

OData gebruikershandleiding

Tim ondersteunt het OData 4 (4.0) protocol voor import en export. Dit is een standaard voor gegevens import en export gebaseerd op het [REST](#) principe. Hieronder links naar detail documentatie.

Dit document is als volgt onderverdeeld:

- **Configuratie Tim server:** Hoofdstuk 1
- **Technische beschrijving:** Hoofdstuk 2, 3, 4, 5 en 6
- **Client integraties:** Hoofdstuk 7

1. Tim server configuratie
 - 1.1 System properties
2. Datamodel
 - 2.1 Sleutels
 - 2.1.2 Primaire sleutel
 - 2.1.3 Alternatieve sleutels
 - 2.2 Relaties
 - 2.3 Datum en tijd
 - 2.3.1 Representatie periode
 - 2.3.2 Representatie lege datum en tijd waarden
 - 2.3.3 Tijdzones
3. Ondersteunde OData features
 - 3.1 Output format
 - 3.2 Service document
 - 3.3 Metadata
 - 3.4 System query parameters
 - 3.4.1 Veld referenties
 - 3.4.1.1 \$orderby
 - 3.4.1.2 \$filter
 - 3.5 Path suffixes
 - 3.6 Aliassen
4. Export van gegevens
 - 4.1. Algemeen
 - 4.2. Enkele records
 - 4.2.1 Ondersteunde OData features
 - 4.3 Collecties
 - 4.3.1 Paginering
 - 4.3.2 Ondersteunde OData features
 - 4.3.3 Aantal records opvragen
 - 4.3.4 Foutafhandeling
 - 4.4 Response codes
5. Import van gegevens
 - 5.1 Response codes
6. Foutafhandeling
7. Client integraties
 - 7.1. Power Query OData feed connector
 - 7.1.1 Gebruik
 - 7.1.2 Links
 - 7.2 OData client integratie software ontwikkel libraries

1. Tim server configuratie [↗](#)

Om OData import en export beschikbaar te maken moet de SOAP module geactiveerd zijn. De OData koppeling deelt zijn credentials en poort met de SOAP koppeling. Na configuratie van de SOAP koppeling is de OData service root beschikbaar onder URL

`/impexp/odata/v1/`. Bijvoorbeeld `https://timenterprise.nl/impexp/odata/v1/`.

1.1 System properties [↗](#)

Property	Omschrijving
<code>nl.aenova.odata.collecti on.max-page-size</code>	Bepaalt de maximale grootte van een collectie response. Standaard waarde is <code>1000</code> . Gebruik waarde <code>2147483647</code> om paginering helemaal uit te schakelen.

Hieronder een uitleg van de Tim specifieke eigenschappen van het OData datamodel. Het datamodel wordt formeel aangeboden via het `/impexp/odata/v1/$metadata` endpoint.

2. Datamodel [↗](#)

2.1 Sleutels [↗](#)

2.1.2 Primaire sleutel [↗](#)

De primaire sleutel is het record id. De veldnaam van dit veld is `ref`. Voorbeeld:

- product met record id 1: `/impexp/odata/v1/Producten(1)` of expliciet `/impexp/odata/v1/Producten(ref=1)`

2.1.3 Alternatieve sleutels [↗](#)

Tim ondersteunt [Alternate Keys](#). De sleutel combinaties zijn in het metadata document terug te vinden via de annotatie `Org.OData.Core.V1.AlternateKey` onder de `Entity` definities. Bij gebruik van alternate keys moeten de sleuteldelen altijd expliciet benoemd worden. Voorbeeld:

- `/impexp/odata/v1/Aanvragen(persoon_HRMID='4035',roostercategorie_code='AGV',periode_start=2024-12-09T00:00:00Z,afdeling_HRMID='853')`
 - `persoon` op basis van `HRMID`
 - `afdeling` op basis van `HRMID`
 - `roostercategorie` op basis van `code`
 - met periode/start van `2024-12-09 00:00 UTC`

2.2 Relaties [↗](#)

Op dit moment worden alleen 1-1 relaties ondersteund. Het is dus bijvoorbeeld niet mogelijk om alle tijden behorende bij een specifiek product op te halen via `/impexp/odata/v1/Producten(1)/Tijden`. Om dit te bereiken kan `$filter` gebruikt worden op de `Tijden` collectie: `/impexp/odata/v1/Tijden?$filter=product/ref eq 1`.

