Wert gleich Differenz der Uhrzeiten. Sie sehen schon, dass eine Mittagspause drinliegt.

C2			
A	B	C	D
1	Arbeitsbeginn	Arbeitsende	Arbeitszeit
2	07:30	17:10	09:40
3			

- b) Wie viele Dezimalstunden sind nun 9 h 40 min? Beim bürgerlichen Rechnen liegt es nahe, alles in Minuten zu verwandeln. 1 h hat 60 Minuten $\rightarrow 9 * 60 + 40 = 580$ Minuten. Dividiert man diese Minutenzahl durch 60, erhalten wir 9.666... Stunden.

Excel verwendet dazu eigentlich dasselbe Prinzip. Allerdings ist die Zelle der Ausgabe als *Zahl* formatiert. Die Formel lautet dann:

D2			
A	B	C	D
1	Arbeitsbeginn	Arbeitsende	Arbeitszeit
2	07:30	17:10	09:40
3			

Damit wäre das mathematische Grundproblem bereits gelöst, noch nicht aber das Mittagspausenproblem!

Mittagspause

Wir wissen: Sie würden also 9.67 h arbeiten, Ihr Soll ist aber nur 8.8 h, somit bleibt für die Mittagspause $9.67 - 8.8 = 0.87$ h. Wunderbar! Doch wie viele Minuten sind das nun wieder? Jetzt geht es rückwärts $0.87 * 60 = 52.2$ Minuten (rund 52 Minuten Pause).

Excel macht das nun so, wobei die Zelle A2 als Zahl und die Zelle B2 als Uhrzeit formatiert ist:

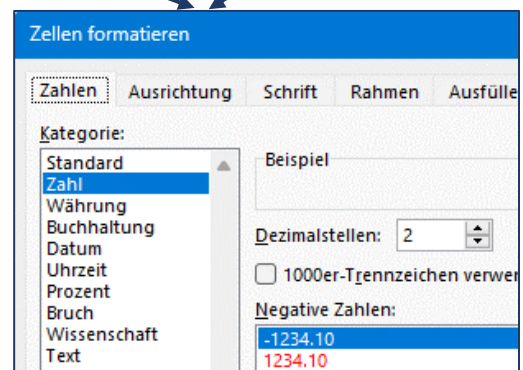
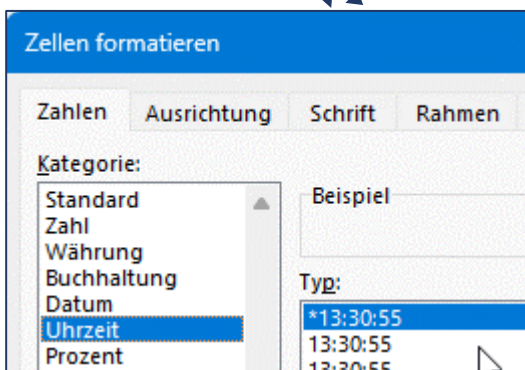
	A	B	C
1	Dezimalstunden	hh:mm:ss	
2	0.87	00:52:12	
3			

Die Exakten unter Ihnen stellen zu Recht die Ungenauigkeit fest, die während des Umrechnens entsteht. Soll es sehr genau werden, so müssen wir über Sekunden oder gar Millisekunden rechnen. Das Verfahren bleibt aber dasselbe.

