



GILDE SAMENSTERKER

Cursus Teamdesk

LES 06

Relaties

Inhoudsopgave

| | |
|---|----|
| Relaties | 3 |
| Relationele database | 3 |
| Normaliseren | 3 |
| Een-op-veel relatie | 6 |
| Veel-op-Veel relatie..... | 7 |
| Een relatie vastleggen | 8 |
| Gebruik van de Record Picker | 12 |
| Conditie verbinden aan een relatie..... | 14 |
| Master-Detail in samenhang bekijken | 16 |
| Lookup kolommen..... | 17 |
| Summary kolommen | 18 |
| Huiswerk | 21 |

Relaties

Relationele database

De kracht van een relationele database zit 'm in de mogelijkheid om gegevens uit verschillende tabellen aan elkaar te kunnen relateren. Dat doe je door het aanbrengen van relaties. TeamDesk stelt de ontwerper in staat records van een tabel te relateren aan records van een andere tabel.

Het systeem veronderstelt twee hoofdtypen relaties tussen records:

- een-op-veel-relatie
- veel-op-veel-relatie.

Voordat we het over relaties in een Teamdesk database gaan hebben wil ik eerst even iets kwijt over het normalisatieproces.

Normaliseren

Databasennormalisatie is een techniek bij het ontwerpen van [databases](#). Ze dient twee doelen: het spaarzaam omgaan met [opslagruimte](#) en het vermijden van meervoudige vastlegging van dezelfde [data \(redundantie\)](#), een potentiële bron van fouten. Tegenwoordig speelt het argument van de opslagruimte geen doorslaggevende rol meer. Maar redundantie in de gegevens is de doodsteek voor elke database:

- **Update-anomalie:** Wanneer dezelfde gegevens op meerdere plekken worden opgeslagen dan zal dit bij een aanpassing ook op meerdere plaatsen moeten gebeuren. Als het adres van een klant in vijf tabellen opgeslagen is dan kan men dit bij een adreswijziging zomaar op vier plekken aanpassen en vergeten dat er nog een vijfde plek is.
- **Insert-anomalie:** Bij een slecht ontworpen database kan men tegen het probleem aanlopen dat een gegeven pas ingevoerd kan worden wanneer ook andere gegevens bekend zijn. Bijvoorbeeld, als de verkoopgegevens en de klantgegevens in dezelfde tabel staan dan kan een klant pas opgevoerd worden wanneer deze ook al iets heeft besteld. Omgekeerd geldt hetzelfde, een nieuw product kan pas opgevoerd worden als het al door iemand is besteld.
- **Verwijder-anomalie:** Dit probleem draagt het probleem van de vorige anomalie in zich. Het verwijderen van een product kan leiden tot het verwijderen van klanten.

Je kunt redundantie voorkomen door de gegevens te normaliseren (op te splitsen in verschillende tabellen). Data normalisatie omvat (tenminste) de volgende stappen:

0. Verwijder alle BEREKENDE velden
1. Vermijd HERHALENDE VELDEN binnen één tabel
2. Zet VELDEN DIE NIET GERELATEERD KUNNEN WORDEN AAN DE SLEUTEL in een afzonderlijke tabel
3. Beoordeel of een attribuut echt iets zegt over het gegeven dat door de sleutel wordt vertegenwoordigd (of gaat dat over iets anders?)

We gaan dit uitwerken in een kort voorbeeld van een factuur:

The screenshot shows an invoice from EnYoi Holding to AFAS Software B.V. The invoice number is PR008388, dated 23-07-2019, with a total amount of EUR 5,423.52. The invoice includes a table of items with columns for Code, Omschrijving, Levering, Aantal, Prijs/eenheid, and Bedrag. The total amount is 4,482.25.

1 EnYoi Holding contact information: EnYoi ICT Services B.V. TEST_Frederik, Philipsstraat 9, 3833 LC LEUSDEN. BTW: NL 123456782B01, KvK: 354652, IncassantID: NL 55 ZZZ 310468210000. IBAN: NL 15 ABNA 0412 5951 25, BIC: ABNANL2A.

2 Factuur details: Factuurnummer PR008388, Totaal EUR 5.423,52, Factuurdatum 23-07-2019, Vervaldatum 06-08-2019.

3 Invoice items table:

| Code | Omschrijving | Levering | Aantal | Prijs/eenheid | Bedrag |
|---------------|---|------------|--------|---------------|-----------------|
| 300 | Consultancy van Tom | 01-07-2019 | 8,00 | 95,00 | 760,00 |
| 300 | Consultancy van Floor | 04-07-2019 | 8,00 | 95,00 | 760,00 |
| 300 | Credit op uren van Tom | 01-07-2019 | -2,00 | 95,00 | -190,00 |
| 7001 | EnYoi Net Magazine van 2019-7-1 tot 2019-7-31 | 23-07-2019 | 1 | 8,25 | 8,25 |
| 7008 | EnYoi Glasvezel internet 120MB van 2019-7-1 tot 2019-7-23-07-2019 | 23-07-2019 | 24 | 93,50 | 2.244,00 |
| 7020 | Garantiecontract van 2019-7-1 tot 2019-7-31 | 23-07-2019 | 1 | 900,00 | 900,00 |
| Totaal | | | | | 4.482,25 |

4 Customer information: AFAS Software B.V., T.a.v. dhr B.B. Maas, Postbus 310, 3830 AJ LEUSDEN.

Welke (normale) velden zouden we in de tabel **FACTUUR** moeten opnemen? Nou, in elk geval niet de **vaste waarden (1)** die op ELKE factuur hetzelfde zijn, zoals de naam en het adres van het bedrijf of de voorwaarden. Ook zogenaamde procesgegevens (waarden die berekend kunnen worden uit andere gegevens) laat je (in eerste instantie) weg (2). Herhalende gegevens, zoals artikelcode, omschrijving, leverdatum, prijs en aantal, komen ook NIET in de factuurtabel (3), maar in een daaraan gerelateerde tabel **FACTUURREGEL**. En tenslotte zeggen de Naam en het adres van de klant niets over de factuur, maar over de klant. Ook die gaan we apart zetten in de tabel **KLANT (4)**.