2.3 Datum en tijd [↗](#)

2.3.1 Representatie periode [↗](#)

Tim ondersteunt complexe periode types met start en eind. In het datamodel worden deze gerepresenteerd als complex type `ASTimestampPeriod` met daarin de members `start` en `end`.

Voorbeeld XML representatie:

```

1 <d:periode m:type="#TimEnterprise.ASTimestampPeriod">
2   <d:start m:type="DateTimeOffset">2025-02-10T23:00:00Z</d:start>
3   <d:end m:type="DateTimeOffset">2025-02-11T22:59:59.999Z</d:end>
4 </d:periode>

```

Voorbeeld in JSON representatie:

```

1 {
2   "start": "2025-02-10T23:00:00Z",
3   "end": "2025-02-11T22:59:59.999Z"
4 }

```

2.3.2 Representatie lege datum en tijd waarden [↗](#)

Tim kent het concept lege start of eind datum/tijd. Een concept dat OData niet kent. Als deze waarden geëxporteerd worden worden deze als volgt gepresenteerd:

Soort	Waarde
-------	--------

Soort	Waarde
Periode start/enkel	
Datum	0001-01-01
Tijd	00:00
Tijdstip	0001-01-01T00:00:00Z
Periode eind	
Datum	9999-12-31
Tijd	23:59:59.999
Tijdstip	9999-12-31T23:59:59.999Z

2.3.3 Tijdzones [↗](#)

Datum/Tijd waarden uit Tim worden geëxporteerd als OData type `Edm.DateTimeOffset`. Dit type vereist een tijdzone.

Het datamodel van Tim ondersteunt geen tijdzones. Dit betekent dat via OData ontvangen datum/tijd waarden eerst geconverteerd worden naar de lokale tijd van de Tim server op basis van de timezone van het systeem (in de SaaS omgeving is dit `Europe/Amsterdam`).

Exporteren van datetime waarden via OData gebeurt altijd in timezone `UTC`. Dus niet met de timezone van de server.

Voorbeelden:

- Wintertijd `2025-03-28 00:00` in Tim met een server timezone van `Europe/Amsterdam` wordt in ISO-8601 gerepresenteerd als `2025-03-28T00:00:00+01:00`. In UTC zal dit er als `2025-03-27T23:00:00Z` uitzien.
- Zometijd `2025-03-31 23:59:59.999` in Tim met een server timezone van `Europe/Amsterdam` wordt in ISO-8601 gerepresenteerd als `2025-03-31T23:59:59.999+02:00`. In UTC zal dit er als `2025-03-31T21:59:59.999Z` uitzien.

3. Ondersteunde OData features [↗](#)

Tim ondersteunt het OData 4 (4.0) protocol voor import en export. De specificatie is beschreven in [Documentation · OData - the Best Way to REST](#).

Voor de export van gegevens is Tim volledig compliant met:

- [OData 4.0 Minimal Conformance Level](#); TimEnterprise 2024.1 en hoger
- [OData 4.0 Intermediate Conformance Level](#); TimEnterprise 2025.1.2 en hoger

Tim voldoet gedeeltelijk aan het [Advanced](#) conformance level.

3.1 Output format [↗](#)

De OData service kan gegevens in `JSON` of `XML` aanleveren. Vanuit de client zijn er 2 manieren om het output formaat te specificeren:

HTTP Accept header	
JSON	application/json
XML	application/xml
System query parameter <code>\$format</code>	
JSON	<code>\$format=json</code>
XML	<code>\$format=xml</code>

3.2 Service document [↗](#)

Het service document beschrijft alle beschikbare collecties. Het service document is op te vragen via de service root URL is `/impexp/odata/v1/`. Het service document volgt het Atom Publishing Protocol (AtomPub) zoals beschreven in [RFC Information on RFC 5023 » RFC Editor](#).

3.3 Metadata [↗](#)

Het metadata document beschrijft het datamodel en is op te vragen via `/impexp/odata/v1/$metadata`.