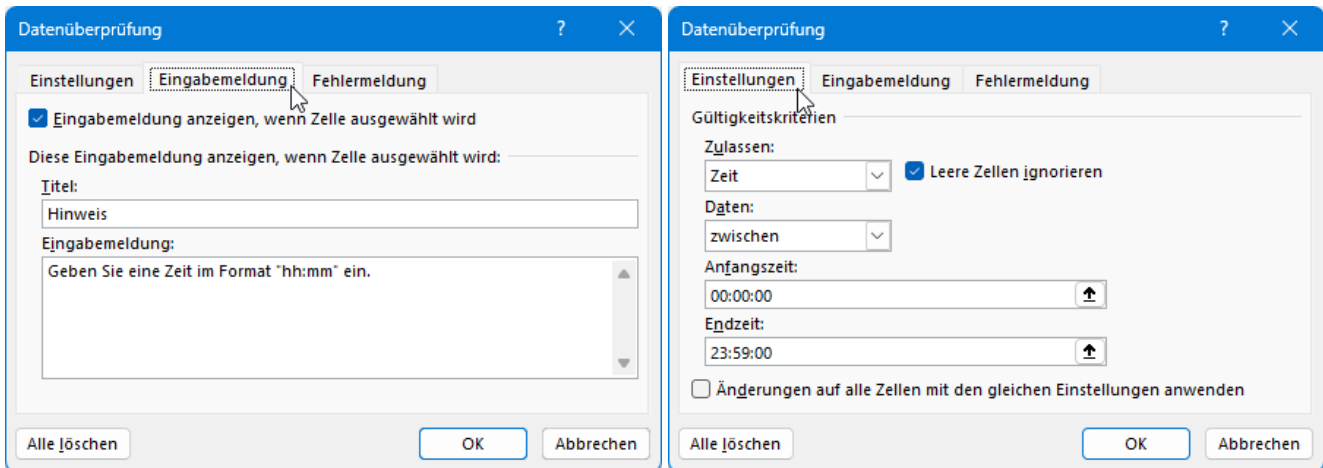
Unser Excel-Tool

Es lohnt sich nun diese Rechnerei in einem kleinen Excel-Tool zusammenzufassen. Starten Sie Excel mit einer leeren Arbeitsmappe und erstellen Sie ein Layout, das folgendermassen aussehen könnte. Oder beginnen Sie mit [dieser](#) formatierten Vorlage.

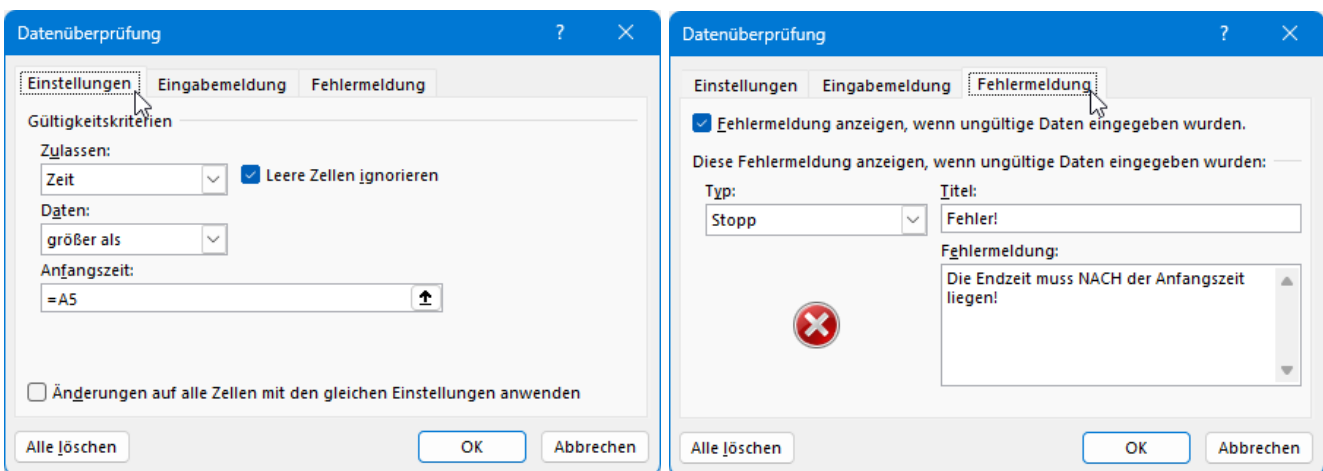
Arbeitszeit dezimal						
3	Beginn	Ende	AZ Soll	Wunschpausenzeit	max. Pausenzeit	geleistet [h]
5	07:30	17:10	8.80		00:00	
7	Berechnungen:					
8	AZ brutto	AZ netto	AZ dez	AZ soll [hh:mm]	Pausenzeit	PZ dez
9						



Nachdem die Zellen richtig formatiert sind (das gilt auf für die Zeile 9), soll die Benutzerin oder der Benutzer noch informiert werden, in welchem Format die Uhrzeiten einzugeben sind. Dazu markieren Sie die Zellen A5 und D5 und wählen im Menü *Daten – Datenüberprüfung – Datenüberprüfung*. Tragen Sie wie folgt ein:



Und für die Zell B5 erstellen wir eine Fehlermeldung über den gleichen Weg.