Omdat we wel een relatie naar de klant nodig hebben voegen we het **Klant ID** (het volgnummer dat een unieke klant identificeert) aan de factuurtabel toe.

De genormaliseerde tabel **FACTUUR** bevat dus de velden Factuurnummer, Factuurdatum, Vervaldatum en Klant ID.

We gaan nog even verder: De tabel **FACTUUREGEL** omvat de velden Code, Omschrijving, Leverdatum, Aantal en Prijs. Maar omdat we ook een relatie naar de factuur nodig hebben, voegen we daar FACTUUR ID (het automatisch aan een Factuur toegekende volgnummer) aan toe.

De tabel **KLANT** zou dan bestaan uit de velden Naam, Straat, Huisnr, Postcode, Plaats en Contactpersoon (en vermoedelijk nog zaken die we niet op de factuur terugzien, zoals als email, telefoon, website, etc.).

Bij een 'normale' database zouden we er nu zijn. Zaken als 'Totaalbedrag' worden 'on-the-fly' berekend in een zgn. SELECT-statement. Maar niet bij Teamdesk. Aan de tabel **FACTUUREGEL** voegen we dus een FORMULE-VELD toe genaamd **Totaalbedrag**. De formule luidt: [Aantal]*[Prijs]. En in de tabel factuur voegen we een SUMMARY-VELD (uitleg volgt later) toe, waarin we alle Totaal bedragen uit de onderliggende Factuurregels optellen.

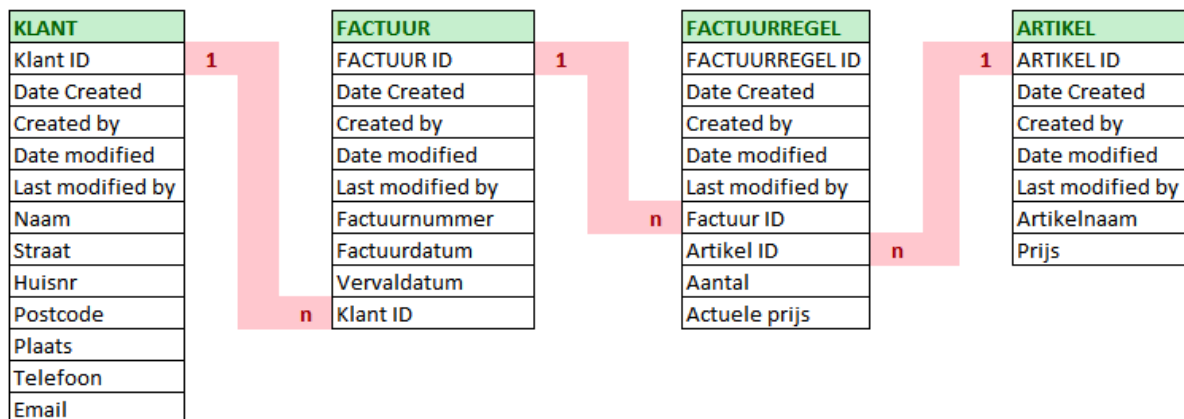
In een real life situatie is meestal een stuk complexer dan wat we nu gedaan hebben. Zo zou er ook een tabel **ARTIKEL** kunnen zijn, waarin elk artikel een unieke sleutel heeft. Dáár zou je dan zaken als de naam van het artikel, de eenheid (stuks, kilogram, liter, etc.) en de prijs per eenheid inzetten, in plaats van in de factuurregel. Maar in het voorbeeld heeft een artikel geen unieke code, wat dat dus lastig maakt....

Ook de vervaldatum zou een berekende waarde kunnen zijn (14 dagen na de factuurdatum). En in dit voorbeeld wordt geen BTW berekend en is er ook geen sprake van kortingen. Maar hopelijk begrijpen jullie het principe van normalisatie nu een beetje. Dit was overigens wel een crash-cursus normaliseren. Ik heb er zelf ongeveer 1,5 jaar over gedaan om dit in de vingers te krijgen.....

Een-op-veel relatie

We hebben een database met twee tabellen gemaakt. De eerste heet Facturen. Deze tabel bevat informatie over alle facturen die we hebben verstuurd. De tweede tabel heet Klanten. Daar slaan we alle gegevens van onze klanten in op. Er zijn klanten waar we méér dan één factuur aan hebben gestuurd, maar een factuur kan aan slechts één klant worden gericht.