3.4 System query parameters [↗](#)

Parameter	Scope	Details	Tim versie
<code>\$top</code>	• Export collectie	Bepert het aantal resultaten tot de opgegeven waarde • Voorbeeld: <code>/impexp/odata/v1/Tijden?\$top=20</code>	2025.1.1
<code>\$skip</code>	• Export collectie	Slaat het opgegeven aantal records over • Voorbeeld: <code>/impexp/odata/v1/Tijden?\$skip=20</code>	2025.1.1
<code>\$count</code>	• Export collectie	Telt het totaal aantal records rekening houdende met <code>\$filter</code> . • Voorbeeld <code>/impexp/odata/v1/Tijden?\$count=true</code> <ul style="list-style-type: none">◦ De count waarde wordt als property toegevoegd aan het resultaat	2025.1

		<ul style="list-style-type: none"> • <code>\$count</code> ondersteunt ook de path segment variant <ul style="list-style-type: none"> ◦ NB: Dit is beperkt tot de root URL van een collectie. En kan dus niet gebruikt worden in <code>\$filter</code>. ◦ Voorbeeld <code>/impexp/odata/v1/Tijden/\$count</code> <ul style="list-style-type: none"> ▪ Het resultaat is de numerieke count waarde 	
<code>\$select</code>	<ul style="list-style-type: none"> • Export collectie • Export record • Import record 	<p>Selecteert specifieke velden. Om velden van referenties te selecteren moet <code>\$expand</code> gebruikt worden.</p> <ul style="list-style-type: none"> • Voorbeeld <code>/impexp/odata/v1/Tijden?\$select=datum</code> 	2024.1
<code>\$expand</code>	<ul style="list-style-type: none"> • Export collectie • Export record • Import record 	<p>Selecteert velden van referenties</p> <ul style="list-style-type: none"> • Expand de target referentie 1 niveau diep. <ul style="list-style-type: none"> ◦ Voorbeeld <code>/impexp/odata/v1/Tijden?\$expand=persoon</code> • Bij gebruik van wildcard <code>*</code> waarde word de expand uitgevoerd op alle referenties <ul style="list-style-type: none"> ◦ Voorbeeld <code>/impexp/odata/v1/Tijden?\$expand=*</code> • Kan gecombineerd worden met geneste <code>\$select</code> en <code>\$expand</code> statements. <ul style="list-style-type: none"> ◦ Voorbeeld <code>/impexp/odata/v1/Tijden?\$expand=persoon(\$select=naam,emailAdres;\$expand=tariefgroep(\$select=naam)),product(\$select=naam,code)</code> ◦ Voorbeeld <code>/impexp/odata/v1/Tijden?\$expand=persoon(\$expand=*)</code> 	2024.1
<code>\$filter</code>	<ul style="list-style-type: none"> • Export collectie 	<p>Filtret het resultaat</p> <ul style="list-style-type: none"> • Ondersteunde operators: <code>and</code>, <code>or</code>, <code>eq</code>, <code>ne</code>, <code>lt</code>, <code>le</code>, <code>gt</code>, <code>ge</code>, <code>not</code> • Ondersteunt veld referenties en constante waarden <ul style="list-style-type: none"> ◦ Voorbeeld veld referentie met duur waarde <code>/impexp/odata/v1/Tijden?\$filter=duration eq 'PT1H35M0S'</code> ◦ Voorbeeld veld referentie met enum waarde <code>/impexp/odata/v1/Aanvragen?\$filter=status eq TimEnterprise.RegistratieAcceptatie'NEUTRAAL'</code> ◦ Voorbeeld veld referentie met periode waarde <code>/impexp/odata/v1/Tijden?\$filter=datum eq 2012-01-02</code> ◦ Voorbeeld geneste veld referentie <code>/impexp/odata/v1/Tijden?\$filter=persoon/code eq 'JVI'</code> ◦ Voorbeeld combinatie <code>/impexp/odata/v1/Tijden?\$filter=(persoon/code eq 'JVI') and (datum eq 2012-01-02)</code> 	2024.1
<code>\$orderby</code>	<ul style="list-style-type: none"> • Export collectie 	<p>Sorteert het resultaat</p> <ul style="list-style-type: none"> • Voorbeeld sorteert oplopend op datum en aflopend op persoon code: <code>/impexp/odata/v1/Tijden?\$orderby=datum asc,persoon/code desc</code> 	2024.1
<code>\$format</code>	<ul style="list-style-type: none"> • Export collectie 	<p>Bepaalt het formaat waarin het antwoord wordt gegeven, mogelijke waarden:</p> <ul style="list-style-type: none"> • <code>json</code> • <code>xml</code> 	2024.1

	<ul style="list-style-type: none"> • Export record • Service document • Metadata • Import record 	
--	--	--

3.4.1 Veld referenties [↗](#)

3.4.1.1 \$orderby [↗](#)

Veld referenties hebben onbeperkte diepte bij gebruik in `$orderby`.