Berechnungsformeln erstellen

- Zuerst berechnen wir die Arbeitszeit brutto in der Zelle A9 mit der Formel: **=B5-A5**
- Nun berechnen wir in der Zelle D9 die Soll-Arbeitszeit in Stunden und Minuten mit der Formel: **=C5/24**
- Bei der nächsten Berechnung geht es darum, die maximal zur Verfügung stehende Pausenzeit vorzuschlagen. Dazu erzeugen wir in der Zelle E5 folgende Formel: **=A9-D9**

Unser Tool sieht bis jetzt so aus. (diesen Zustand könne Sie mit der Datei [Dezimalstunden-R1.xlsx](#) herunterladen)

	A	B	C	D	E	F
1	Arbeitszeit dezimal					
2						
3	Beginn	Ende	AZ Soll	Wunschpausenzzeit	max. Pausenzzeit	geleistet [h]
4						
5	07:30	17:10	8.80		00:52	
6						
7	Berechnungen:					
8	AZ brutto	AZ netto	AZ dez	AZ soll [hh:mm]	Pausenzzeit	PZ dez
9	09:40				08:48	

Wunschpausenzzeit

Es könnte ja sein, dass Sie nicht unbedingt die ganze zur Verfügung stehende Pausenzzeit benutzen möchten, sondern lieber kürzer oder länger in die Pause gehen möchten, was sich dann natürlich am Schluss auf die geleisteten Stunden auswirkt. Dazu brauchen wir eine Bedingungsformel (WENN), womit ein Pausenwunsch, wenn vorhanden, berücksichtigt wird.

- In die Zelle E9 kommt somit folgende Formel: **=WENN(D5="";E5;D5)**
- Jetzt, da die Pausenzzeit bekannt ist, können wir die Nettoarbeitszeit in Stunden und Minuten berechnen. Dies erledigen wir bei den Berechnungen in der Zelle B9 mit der Formel: **=A9-E9**

Die geleisteten Arbeitsstunden (dem Chef bekanntzugeben!)

Damit nun die geleisteten Arbeitsstunden in der rot umrandeten Zelle F5 dezimal ausgegeben werden, sind noch die folgenden Formeln nötig:

- Zelle C9 **=STUNDE(A9)+MINUTE(A9)/60**
- Zelle F9 **=STUNDE(E9)+MINUTE(E9)/60**
- Zelle F5 **=C9-F9**

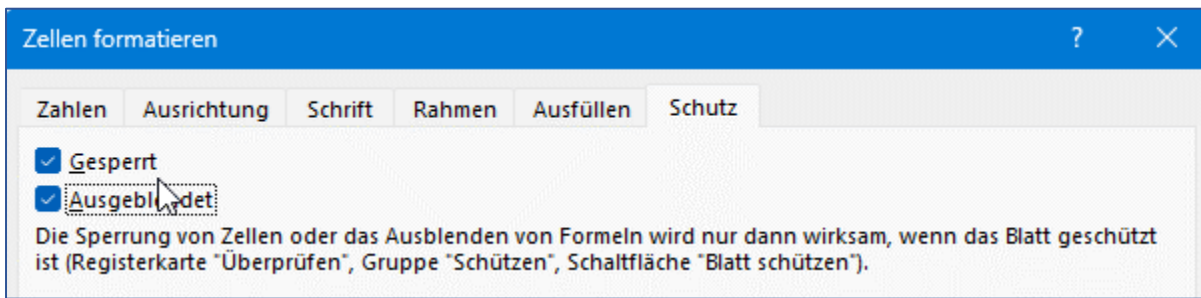
Das war's denn auch schon im Rohbau (Download unter [Dezimalstunden-R2.xlsx](#))!