En dus maken we een relatie tussen deze tabellen om facturen en klanten te koppelen. Records zijn gerelateerd aan de waarden van hun Id-kolommen. De relatie maakt selectie mogelijk van een klantrecord (Klant ID) voor elk factuurrecord (Project ID). Als gevolg hiervan worden dan bijvoorbeeld de namen en adressen van klanten weergegeven in de tabel Facturen, zonder dat ze feitelijk deel uitmaken van die tabel. In onderstaand voorbeeld zie je vier tabellen en drie één-op-veel (1:n) relaties:



Ik heb eerder al laten zien dat elke tabel een ID-kolom heeft. Deze wordt (bijna) ALTIJD gebruikt als de primaire sleutel omdat het verwijst naar een uniek record. Vanuit een andere tabel verwijst je dan naar dat ID. Aan de n-kant heet dat dan de secundaire sleutel.

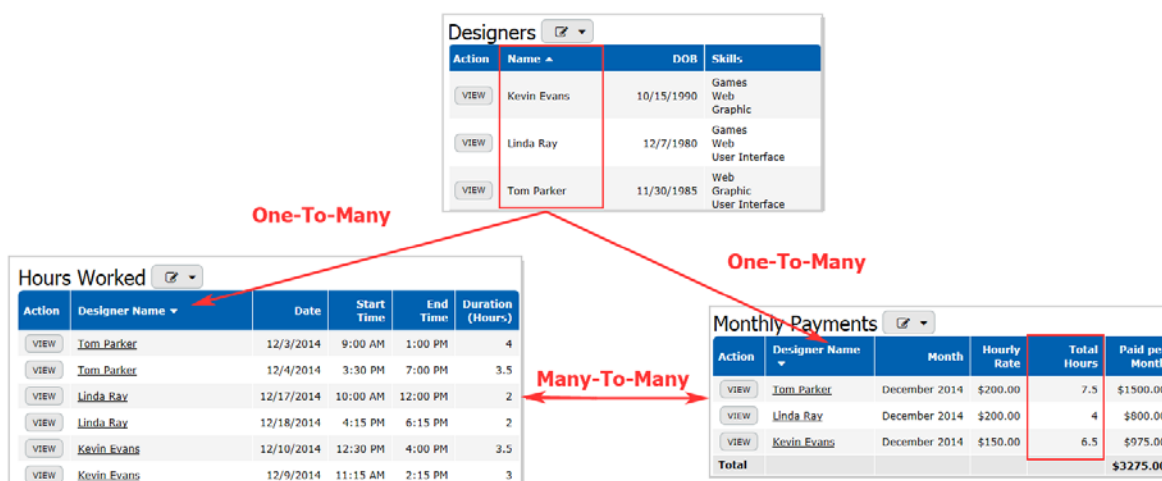
Heel soms kun je in plaats van het ID ook een ander veld als primaire sleutel gebruiken, zoals bijvoorbeeld een ISBN-nummer voor een boek of een BSN voor een persoon. Maar die laatste optie is meestal vanuit de AVG niet toegestaan. Ik doe het zelf NOOIT, want hoewel het in uitzonderlijke gevallen zou kunnen, is er ook geen enkele reden te bedenken om niet het ID als primaire sleutel van de tabel te gebruiken.

Elke één-op-veel relatie bestaat uit een MASTER (1) en een DETAIL (n) tabel. Via de relatie kun je gegevens uit de DETAIL tabel zien vanuit de MASTER en vice versa.

Veel-op-Veel relatie

Dit type relatie veronderstelt een veel-op-veel (n:m) relatie tussen records in twee tabellen. Veel-op-veel relaties worden niet heel vaak gebruikt, maar er zijn situaties denkbaar waarin het nuttig is.

Een voorbeeld: We hebben ontwerpers in dienst, die elke dag hun gewerkte uren registreren. We betalen onze ontwerpers elke maand uit op basis van de in die maand gewerkte uren.



In de tabel Designer houden we de profielen bij van onze ontwerpers. De tabel Hours worked omvat het logboek van de gewerkte uren per ontwerper. En de derde tabel (Monthly payments) worden de aan de ontwerpers uitbetaalde bedragen per maand opgeslagen. Zowel in de tabel Monthly Payments als in de tabel Hours Worked wordt gerefereerd aan een Designer (2x 1:n relatie). In zo'n dit geval kan een n:m relatie tussen de twee onderliggende tabellen worden gemaakt. Als resultaat kan dan in de tabel Monthly payments een summary kolom (Total Hours) worden opgenomen die het totaal aantal gewerkte uren in die maand voor die ontwerper berekent (en samen met het uurtarief het uit te betalen bedrag bepaalt).

Een relatie vastleggen

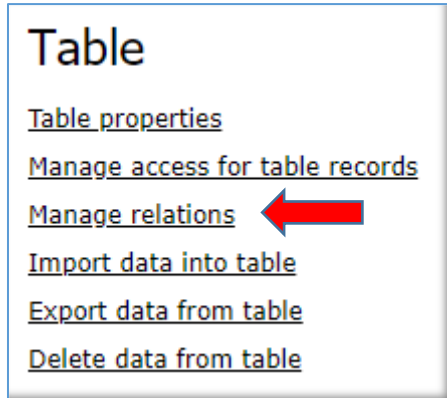
Stel: We hebben een tabel met klanten (Cliënts) en een andere tabel met Projecten. Een project hoort altijd bij één klant, terwijl een klant meerdere projecten kan hebben. Zoiets dus:

| Klant |
|------------------|
| ID |
| Date Created |
| Created by |
| Date modified |
| Last modified by |
| Naam |
| Adres |
| Telefoon |
| Email |

