

TA AbsenceEntryConnector

Der **TaAbsenceEntry Konnektor** überträgt alle **Fehlzeiten**, die in der Fehlzeitkartei abgelegt werden. Dies betrifft nur **ganztägige Fehlzeiten** und **Zeitkorrekturen in die Zukunft**.

Auf diese Weise werden Informationen geliefert, ob ein Mitarbeiter an einem Tag einen bestimmten Fehlgrund gebucht hat.

Die Fehlgründe lassen sich individuell konfigurieren, ebenso wie bestimmte Personen, Gruppen oder Mandanten definiert werden können.

Konfigurationsmöglichkeiten

- Zusätzliche Felder mit Benutzer-Informationen (z.B. Benutzer ID, Name, Org-Einheit etc.)
- Zeitraum: Tage in Zukunft, Tage in die Vergangenheit
- Genehmigte / Ungenehmigte Fehlzeiten etc.

Konfiguration von Konnektoren

Speichern Speichern & Schließen Zurück Löschen

Vaterkonnektor: taAbsenceEntryConnector
Name: Absence Entry Connector
Connector kann schreiben:

Tage in die Zukunft: 14
Tage in die Vergangenheit: 14
Genehmigte Fehlzeiten:
Ungenehmigte Fehlzeiten:
Pro Tag zumindest ein Eintrag:
Auch für Tage ohne Fehlzeit Eintrag erzeugen:

Felder

employeeId
Hinzufügen

Fehlgründe

Verfügbare Fehlgründe

- Abwesend (-1)
- Bundeswehr (13)
- Dienstgang (9)
- Dienstreise (8)
- Krank Lfz (4)
- Krank ohne Lfz (5)
- Kur (17)
- Mutterschutz (10)
- Pflegefreist. (12)
- Reise kurz (18)
- Reisezeit (90)

Ausgewählte Fehlgründe

- Anwesend (0)
- Arzt (11)
- Behördenweg (14)

Personen
Gruppen
Mandanten

Zur Verfügung gestellte Felder

Fields	Erklärung
employeeId	Personalnummer der Person, dessen Fehlzeit-Datensatz geliefert wird.
firstName	Vorname der Person
lastName	Nachname der Person
orgUnit	Organisationseinheit (orgService.getPersonsHierarchicalgroup().getShortName())
person	Personenobject. Hier wird das komplette PoPerson Objekt übergeben, sodaß mit Javascript dann noch auf die einzelnen Eigenschaften des Objekts zugegriffen werden kann.
absenceCode	Fehlgrundcode (ta_absencecode_fd_XXX)
absenceShortDescr	Fehlgrund-Kürzel (z.b. U, DR...)
absenceBookingType	Fehlgrundtyp aus TaCalendarTimeRangeEvent.bookingType
fromDate	von Datum als java.util.Date
toDate	bis Datum als java.util.Date (nur wenn Pro Tag = false)
fromTime	Zeit von als java.util.Date (wenn ganztägig dann leer!)
toTime	Zeit bis als java.util.Date (wenn ganztägig dann leer!)
approved	genehmigt
dateEmployeeId	Ein künstlicher 'eindeutiger' Schlüssel zusammengesetzt aus dem Tag in der Form ddMMyyyy und der Personalnummer der Person