

Fehlzeitenquoten-Rechner

Fehlzeiten_Absence_Calculator.xlsx – Dokumentation und Hinweise

HR Business Transparency Consulting

1. Zielsetzung und Modellidee

Das Modell beantwortet eine Frage, die in fast jedem Fehlzeiten-Reporting falsch beantwortet wird: „Wie hoch ist die Fehlzeitenquote?“ Die Antwort hängt davon ab, wogegen man rechnet — Kalendertage, Arbeitstage oder Sollstunden geben drei unterschiedliche Zahlen für denselben Sachverhalt. Und sie hängt davon ab, wie man mit Abwesenheiten umgeht, die einen Monat überschreiten, zum Beispiel eine Kur vom 20. Januar bis 14. Februar.

Der Kerngedanke: statt Von-Bis-Intervalle als ganze Zeilen zu behandeln und mit Formeln über Monatsgrenzen hinweg zu jonglieren, wird jedes Intervall in einzelne Kalendertage zerlegt („explodiert“). Ein 26-Tage-Intervall wird zu 26 Zeilen mit je einem Datum. Danach ordnet sich jeder Tag automatisch seinem echten Kalendermonat zu — das Monatsgrenzen-Problem verschwindet, weil es gar nicht erst entsteht.

Um dies in Excel umzusetzen muss man Daten transformieren und dafür bietet sich **Power Query** an.

In **Power Query** übernimmt **List.Dates** genau diesen **Explodier-Schritt** — und zwar zwingend: Klassische Excel-Formeln wie XVERWEIS oder SUMPRODUCT können nur Werte für bereits existierende Zeilen berechnen, aber keine neuen Zeilen aus einem Von-Bis-Intervall erzeugen. Der Explodier-Schritt selbst ist daher nicht optional, sondern eine Notwendigkeit, die nur Power Query (oder ein gleichwertiger externer Schritt) leisten kann.

Dieses Workbook trennt deshalb zwei Dinge: Der **Explodier-Schritt wurde einmalig ausgeführt**, sein Ergebnis liegt in Fehltagge_Explodiert als eingefrorene Werte vor (Spalten MitarbeiterID, Abwesenheitsart, Datum).

Alles, was danach mit diesen Zeilen geschieht — Kostenstelle nachschlagen, Arbeitstag bestimmen, Sollstunden berechnen — läuft als Live-Formel und ist ohne erneuten Power-Query-Lauf nachprüfbar. Der tatsächliche M-Code für den Explodier-Schritt selbst steht separat im Blatt PowerQuery_M_Code als Vorlage zum Einfügen in den Power-Query-Editor bereit, für den Fall, dass die Tabelle produktiv aktualisierbar gemacht werden soll.

2. Aufbau des Workbooks

Sechs Arbeitsblätter, in der Reihenfolge, in der die Daten fließen:

Blatt	Funktion	Zeilen
Eingaben	Abwesenheitsarten-Lookup, Mitarbeiterkalender, Feiertage Berlin	≈ 190
Fehlzeiten_Rohdaten	Rohe Abwesenheitsintervalle (Von/Bis) — der Input für Power Query	550
Fehltagge_Explodiert	Ein Kalendertag je Zeile, mit Live-Formeln zu Kostenstelle, Arbeitstag-Flag, Sollstunden	3.295
Fehlzeiten_Report	Drei Quoten je Monat: Kalendertage, Arbeitstage, Sollstunden	18 Monate
Zeiterfassung_Abgleich	Reconciliation der berechneten Fehlstunden gegen ein Zeiterfassungssystem	6 Monate
PowerQuery_M_Code	Der echte M-Code für List.Dates, Arbeitstag-Logik und Monatsgruppierung	3 Schritte

Nur Fehlzeiten_Rohdaten und Eingaben enthalten von Hand eingetragene Werte. Alles andere — auch die 3.295 Zeilen in Fehltage_Explodiert — besteht aus Formeln, die auf diese beiden Blätter zurückverweisen.

3. Eingaben im Detail

3.1 Abwesenheitsarten-Lookup

Nicht jede Abwesenheit zählt zur Fehlzeitenquote. Urlaub und Sonderurlaub sind geplante, keine ungeplanten Abwesenheiten und fließen deshalb nicht ein — ein häufiger Fehler in Benchmark-Vergleichen ist, dass die verglichenen Kennzahlen unterschiedlich definiert sind, ohne dass das irgendwo sichtbar wird.