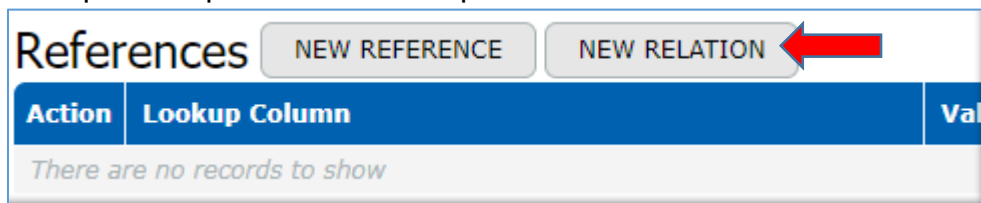
| Project |
|------------------|
| ID |
| Date Created |
| Created by |
| Date modified |
| Last modified by |
| Klant ID |
| Projectleider |
| Projectnaam |
| Begroot bedrag |

We willen nu een relatie gaan aanleggen tussen de klant en de bijbehorende projecten. Dat gaat als volgt:

- Klik op de link **SETUP** in de rechterbovenhoek van het venster.
- Selecteer de **Projecten** tabel.
- Selecteer in het setup-menu Table → **Manage Relations**



- Klik op de knop **New Relation** op het formulier Relaties.



Selecteer de tabellen die een rol spelen in de te vormen relatie. In dit geval dus Klanten en Projecten en druk op tussen in te stellen en

New Relation

Use the form below to create or modify the relationship.

CANCEL NEXT

Select the two tables you want to establish a relationship

First Table Second Table

CANCEL NEXT

klik op de knop NEXT. Je ziet nu dit:

New Relation

Use the form below to create or modify the relationship.

CANCEL PREVIOUS NEXT

Select the relationship direction

| | | | | |
|--------------------------|---------|--|---------|---|
| <input type="checkbox"/> | Project | | Klant | Each Klant may be related to a single Project , while each Project can be related to multiple Klanten . |
| <input type="checkbox"/> | Klant | | Project | Each Project may be related to a single Klant , while each Klant can be related to multiple Projecten . |
| <input type="checkbox"/> | Project | | Klant | Each Project may be related to multiple Klant , while each Klant can be related to multiple Projecten . |

Bepaal hoe de relatie in elkaar zit:

1. Aan elk project kun je meerdere klanten koppelen. Een klant kan met maar één project in verband worden gebracht.
2. Een klant kan meerdere projecten hebben. Een project kan maar aan één project worden gekoppeld.
3. Aan een project kunnen meerdere klanten hangen, terwijl ook aan een klant meerdere projecten kunnen kleven.

In dit geval kies je dus optie 2. Druk op NEXT.

Vul een naam en een omschrijving in voor de relatie. Dit is optioneel. Je kunt het dus leeg laten, maar het helpt je wel overzicht te houden....

Master-Detail Relation

| | |
|--------------|--|
| Name | Klant <=>> Projecten |
| Notes | Een klant kan meerdere projecten hebben lopen, maar elk project verwijst naar slechts één klant. |

Vul in de sectie Master Table (in dit geval Klant) een **Proxy column** in. Dat is het veld dat in plaats van het (Klant) ID in het Projectenformulier zal worden getoond. We kiezen hier de **Naam** van de klant. Bij Record Picker heb je een aantal keuzen. We laten deze voor nu op de standaardwaarde (simple choices) staan. Maar we zetten wel het vinkje aan bij **Override default record picker**.

Master Table

Table Klant

Key Column Id

Proxy Column Naam

Record Picker Simple Choices
 Dropdown with 'Browse Choices' option
 Type-in with 'Search' button
 Radio Buttons
 Override default record picker

We stellen voor nu geen filter in, want we willen kunnen kiezen uit alle klanten. Maar stel dat we alleen projecten zouden doen voor klanten uit Brabant, dan zou dat een filtervoorwaarde kunnen zijn. Maar we zorgen er wel voor dat we een lijst met namen en plaatsen te zien krijgen (sommige bedrijven hebben meerdere vestigingen, die als afzonderlijke klanten in de tabel staan).

Record Picker Options

Filter All records Only records that meet certain criteria Custom formula

Ask the User Nothing Ask columns

Sorting Default sorting Custom sorting

Columns Name Column Custom columns

Search Available Columns

- Created By
- Date Created
- Date Modified
- email
- Huisnr
- Id
- Last Modified By
- Record Owner
- Straat
- Telefoon

Selected Columns

- Naam
- Plaats

ADD REMOVE

TOP UP DOWN BOTTOM

Kies onder het kopje Detail Table de referentie kolom (de kolom die wijst naar de klanten tabel), in dit geval dus **Klant ID**. Kies als Details View in eerste instantie altijd voor List All. Onder elke klant zie je dan een lijst met projecten, die we voor deze klant uitvoeren. Je kunt eventueel ook een naam voor deze view kiezen (in plaats van de naam van de tabel). In dit geval zou je bijvoorbeeld als alternatieve naam kunnen typen “projecten voor deze klant”).