3.4.1.2 \$filter [↗](#)

Bij gebruik in `$filter` kunnen velden met maximaal 1 indirectie gespecificeerd worden. De enige uitzonderingen zijn de primaire sleutel en onderdelen van de complexe types (periode).

Voorbeeld van veld in `$filter`:

- Geldig: `/impexp/odata/v1/Tijden?$filter=datum eq 2012-01-02`
- Geldig: `/impexp/odata/v1/Tijden?$filter=product/code eq 'code'`
- Geldig: `/impexp/odata/v1/Tijden?$filter=product/tariefgroep/ref eq 39`
- Geldig: `/impexp/odata/v1/Tijden?$filter=product/periode/start eq 2009-01-01`
- Ongeldig: `/impexp/odata/v1/Tijden?$filter=product/tariefgroep/code eq 'code'`

3.5 Path suffixes [↗](#)

Path segment	Scope	Details	Tim versie
<code>/\$count</code>	• Export collectie	Geeft het aantal records van een (gefilterde) collectie als numerieke waarde <ul style="list-style-type: none"> • Voorbeeld: <code>/impexp/odata/v1/Producten/\$count?\$filter=naam eq 'prod'</code>. 	2025.1
<code>/\$value</code>	• Export record	Geeft de waarde van een record veld/property terug <ul style="list-style-type: none"> • Voorbeeld <code>/impexp/odata/v1/Tijden(1)/duur/\$value</code> zal <code>PT1H0.000S</code> teruggeven voor een duur van 1 uur • Voorbeeld <code>/impexp/odata/v1/Tijden(1)/gewijzigdDoor/persoon/naam/\$value</code> zal de naam van de persoon teruggeven die de tijdregistratie als laatste heeft aangepast 	2025.1.2

3.6 Aliassen [↗](#)

Aliassen worden alleen ondersteund bij gebruik in `$filter`. Voorbeeld: `/impexp/odata/v1/Tijden?$filter=persoon/code eq @code&@code='JVI'`

4. Export van gegevens [↗](#)

4.1. Algemeen [↗](#)

De collectie end points zijn te vinden in het service document dat op te vragen is via service root `/impexp/odata/v1/`.

4.2. Enkele records [↗](#)

Enkele records zijn op te vragen door de sleutel tussen haakjes achter het collectie end point te plaatsen. Tim ondersteunt primaire sleutels en alternatieve sleutels.

Primaire sleutel is het record id en gebruikt veldnaam `ref`

- Voorbeeld `/impexp/odata/v1/Producten(1)`.
- Voorbeeld `/impexp/odata/v1/Producten(ref=1)`

Alternatieve sleutels zijn sleutels die niet de primaire sleutel zijn en kunnen samengesteld zijn. Bij gebruik van alternatieve sleutels zijn de namen van sleuteldelen verplicht

- Voorbeeld `/impexp/odata/v1/Producten(code='123')`.
- Voorbeeld
`/impexp/odata/v1/Tijden(persoon_code='pers',product_code='prod',afdeling_afdelingscode='afd',datum=2025-02-17)`.

4.2.1 Ondersteunde OData features [↗](#)

Voor ondersteunde OData features zie de features met scope “*Export record*” in de betreffende hoofdstukken:

- 3.4 System query parameters

4.3 Collecties [↗](#)

4.3.1 Paginering [↗](#)

Voor collecties wordt standaard een maximum page size van `1000` records toegepast. Deze maximum page size is configureerbaar, zie hoofdstuk: 1.1 System properties .

Clients kunnen binnen deze maximum page size een eigen page size verzoeken. Dit gaat via de HTTP `Prefer` header met property `odata.maxpagesize`.

Een voorbeeld request:

```
1 GET https://www.timenterprise.nl/impexp/odata/v1/Personen
2 Accept: application/json
3 Cache-Control: no-cache
4 Prefer: odata.maxpagesize=10
```

De daadwerkelijk toegepaste page size is afhankelijk van de maximum page size zoals ingesteld in de Tim Server. Deze toegepaste waarde wordt in de HTTP response teruggeven via de `Preference-Applied` header.