Kosmetik und Schutz

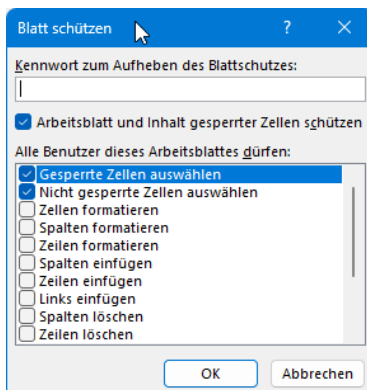
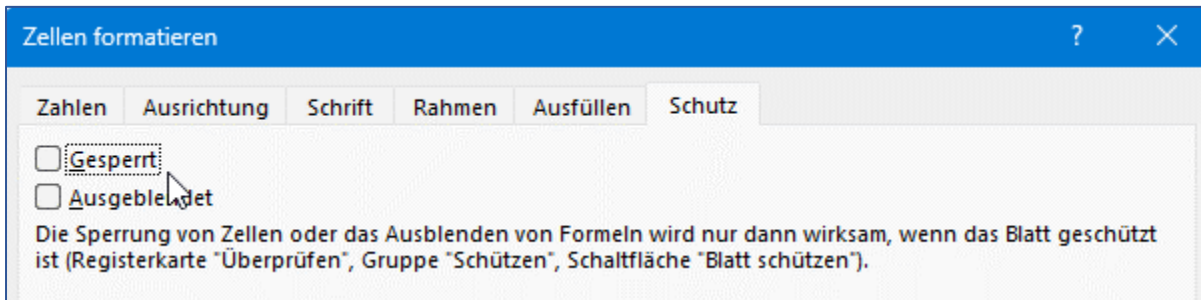
Sie können nun das Blatt noch verschönern, indem Sie die Gitternetzlinien ausblenden und die Berechnungen mit einem Bild oder weissen Rechteck zudecken, oder die Schrift auf weiss stellen und die Schattierung entfernen.

Damit Sie nicht versehentlich durch Eingabe Formeln löschen, sollten Sie das Sheet schützen. Gehen Sie so vor:

Alle Zellen mit Inhalt markieren – *Rechtsklick* – *Zellen formatieren* – *Schutz*



Nun lediglich die Zellen markieren, in die Sie Eingaben machen müssen, also A5 bis D5 und dann auf gleichem Weg die Sperrung für diese Zellen aufheben.



Über das Menü Überprüfen – Blatt schützen gelangen Sie zur gleichnamigen Dialogbox, wo Sie das Blatt mit oder ohne Kennwort schützen können. Nach dem Speichern sind in den gesperrten Zellen keine Eingaben mehr möglich.

Unser Web-Tool

Arbeitszeitberechnung

Beginn:

Ende:

Pausendauer:

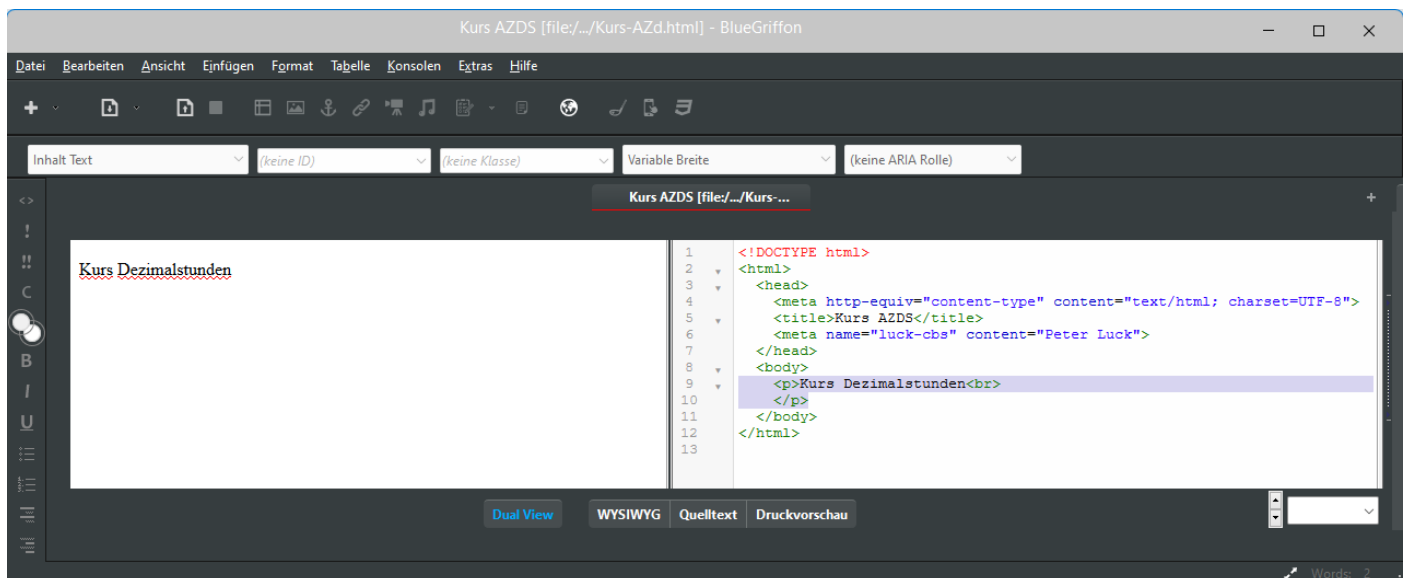
Arbeitszeit:

Dezimalstunden:

Sie möchten das Tool zur dezimalen Arbeitszeitberechnung gerne auf Ihrer Website zur Verfügung haben? Das hat den Vorteil, dass Sie kein Excel benötigen und überall, wo Sie online gehen können, auch Zugriff auf das Instrument haben. Die grundlegenden Überlegungen haben wir ja bereits im Excel-Toll gemacht. Sie wissen, was an Eingaben erforderlich ist und wie etwa die Rechenfunktionen ablaufen müssen. In HTML und JAVA ist nun aber die Syntax der Anweisungen etwas anders. Probieren wir's aus!

Sie benötigen vorzugsweise einen HTML-Editor (welche beliebt sind, sehen Sie [hier](#) im Register html). Sie können den Code aber auch in jedem beliebigen Textprogramm schreiben und in Ihre Website implementieren. Ich verwende hier den Editor [Blue Griffon](#).

Wenn Sie im Editor eine neue Datei beginnen, wechseln Sie zugleich in den *Dual View Mode*. Das hat den Vorteil, dass Sie rechts den Code bearbeiten und links gleich die Auswirkungen sehen. Die HTML-Grundstruktur ist vorgegeben und daran müssen Sie nicht viel ändern, ausser eventuell einen Titel geben und den Autor benennen. Hier, im markierten Bereich beginnt nun Ihre Arbeit. Speichern Sie die Datei gleich unter einem individuellen Namen – bei mir ist das *Kurs AZDS*.



Im ersten Schritt erstellen Sie ein *Script* für einen *addEventListener()* mit *DOMContentLoaded* und bezeichnen die drei Variablen sinnvoll mit einer ID ("bZeit"), ("eZeit") und ("pZeit") für Beginnzeit, Endzeit und Pausenzeit.

```

9   <script>
10
11  window.addEventListener("DOMContentLoaded", function() {
12    document.getElementById("bZeit").addEventListener("change", SumHours);
13    document.getElementById("eZeit").addEventListener("change", SumHours);
14    document.getElementById("pZeit").addEventListener("change", SumHours);
15  });
16