Detail Table

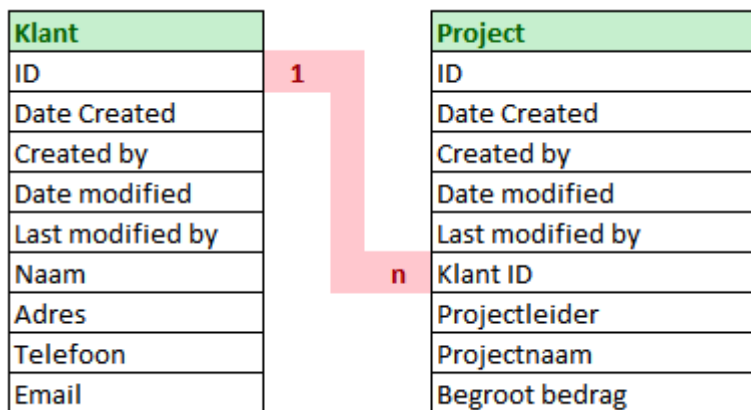
Table Project

Reference Column Klant ID ▼

Details View List All ▼

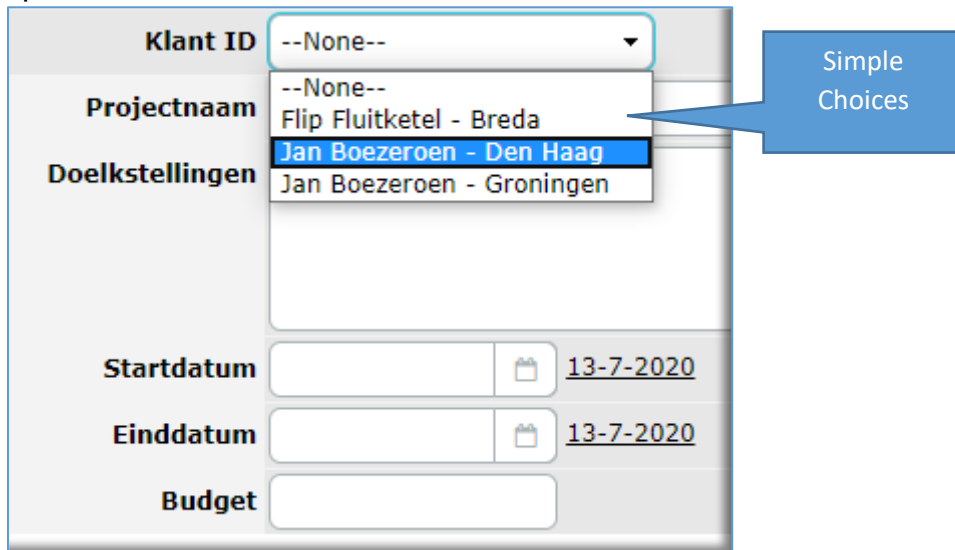
Alternate Name

Druk op nu SAVE en verlaat de SETUP modus. We hebben nu deze relatie gemaakt:



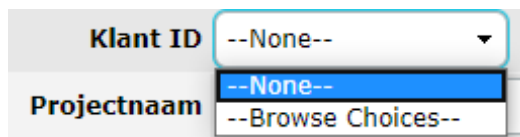
Gebruik van de Record Picker

We hadden bij Record Picker gekozen voor **Simple Choices**. Je kunt in deze situatie niet even snel een nieuwe klant opvoeren. Je MOET kiezen uit de beschikbare opties:



The screenshot shows a form with several fields. The 'Klant ID' field is a dropdown menu currently showing '--None--'. A blue callout box labeled 'Simple Choices' points to the dropdown menu. The dropdown menu is open, showing three options: '--None--', 'Flip Fluitketel - Breda', and 'Jan Boezeroen - Den Haag' (which is highlighted in blue). Below the dropdown menu, there are fields for 'Projectnaam', 'Doelstellingen', 'Startdatum', 'Einddatum', and 'Budget'. The 'Startdatum' and 'Einddatum' fields are date pickers showing '13-7-2020'.

Een andere aanpak zou kunnen zijn om te kiezen voor één van de andere opties bij Record Picker in de relatie, bijvoorbeeld **Dropdown with 'Browse Choices'**.



The screenshot shows a form with two fields: 'Klant ID' and 'Projectnaam'. The 'Klant ID' field is a dropdown menu currently showing '--None--'. The 'Projectnaam' field is also a dropdown menu currently showing '--None--'. A blue callout box labeled 'Browse Choices' points to the 'Projectnaam' dropdown menu. The dropdown menu is open, showing two options: '--None--' and '--Browse Choices--'.

Je krijgt dan een dropdown te zien, maar daarin kun je niks invullen. Je moet dan eerst op **Browse choices** klikken.

Dan zie je dit:

Select a Klant

Select from the list below, or search to see a subset of Klanten.

SEARCH

Search |

SEARCH

Klanten (+)

| Select | Naam | Plaats |
|--------|-----------------|-----------|
| SELECT | Flip Fluitketel | Breda |
| SELECT | Jan Boezeroen | Den Haag |
| SELECT | Jan Boezeroen | Groningen |

Je gebruikt dit alleen in situaties, waarin je honderden klant hebt, omdat je hier wat beter kunt zoeken en je kunt hier WEL een klant toevoegen, terwijl je met een project bezig bent. (+). Persoonlijk ben ik niet zo kapot van deze oplossing en ik kies ik altijd voor **Type-in with 'Search' button**. Dat geeft dezelfde mogelijkheden, maar werkt wat vlotter:

Klant ID

Projectnaam

Doelstellingen

Startdatum 13-7-2020

Einddatum 13-7-2020

Budget

Je kunt hier een (deel van de) naam typen en/of op de zoekknop (vergrootglas) drukken. Dat scheelt dus één klik.

