

Portaltegel

Applicatiebouw

Screenidentifiers: MDLC_getPortalTilesList.xml en MDDC_getPortalTilesDetail.xml

Betekenis van de kolommen

- De tekst van de kolom **kopregel** (dvtileheadertext) komt bovenaan in de tegel in blauw. Mag zonder gevaar aangepast worden. Mag Leeg zijn. Puntkomma's in deze tekst veroorzaken een nieuwe regel (wagenterugloop). De tekst insluiten in rechte vierkante haken ([]) betekent dat de tekst vet wordt afgedrukt.
- De tekst van de kolom **vast opschrift** (dvcontent) komt onder de kopregel in de tegel in zwart. Mag zonder gevaar aangepast worden. Mag Leeg zijn. Puntkomma's in deze tekst veroorzaken een nieuwe regel (wagenterugloop). De tekst insluiten in rechte vierkante haken ([]) betekent dat de tekst vet wordt afgedrukt.
- De achterliggende waarde van de kolom **Tegelopschrift dynamisch met API** **getTileContent** of **getFlexChart** (dvgettilecontent) komt uit een geëvalueerd SQL-statement en wordt geplaatst onder het vaste opschrift in de tegel. Indien deze kolom dvgettilecontent wordt gevuld met de aanroep `getTileContent(querynaam)` dan wordt de geëvalueerde query als tekst getoond. Indien deze kolom dvgettilecontent wordt gevuld met de aanroep `getFlexChart(type,querynaam)` dan wordt de geëvalueerde query omgezet naar een staaf-, of taart- of ander diagram). Mag met kennis van zaken aangepast worden. Mag Leeg zijn.
 - Toepassing met `getFlexChart`: zie [Grafieken](#)
 - Toepassing met `getTileContent` is bijvoorbeeld om aan te geven hoeveel kaarten de lijst achter de tegel bevat. De inhoud van deze kolom is altijd in de vorm van: `getTileContent(codering)` OF `getTileContent(codering,{id})` bijvoorbeeld: `getTileContent(omgeving_chklist,{id})`. Bij tegels op zaakportalen moet de tweede vorm gebruikt worden, waarbij dynamisch {id} zal worden vervangen door de primary key van het betreffende zaakportaal (de identifier die opgenomen is in de URL van de portaalpagina).
De eerste parameter - de codering (in het voorbeeld: `omgeving_chklist`,) - verwijst naar een kaart in de tabel `tbqueries` met dezelfde codering. Het resultaat van die query wordt dus op de tegel geplaatst. Indien de gebruiker geen rechten heeft op de query of indien de query niet goed is gedefinieerd, dan blijft het dynamische opschrift op de tegel leeg of er komt te staan: `fout.xml`. Zie ook bij [Queries](#).
Toevoeging: in tegelopschrift indien er in de naam van de query haakjes voorkomen dan kan het programma deze niet interpreteren. Dit is als volgt op te lossen: stel de naam van de query is `test(kopie)` dan zal het veld tegelopschrift gevuld moeten worden met:
`getTileContent(beginarg(test(kopie))endarg)`
Zie voor voorbeeld gebruik HTML ook kopje *queries tegelopschrift* bij [Queries](#)
- De kolom **volgorde** bepaalt de verticale positie van een tegel onder een kolom.
- De kolom **hoort bij kolom**. Zie hieronder bij verplaatsen van tegels.
- Met het **aanvinkvakje actief** wordt de tegel klikbaar voor de gebruikers. Niet actief betekent wel zichtbaar (mits toegekend aan de medewerkers), maar disabled. Let op: indien de kolom `sql_tegel_onzichtbaar` gevuld is, dan kan deze waarde overruled worden.
- De kolom **altijd verversen** betekent dat bij het herstel van de focus op een portaal waar de tegel bij hoort, de inhoud van deze tegel opnieuw wordt uitgeschreven.