Abwesenheitsart	Zählt zur Fehlzeitenquote?
Krankheit (kurz)	Ja
Krankheit (lang, mit AU)	Ja
Kur/Reha	Ja
Kind krank	Ja
Urlaub	Nein
Sonderurlaub	Nein

3.2 Mitarbeiterkalender und Feiertage

140 Mitarbeitende über 6 Kostenstellen, mit individuellen Vertragsstunden (Vollzeit 40 Std., Teilzeit 20–36 Std.) — das ist die Grundlage für die Sollstunden-Quote, die als einzige der drei Kennzahlen unterschiedliche Arbeitszeiten korrekt gewichtet. Dazu der Berlin-Feiertagskalender 2025–2026 (inklusive Internationaler Frauentag, der in Berlin seit 2019 gesetzlicher Feiertag ist), als Grundlage für die Arbeitstag-Erkennung.

3.3 Fehlzeiten_Rohdaten

550 Intervalle über 18 Monate (Januar 2025 bis Juni 2026), jede Zeile ein Von-Bis-Zeitraum mit MitarbeiterID und Abwesenheitsart. Bewusst enthalten: mehrere lange Kur/Reha-Intervalle (21–28 Tage), die zuverlässig über eine Monatsgrenze laufen — genau der Fall, an dem sich zeigt, ob die Monatszuordnung robust ist.

4. Fehltage_Explodiert — der zentrale Baustein

Jede der 550 Zeilen aus Fehlzeiten_Rohdaten wird in ihre einzelnen Kalendertage aufgelöst. Ein 26-Tage-Intervall wird zu 26 Zeilen. Ergebnis: 3.295 Zeilen, eine je abwesendem Kalendertag.

Warum es ohne Power Query nicht geht

Excel-Formeln können:

- Werte berechnen
- Arrays ($n \times m$ Datentabellen)
- dynamische Bereiche zurückgeben

Aber sie können keine neuen Zeilen erzeugen oder Tabellen „explodieren“.

Spalte	Herkunft	Bedeutung
MitarbeiterID, Abwesenheitsart, Datum	Exponierter Wert (ein Tag aus einem Intervall)	Die eigentlichen „explodierten“ Daten
Kostenstelle	=XVERWEIS gegen Eingaben	Nachgeschlagen, nicht dupliziert
Jahr, Monat	=JAHR(Datum), =MONAT(Datum)	Ordnet jeden Tag seinem echten Kalendermonat zu
Zählt zur Quote	=XVERWEIS gegen Abwesenheitsarten-Lookup	Filtert Urlaub/Sonderurlaub automatisch heraus
Ist Arbeitstag	=WOCHENTAG + ZÄHLENWENN gegen Feiertage	Mo–Fr und kein Feiertag = „Ja“
Tages-Sollstunden	=XVERWEIS gegen Vertragsstunden, geteilt durch 5	Individuelle Tagesarbeitszeit je Mitarbeiter

Der Clou an dieser Konstruktion: Monat, Arbeitstag-Status und Sollstunden werden nicht für das ganze Intervall auf einmal berechnet, sondern für jeden einzelnen Tag separat. Ein Intervall, das einen Monat überschreitet, muss deshalb nirgends speziell behandelt werden — jeder Tag trägt bereits seinen eigenen, korrekten Monat.

5. Fehlzeiten_Report — drei Quoten, ein Zähler-Konzept

Für jeden der 18 Monate werden drei Kennzahlen berechnet, alle mit derselben Zähler-Definition (nur Abwesenheitsarten mit „Zählt zur Quote“ = Ja), aber mit unterschiedlichem Nenner:

Kennzahl	Zähler	Nenner
Kalendertage-Quote	Fehltag (alle 7 Wochentage)	Mitarbeiterzahl × Kalendertage im Monat
Arbeitstage-Quote	Fehltag, nur an Arbeitstagen	Mitarbeiterzahl × NETTOARBEITSTAGE im Monat
Sollstunden-Quote	Fehlstunden (Tages-Sollstunden je Fehltag)	NETTOARBEITSTAGE × Summe Vertragsstunden ÷ 5

Die Sollstunden-Quote ist die einzige der drei, die Teilzeitkräfte korrekt gewichtet: ein Krankheitstag einer 20-Stunden-Kraft entzieht dem Unternehmen 4 Stunden, ein Krankheitstag einer Vollzeitkraft 8 Stunden. Kalendertage- und Arbeitstage-Quote zählen beide als „1 Fehltag“ — für einen Abgleich mit dem Zeiterfassungssystem, das in Stunden bucht, ist deshalb ausschließlich die Sollstunden-Quote der richtige Bezug.