Je zou ook kunnen kiezen voor Radio Buttons, maar dat werkt alleen goed als je maar enkele klanten hebt:

Klant ID Flip Fluitketel - Breda
 Jan Boezeroen - Den Haag
 Jan Boezeroen - Groningen

Zijn er méér klanten, dan wordt dit al gauw een zootje.

Conditie verbinden aan een relatie

We hadden het eerder over klanten uit Brabant. Dat is makkelijk met een Filtervoorwaarde in de Record Picker op te lossen. Maar ook als het wat complexer wordt, laat Teamdesk je niet in de kou staan. Stel dat een project gekoppeld is aan een bepaalde provincie en je alleen klanten uit die provincie wilt kunnen selecteren. Dit veronderstelt een extra veld **Provincie** in zowel de tabel Projecten als in de tabel Klanten. Daarvoor moet je de bestaande relatie aanpassen. Onder de eerder ingevulde gegevens zie je dit:

Master-Detail Relation

| | |
|-------|--|
| Name | |
| Notes | |

Master Table

| | |
|---------------|--------------------------------|
| Table | Klant |
| Key Column | Id |
| Proxy Column | Naam |
| Record Picker | Simple Choices |
| | Override default record picker |

Detail Table

| | |
|------------------|--------------------------|
| Table | Project |
| Reference Column | Klant_ID |
| Details View | List All |
| Alternate Name | |

Record Picker Options

| | |
|--------------|---|
| Filter | All records |
| Ask the User | Nothing |
| Sorting | Sort by Naam order by Ascending and then by Plaats order by Ascending |
| Columns | Naam , Plaats |

EDIT DELETE BACK

Match Conditions

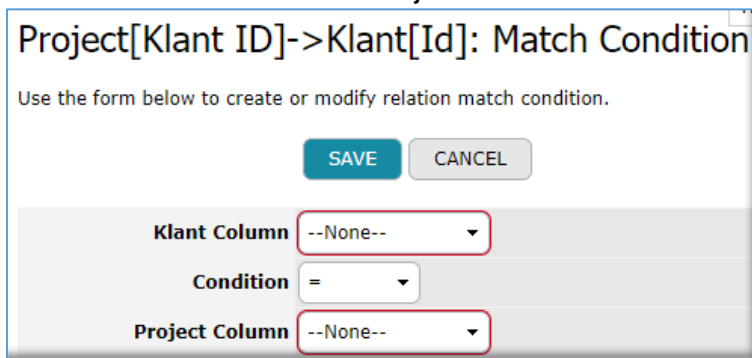
| Action | Klant Column | Condition | Project Column |
|--------|--------------|-----------|----------------|
|--------|--------------|-----------|----------------|

Druk in de sectie Match Conditions op NEW:

Match Conditions

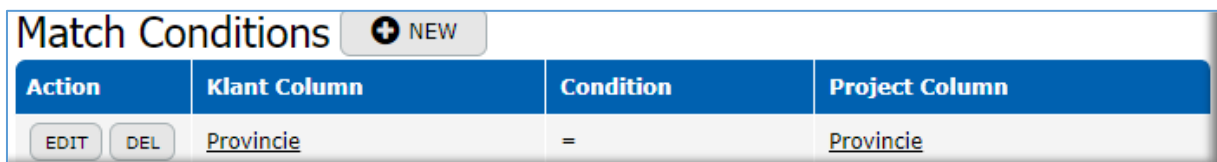
| Action | Klant Column | Condition | Project Column |
|------------------------------|--------------|-----------|----------------|
| There are no records to show | | | |

Je ziet dan een formulier waarin je een vergelijking kan maken tussen een kolom in KLANT en een andere in Project:



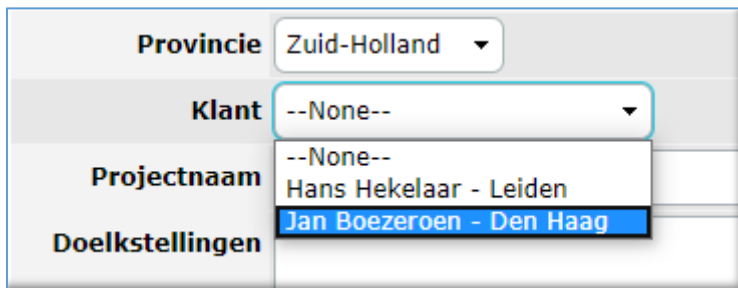
Vul hier zowel bij de klant als het project het veld **Provincie** in. De conditie hou je op '='. Druk op Save.

Dan is dit het resultaat:



| Action | Klant Column | Condition | Project Column |
|----------|--------------|-----------|----------------|
| EDIT DEL | Provincie | = | Provincie |

Om deze functionaliteit goed te lasten werken moet je nog wel het formulier PROJECT zo vormgeven dat EERST een provincie moet worden ingevuld, voordat de gebruiker een klant selecteert. Maar daarna werkt het zo, dat als je bijvoorbeeld 'Zuid-Holland' als provincie hebt gekozen, je alleen maar klanten te zien krijgt die ook in Zuid-Holland zijn gevestigd.



Het zou nu wel kunnen dat je een provincie selecteert, waarin geen klanten zitten ☺. Dan valt er dus ook niks te kiezen. Gebruik deze truc dus met verstand!

Master-Detail in samenhang bekijken

Onder een formulier van een MASTER tabel kun je één of meer tabbladen zien met de gegevens van verschillende DETAIL tabellen.