- De inhoud van de kolom **actie** geeft aan welke API of hyperlink met het indrukken van de tegel wordt aangeroepen en met welke parameters. Zie hieronder bij hoofdstukje mogelijkheden. Er kunnen in de parameterlist variabelen gebruikt worden: (niet alle onderstaande variabelen kunnen in alle situaties gebruikt worden):
 - de variabele {id} wordt dynamisch vervangen door de primary key van het zaakportaal (de identifier die opgenomen is in de URL van de portaalpagina) waar de tegel onder valt
 - de variabele {inlogger} door de dvcode van de medewerker die ingelogd is
 - de variabele {locatieid} door de primary key van tbperceeladressen (de locatie waar een zaak aan is gekoppeld)
 - de variabele {inrichtingid} door de primary key van een tbmilinrichting.
- Met de kolom **sql tegel onzichtbaar indien result = 0** kan aangegeven worden onder welke conditie een tegel zichtbaar en enabled is, of zichtbaar maar disabled of onzichtbaar. Bijvoorbeeld: door het onderstaande statement is de tegel proceschecklijsten alleen zichtbaar bij de omgevingszaak indien er checklijstitems zijn gekoppeld aan een procedure (van die omgevingszaak) en anders niet:
In de kolom kan hiertoe een SQL-statement worden opgenomen. Het statement moet beginnen met 'select' en er mag geen puntkomma in het statement voorkomen. De variabele {id} en {inlogger} wordt dynamisch vervangen met de primary key van het zaakportaal (de identifier die opgenomen is in de URL van de portaalpagina) en de dvcode van de ingelogde medewerker waar de tegel onder valt.
Indien het resultaat:
 - is null of het SQL-statements niet goed of de uitkomst <> 0,1 of 2 dan:
 - zichtbaar
 - attribuut enabled krijgt de waarde van kolom *aanvinkvakje actief* (dlenabled).
 - is 0 dan:
 - tegel is onzichtbaar
 - attribuut enabled doet niet ter zake, maar krijgt de waarde van kolom *aanvinkvakje actief* (dlenabled).
 - is 1 dan:
 - tegel is zichtbaar
 - attribuut enabled krijgt de waarde van kolom *aanvinkvakje actief* (dlenabled).
 - is 2 dan:
 - tegel is zichtbaar
 - attribuut enabled krijgt de waarde false ongeacht de waarde van kolom *aanvinkvakje actief* (dlenabled).

voorbeeld zichtbaarheid query

De tegel is zichtbaar inbden er een cchecklijst is gedefinieerd bij de zaak:

```
select count(1) from tbchkitwerk
  where dnkeyomgvergunningen = {id}
  and dnkeyprocedure is not null;
```

Of bijvoorbeeld de tegel mag niet zichtbaar zijn bij een bepaalde zaaktype:

```
select case when dnkeysoortomgverg = 134
  then 0 else 1 end
  from tbomgvergunning where dnkey = {id};
```

Verplaatsen van tegels

Het verplaatsen van tegels van de ene kolom naar de andere kolom kan zonder gevaar. De eenvoudigste manier is om op het detailscherm van de tegel de waarde van de keuzelijst *hoort bij kolom* aan te passen. Die keuzelijst bevat alleen de kolomnamen die bij hetzelfde portaal zijn gedefinieerd: men kan dus geen tegel van portaal A verplaatsen naar portaal B.

Triggers linsonder in scherm

- **ken medewerkers toe aan tegel** , met de wizard achter deze knop kan geregeld worden dat de tegel zichtbaar of onzichtbaar wordt voor de toegekende medewerkers. De wizard spreekt voor zich.
Omgekeerd kunnen vanuit medewerkers-detailscherm één of meer tegels toegekend worden aan een medewerker.
- **Controleer de zichtbaarheid sql-statement.** Een wizardscherm controleert de syntax van het opegegeven statement.

Triggers in scherm

* **controleer en toon responsetijd van onderliggende dynamische tegelopschrift** . Achter de kolom *Tegelopschrift dynamisch met API* `getTileContent()` of `getFlexChart()` is een knop waarmee de responsetijd van de aangeroepen query kan worden gemeten.

Openwave vraagt alvorens de metingen uit te voeren voor welke inlogger de eventuele substitutie van de variabele `:keyaccount` in de queries moet plaatsvinden en ook , indien de tegelk op een zaakportaal of inrichtingsportaal staat, voor welke portalid (is de primary key – dnkey - van de zaak die de variabele `{id}` vervangt) de queries uitgevoerd moeten worden.

Indien deze laatste portalid niet wordt opgegeven gebruikt Openwave de niet bestaande dnkey met waarde 0

Mogelijkheden van kolom actie

Zie: [Actions](#).

From:
<https://acc-doc.open-wave.nl/> - **Documentatie**

Permanent link:
https://acc-doc.open-wave.nl/doku.php/openwave/1.30/applicatiebeheer/instellen_inrichten/portaldefinitie/portal_tegel

Last update: **2025/05/26 12:28**