Een voorbeeld response header:

```
1 Preference-Applied: odata.maxpagesize=10
```

Let op: Bij het navigeren naar vervolg paginas via de next link in de response moet de `Prefer` header elke keer opnieuw toegepast worden.

4.3.2 Ondersteunde OData features [↗](#)

Voor ondersteunde OData features zie de features met scope “*Export collectie*” in de betreffende hoofdstukken:

- 3.4 System query parameters

- 3.5 Path suffixes

4.3.3 Aantal records opvragen [↗](#)

Het aantal records van een (gefilterde) collectie is direct op te vragen kan door `/$count` achter de URL van de collectie te plaatsen. Bv `/impexp/odata/v1/Producten/$count?$filter=naam eq 'prod'`.

4.3.4 Foutafhandeling [↗](#)

Collectie responses gebruiken server side streaming. Gevolg hiervan is dat de HTTP status in de response in het geval van een fout tijdens het streamen niet klopt in geval van een fout tijdens het streamen. De HTTP status zal dan `Ok (200)` zijn. Bij fouten tijdens het streamen wordt de foutmelding als tekst in de structuur van de payload geplaatst waardoor deze niet meer correct ontleedt kan worden. Bij een fout tijdens het parsen moet ervanuit gegaan worden dat de hele request foutief is en opnieuw gedaan moet worden.

4.4 Response codes [↗](#)

HTTP Status code	Betekenis
Ok (200)	Ok
Bad Request (400)	Fout in de request.

5. Import van gegevens [↗](#)

Tim ondersteunt alleen import van losse records via HTTP `PATCH` verb. Dit is een Upsert operatie. Dit betekent dat als een record niet bestaat het gemaakt wordt en anders wordt het bestaande records geüpdatet. Er moet altijd een sleutel aanwezig zijn om het record te identificeren. Dit kan een primary key of alternate key zijn. Bij het aanmaken van een nieuw record hebben de sleutelwaarden altijd voorrang op de payload.

Voorbeeld request (JSON):

```

1 PATCH
2 https://www.timenterprise.nl/impexp/odata/v1/Afdelingsleden(afdeling_HRMID='016',persoon_HRMID='042',periode_start=0001-01-01) HTTP/1.1
3 Accept: application/json
4 Content-Type: application/json
5 OData-MaxVersion: 4.0
6 OData-Version: 4.0
7 Connection: Keep-Alive
8 User-Agent: test
9 {
10   "periode": {
11     "end": "2025-12-31"
12   },
13   "afdeling@odata.bind": "Afdelingen(HRMID='015')"
14 }

```

Als bovenstaande voorbeeld resulteert in een nieuw record zal de afdeling met `HRMID '016'` toegepast zijn volgens de sleutel. Bij een update zal de afdeling en daarmee de sleutel geüpdatet worden naar de afdeling met `HRMID '015'` volgens de payload.

5.1 Response codes [↗](#)

HTTP Status code	Betekenis
Ok (200)	Bestaand record geüpdatet

Created (201)	Nieuw record toegevoegd
Internal Server Error (500)	Record afgewezen door business logica
Bad Request (400)	Fout in de request (payload)

6. Foutafhandeling [↗](#)

Als er iets misgaat tijdens een request wordt een detail melding in het onderstaande formaat teruggegeven. Het formaat is afhankelijk van de HTTP `Accept` header. De `$format` parameter heeft geen effect op de foutmelding.

NB: Dit geldt niet voor collectie responses. Zie het betreffende hoofdstuk 4.3.4 Foutafhandeling voor meer informatie.

JSON

```
1 {
2   "error": {
3     "code": null,
4     "message": "The URI is malformed."
5   }
6 }
```

XML

```
1 <error xmlns="http://docs.oasis-open.org/odata/ns/metadata">
2   <code>null</code>
3   <message>The URI is malformed.</message>
4 </error>
```