Team: Onze Helden 1

EDIT ADD SIMILAR DELETE BACK

| Club ID | Team | Divisie ID | Aantal wedstrijden |
|-----------------|------------------|------------|--------------------|
| Onze Helden | Onze Helden 1 | Eredivisie | 3 |
| Doelpunten Voór | Doelpunten Tegen | Doelsaldo | Punten Totaal |
| 7 | 0 | 7 | 9 |

System Columns

Id 1
Date Created 29-7-2019 11:17:19
Date Modified 25-2-2020 10:35:31
Record Owner Ton Jense
Created By Ton Jense
Last Modified By Ton Jense

EDIT ADD SIMILAR DELETE BACK

TEAMLEDEN THUJSWEDSTRIJDEN UITWEDSTRIJDEN

Teamleden NEW PRINT EXPORT SEND CUSTOMIZE

| Action | Pasfoto | Naam | Adres | Telefoon | Mobiel | Email |
|-------------------|---------|---------------------------|--------------------------------|-------------|--------|--------------------------------------|
| Buiten Zoetermeer | | Bronckhorst, Giovanni van | Floralaan 32, 2643HC Pijnacker | 079-3213952 | | BronckhorstvanGiovanni@onzehelden.nl |

2 records

Dat werkt overigens alleen in de VIEW modus, niet in de EDIT stand. Onder elk van de tabbladen vind je een view met gegevens uit de DETAIL tabel. Welke view hier wordt getoond wordt bepaald bij het configureren van de relatie:

Detail Table

Table [Lid](#)

Reference Column [Team ID](#)

Details View [Alle leden](#)

Alternate Name Teamleden

Het label van een tabblad is standaard de naam van de detail tabel. Dit kun je veranderen door gebruik te maken van een alternatieve naam:

Detail Table

Table [Lid](#)

Reference Column [Team ID](#)

Details View [Alle leden](#)

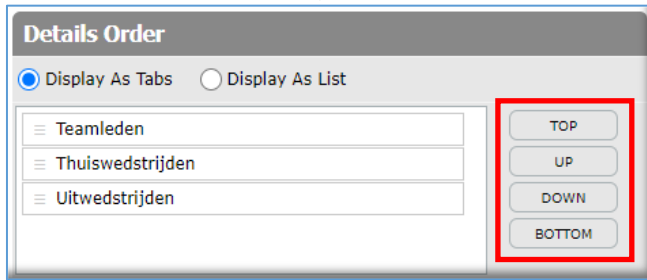
Alternate Name [Teamleden](#)

Als er maar één view kan worden getoond, wordt deze weergegeven zonder tabbladen. En je kunt views ook onder elkaar plaatsen i.p.v. in tabbladen. Dat doe je achter de knop 'Order details' in de configuratie van de relatie:

Details Order

Display As Tabs Display As List

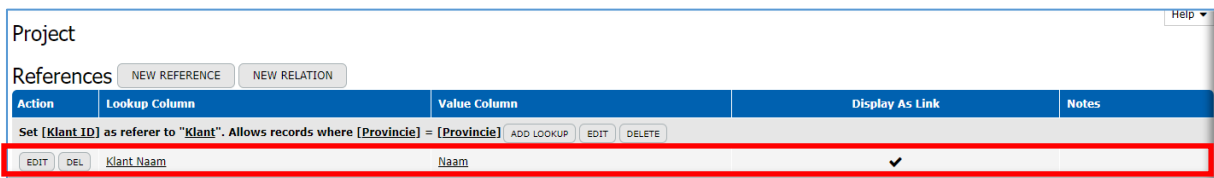
Hier kun je ook de volgorde van de views bepalen, waarin de views worden getoond:



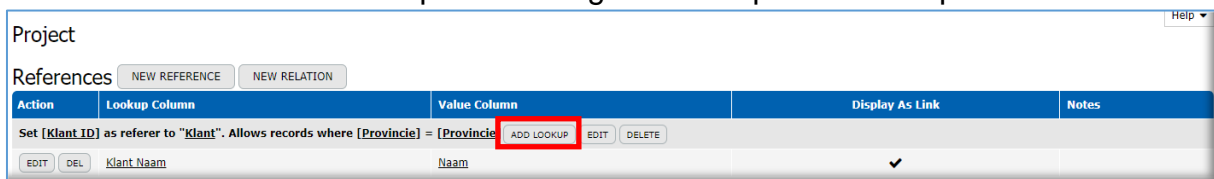
Wat de gebruiker met de gegevens in de detail tabel kan doen is afhankelijk van de ROL en de instellingen in de gebruikte VIEW. Als je een view gebruikt, waarin geen delete knoppen zijn opgenomen, kun je detailrecords hier niet verwijderen, zelfs niet als je dat vanuit je rol wel zou mogen.

Lookup kolommen

Een lookup kolom wordt bijvoorbeeld gebruikt in de tabel Projecten om de naam van de klant uit de tabel Cliënten op te halen. Wanneer een 1:n relatie wordt gemaakt, wordt de referentie kolom van de Master automatisch toegevoegd als lookup aan de tabel Details:



Maar stel nu dat we ook het adres van de klant in een projectformulier zouden willen zien? Dat kan door een lookup toe te voegen. Druk op 'Add Lookup':



Je ziet dan een invulformulier:

Project: Adres - Lookup to Provincie from Klanten

Use the form below to modify the column.