```

Im nächsten Schritt schreiben Sie eine Funktion, um die Zeiten zu summieren. Darin kommen nun Ihre zuvor definierten Variablen zum Zug. Die Zeiten werden nun in Sekunden konvertiert, die Zeitdifferenz berechnet (mit `.toFixed(2)` wird festgelegt, dass in der Dezimalausgabe 2 Nachkommastelle sein sollen) und die IDs der Ausgaben werden auch noch hier erzeugt.

```

17  function SumHours() {
18    var bzeit = document.getElementById('bZeit').value;
19    var ezeit = document.getElementById('eZeit').value;
20    var pzeit = document.getElementById('pZeit').value;
21    var diff = 0;
22
23    if (bzeit && ezeit && pzeit) {
24      bzeit = ConvertToSeconds(bzeit);
25      ezeit = ConvertToSeconds(ezeit);
26      pzeit = ConvertToSeconds(pzeit);
27      diff = Math.abs(ezeit - bzeit - pzeit);
28      document.getElementById('outdez').value = (diff/3600).toFixed(2);
29      document.getElementById('out').value = secondsToHHmmSS(diff)
30    }
31

```

Ab Zeile 32 beginnt die Rechnerei wie zuvor in Excel, einfach etwas anders geschrieben und mit `.splitTime` wird die Ausgabe für die Stunden- und Minutenschreibweise definiert. Dazu schreiben Sie zwei weitere Funktionen und schliessen das Script ab.

```

32  function ConvertToSeconds(time) {
33    var splitTime = time.split(":");
34    return splitTime[0] * 3600 + splitTime[1] * 60;
35  }
36
37  function secondsToHHmmSS(secs) {
38    var hours = parseInt(secs / 3600);
39    var seconds = parseInt(secs % 3600);
40    var minutes = parseInt(seconds / 60);
41    if (minutes < 10){
42      minutes = '0' + minutes;
43    }
44    return hours + ":" + minutes;
45  }
46  }
47  </script>

```

Sie haben sicher festgestellt, dass sich auf der linken Seite in der Vorschau überhaupt nichts getan hat. Das, was Sie bis jetzt geschrieben haben, hat ja schliesslich im Hintergrund zu laufen. Doch jetzt wird sich das ändern. Sie erzeugen in html die Ein- und Ausgabefelder, sowie den sichtbaren Titel des Formulars. Das `<label type="time">` beispielsweise erzeugt eine Uhr für die Zeitauswahl. Schliessen Sie `body` und `html`

ab – fertig!

```
48
49 <h2>Arbeitszeitberechnung</h2>
50 <br>
51 <p><label>Beginn: <input id="bZeit" type="time"></label></p>
52 <p><label>Ende: <input id="eZeit" type="time"></label></p>
53 <p><label>Pausendauer: <input id="pZeit" value="00:00" type="time"></label></p>
54 <p><b>Arbeitszeit: </b> <input id="out" readonly="readonly" type="text"></p>
55 <p><b>Dezimalstunden: </b> <input id="outdez" readonly="readonly" type="text"></p>
56 </body>
57 </html>
58
```

Speichern Sie jetzt Ihr Werk und testen Sie es in der Browservorschau. Sollte es Widererwarten nicht funktionieren, müssen Sie irgendwo einen kleinen Tippfehler gemacht haben – und suchen. Damit alles noch etwas geordneter aussieht, habe ich das Layout noch auf rechtsbündig gesetzt.

Arbeitszeitberechnung	
Beginn:	<input type="time" value="07:30"/>
Ende:	<input type="time" value="17:00"/>
Pausendauer:	<input type="time" value="00:25"/>
Arbeitszeit:	<input type="text" value="9:05"/>
Dezimalstunden:	<input type="text" value="9.08"/>

Arbeitszeitberechnung	
Beginn:	<input type="time" value="07:30"/>
Ende:	<input type="time" value="17:00"/>
Pausendauer:	<input type="time" value="00:25"/>
Arbeitszeit:	<input type="text" value="9:05"/>
Dezimalstunden:	<input type="text" value="9.08"/>

Auf der [HTML-Seite](#) haben Sie notfalls die Möglichkeit, das ganze Script nachzulesen oder herunterzuladen.