7. Client integraties [↗](#)

7.1. Power Query OData feed connector [↗](#)

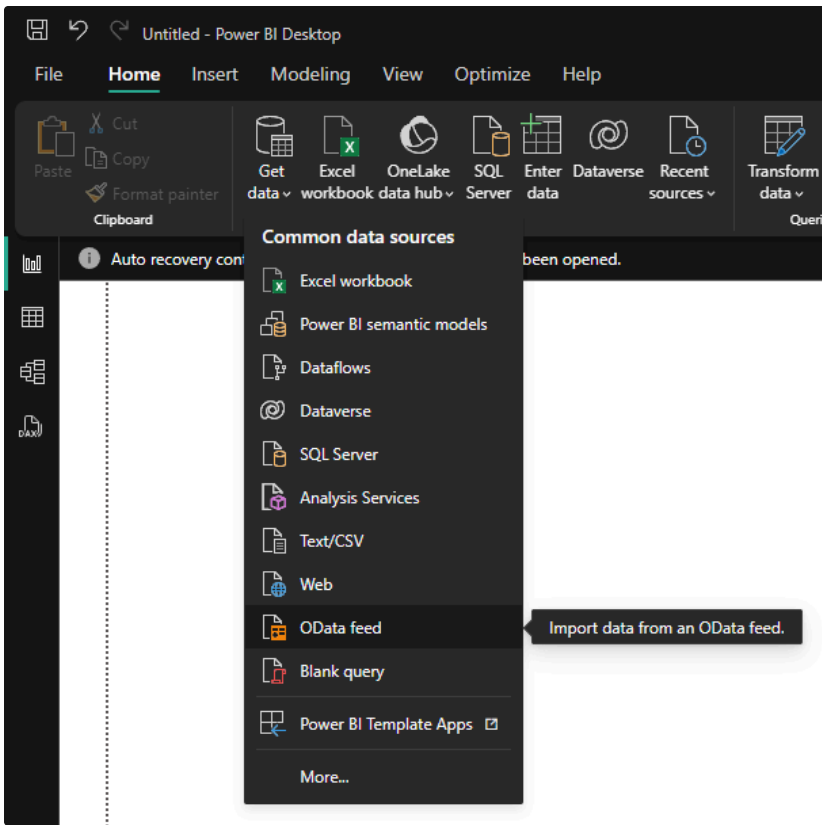
Deze [connector](#) is geïntegreerd in de volgende Microsoft software:

- Excel
- Power BI (Semantic models)
- Power BI (Dataflows)
- Fabric (Dataflow Gen2)
- Power Apps (Dataflows)
- Dynamics 365 Customer Insights
- Analysis Services

Met de OData feed connector kunnen direct gegevens geïmporteerd en getransformeerd worden vanuit Tim.

7.1.1 Gebruik [↗](#)

Kies voor OData feed.



Vul de OData service root url in van de Tim server. Dit is `/impexp/odata/v1/`:



Selecteer en transformeer de gegevens uit de genodigde collectie:

Navigator

Display Options ▾

- Producten
- Productrechten
- Productrubrieken
- ProjectAfhankelijkheden
- ProjectStuurgroepen
- RapportageStatistieken
- Rekeningnummers
- ResourceAanvraagUitzonderingen
- ResourceAanvragen
- RoosterBlokken
- RoosterCategorieën
- RoosterPatronen
- RoosterRegistraties
- ScheduledJobs
- Tariefgroepen
- TariefHistorie
- Tijden**
- VergoedingLocaties
- VerplichteRoosterBlokken
- Werknemertypes

Tijden

ref	datum	duur	soort	bedrag	overheadBedrag	compensatie
1	2/11/2025	0.02:00:00	NORMAAL	0	0	0
2	2/11/2025	0.06:00:00	NORMAAL	0	0	0
4	2/17/2025	0.01:00:00	NORMAAL	0	0	0
9	2/12/2025	0.02:00:00	NORMAAL	0	0	0

Select Related Tables

Load Transform Data Cancel

7.1.2 Links [↗](#)

- [Power Query OData-feedconnector - Power Query](#)
- [Naslaginformatie over de formuletaal Power Query M - PowerQuery M](#)
- [DateTimeZone-functies - PowerQuery M](#)

7.2 OData client integratie software ontwikkel libraries [↗](#)

Hieronder een collectie van software ontwikkel libraries waarmee custom integraties gemaakt kunnen worden met de OData koppeling van Tim:

- .NET: [GitHub - OData/odata.net: ODataLib: Open Data Protocol - .NET Libraries and Frameworks](#)
- Java: [GitHub - davidmoten/odata-client: Java client generator for a service described by OData CSDL 4.0 metadata. Includes Micro soft Graph clients \(v1.0 and Beta\), Graph Explorer client, Analytics for DevOps, Dynamics CRM clients](#)
- Python: [GitHub - SAP/python-pyodata: Enterprise-ready Python OData client](#)