SAVE CANCEL

General

Name Adres

Notes Het adres van de klant

Display Include in views by default
 Display in bold
 Display without wrapping

Access Restrict access right by role

Lookup Options

Table Klant Relation Project[Klant ID]->Klant[Id]

Value Column Adres

Display As Link

Help

Kies hier het (formule)veld 'Adres' vanuit de tabel KLANT. Haal het vinkje weg bij 'Display As Link' (het is niet logisch dat klikken op een adres leidt tot de gegevens van een klant) en bepaal bij 'Display' of het adres standaard moet worden opgenomen in elke nieuwe view die je maakt. Druk op SAVE, en geef het nieuwe veld een plekje in het PROJECT formulier (via Customize form layout).

Standaard zijn formule- en lookupvelden NIET zichtbaar in de EDITmodus. Maar je kunt dat - als je dat wilt - aanpassen via Customize form behavior.

Summary kolommen

Waar een Lookup gegevens ophaalt van de MASTER, brengt een Summary kolom gegevens vanuit DETAILS (maar wel geaggregeerd) naar de MASTER. Zo kun je bijvoorbeeld de budgetten van projecten bij elkaar optellen per klant.

List All NEW PRINT EXPORT SEND CUSTOMIZE

| Action | Naam | Adres | Budget |
|--|-----------------|------------------------------------|-------------|
| VIEW EDIT DEL | Jan Boezeroen | Van Beverenstraat 145, Groningen | €0,00 |
| VIEW EDIT DEL | Flip Fluitketel | Tilburgsestraatweg 7, Breda | €40.000,00 |
| VIEW EDIT DEL | Jan Boezeroen | Noorderbeekdwarsstraat 8, Den Haag | €120.000,00 |
| VIEW EDIT DEL | Hans Hekelaar | Jojusteeg 1, Leiden | €0,00 |

Displaying 4 records.

Dit hebben we voor elkaar gekregen door een Summary veld op te nemen in de relatie:

Klant

References

| Action | Lookup Column | Value Column | Display As Link | Notes |
|------------------------------|---------------|--------------|-----------------|-------|
| There are no records to show | | | | |

Selections

| Action | Summary Column | Function | Filter | Notes |
|--|---|-----------------|-------------|-------|
| Select "Projecten" where [Id] = [Klant ID] and [Provincie] = [Provincie]. Display details using "List All" view <input type="button" value="ADD SUMMARY"/> <input type="button" value="EDIT"/> <input type="button" value="DELETE"/> | | | | |
| <input type="button" value="EDIT"/> | <input type="button" value="DEL"/> Budget | Total of Budget | All records | |

Stel nu dat we ook het aantal projecten terug willen zien in een klantformulier? Dat kan door een summary toe te voegen. Ga naar de relatie en druk op 'Add Summary':

Klant

References

| Action | Lookup Column | Value Column | Display As Link | Notes |
|------------------------------|---------------|--------------|-----------------|-------|
| There are no records to show | | | | |

Selections

| Action | Summary Column | Function | Filter | Notes |
|--|---|-----------------|-------------|-------|
| Select "Projecten" where [Id] = [Klant ID] and [Provincie] = [Provincie]. Display details using "List All" view <input type="button" value="ADD SUMMARY"/> <input type="button" value="EDIT"/> <input type="button" value="DELETE"/> | | | | |
| <input type="button" value="EDIT"/> | <input type="button" value="DEL"/> Budget | Total of Budget | All records | |

Je ziet dan een invulscherm:

The screenshot shows a configuration window with two main sections: 'General' and 'Aggregate Options'.
In the 'General' section:
- Name: New Summary
- Notes: (empty text area)
- Display: Include in views by default, Display in bold, Display without wrapping
- Access: Restrict access right by role
In the 'Aggregate Options' section:
- Table: Project
- Relation: Project[Klant ID]->Klant[Id]
- Aggregate Function: # of records (dropdown menu is open showing options: # of records, Total, Average, Minimum, Maximum, Concatenate, Index, # of unique values, Standard Deviation, Population Standard Deviation, Variance, Population Variance)
- Filter: All records, Only records that meet certain criteria, Custom formula
- Buttons: SAVE, CANCEL

Geef het nieuwe summaryveld een naam en beschrijf het doel. Waarschijnlijk wil je het NIET standaard in elke klantenview terugzien. Zet het vinkje bij 'Include in views by default' dan uit. Bij Aggregate kies je (in dit geval) # of records.

Zou je hier iets anders dan # of records hebben gebruikt, dan moet je ook een Value Column invullen, zoals bijvoorbeeld bij het totaal aan budget:

The screenshot shows the 'Aggregate Options' section with:
- Table: Project
- Relation: Project[Klant ID]->Klant[Id]
- Aggregate Function: Total
- Value Column: Budget (highlighted with a red box)
- Filter: All records, Only records that meet certain criteria, Custom formula

Een filter is bij een summaryveld zelden nodig, omdat de functie automatisch rekening houdt met de restricties in de relatie. Hij telt dus altijd het aantal projecten of het totaal budget **per klant**. Maar heel soms wil je bijvoorbeeld twee summaries maken: Eén met het aantal reeds afgeronde projecten en nòg één voor het aantal nog lopende projecten. In zo'n geval gebruik je dus wèl een filter:

The screenshot shows the filter configuration with:
- Filter: All records, Only records that meet certain criteria, Custom formula
- 1: Einddatum is <blank>

en

The screenshot shows the filter configuration with:
- Filter: All records, Only records that meet certain criteria, Custom formula
- 1: Einddatum is not <blank>

Huiswerk

Verzin nog even wat.....