Über den gesamten Betrachtungszeitraum liegen alle drei Quoten nah beieinander (rund 2,9 %) — erwartungsgemäß, da sie korreliert sind. In einzelnen Monaten weichen sie sichtbarer voneinander ab, zum Beispiel wenn viele Abwesenheiten von Teilzeitkräften stammen.

6. Zeiterfassung_Abgleich

Vergleicht die berechneten Fehlstunden aus dem Report mit einem (hier zu Demonstrationszwecken angenommenen) Export aus dem Zeiterfassungssystem, für die letzten 6 Monate des Betrachtungszeitraums. Eine Kontrollspalte markiert Abweichungen über 10 Stunden als „Prüfen“.

Das ist der Unterschied zwischen einem Modell, das nur in sich konsistent ist, und einem Modell, das audit-fest ist: Erst der Abgleich mit einer unabhängigen Datenquelle zeigt, ob die Sollstunden-Logik tatsächlich mit der Realität übereinstimmt.

7. PowerQuery_M_Code — die produktive Umsetzung

Das Blatt enthält den tatsächlichen M-Code für drei Schritte, als Text zum Einfügen in Excels Power-Query-Editor (Daten → Daten abrufen → Abfragen bearbeiten → Erweiterter Editor):

7.1 Intervalle in Tage splitten

```
TageProZeile = Table.AddColumn(Quelle, "Tage", each
    List.Dates([Von], Duration.Days([Bis]-[Von])+1, #duration(1,0,0,0))
),
Expandiert = Table.ExpandListColumn(TageProZeile, "Tage")
```

List.Dates(Start, Anzahl, Schrittweite) erzeugt die Liste der Kalendertage; Table.ExpandListColumn macht aus einer Zeile mit einer Liste von n Tagen n einzelne Zeilen. Das ist der Explode-Schritt, den eine SUMPRODUCT-Formel über Von/Bis-Spannen nicht sauber leisten kann, sobald sich Abwesenheiten überlappen.

7.2 Kalendertage von Arbeitstagen trennen

```
MitArbeitstag = Table.AddColumn(MitWochentag, "IstArbeitstag", each
    [Wochentag] < 5 and not List.Contains(FeiertagsListe, [Datum])
)
```

Date.DayOfWeek mit Montag als Referenztag liefert 0–6; Werte ab 5 sind Samstag/Sonntag. List.Contains gegen die Feiertagsliste übernimmt den zweiten Ausschluss. Ab hier stehen beide Nenner — Kalendertag und Arbeitstag — gleichzeitig zur Verfügung.

7.3 Monatsgrenzen

Kein separater Schritt nötig. Sobald jeder Tag seine eigene Zeile mit eigenem Datum ist, ordnet Date.ToText([Datum], "yyyy-MM") jeden Tag automatisch seinem echten Monat zu. Ein Intervall vom 28. Januar bis 6. Februar liefert 4 Tage für Januar und 6 Tage für Februar — korrekt, ohne Zusatzlogik für den Grenzfall.

8. Bedienung: Was passiert bei einer Änderung?

Änderung	Was passiert automatisch
Neue Zeile in Fehlzeiten_Rohdaten	Muss zusätzlich in Fehltage_Explodiert nachgezogen werden (bzw. per Power-Query-Refresh, wenn produktiv umgesetzt)
Abwesenheitsart als „zählt nicht“ umklassifizieren	Wirkt sofort in allen drei Quoten, für alle 18 Monate gleichzeitig
Neuer Mitarbeiter oder geänderte Vertragsstunden	Wirkt in der Sollstunden-Quote (Nenner) für alle Monate
Feiertag hinzufügen oder entfernen	Verändert Arbeitstage-Quote und Sollstunden-Quote, Kalendertage-Quote bleibt unverändert