

Vereinbarung über Datenbereitstellung

abgeschlossen zwischen dem

Österreichischer Automobil-, Motorrad- und Touring Club

Baumgasse 129

A-1030 Wien

UID ATU36821301

ZVR 730335108

als Datenlieferant im Folgenden kurz „**ÖAMTC**“ genannt

und

[Vertragspartner]

[Adresse]

[UID]

[FB-Nr]

im Folgenden kurz „**Datennutzer**“ genannt

I. Präambel

Der ÖAMTC erfasst und verifiziert Informationen über Park- & Ride-Anlagen, Parkhäuser und Kurzparkzonen (Parkraumbewirtschaftete Zonen). Neben Geo-Positionen für die Anzeige in digitalen Kartenanwendungen umfassen die Daten zahlreiche weitere Attribute wie Preise, Bezahlungsmöglichkeiten sowie Öffnungs- bzw. Gültigkeitszeiten (vgl. Anlage /1 Datenbeschreibung). Sämtliche Inhalte werden in den ebenfalls in Anlage /1 beschriebenen Intervallen regelmäßig aktualisiert.

[Vom Vertragspartner zu ergänzen: Der Datennutzer [Beschreibung der Aktivitäten / Geschäftsfelder des Datennutzers inkl. wozu die Daten des ÖAMTC benötigt werden]

II. Vertragsgrundlagen

(1) Die nachstehenden Unterlagen bilden einen integrierenden Bestandteil der gegenständlichen Kooperationsvereinbarung:

- Anlage /1: Beschreibung der ÖAMTC-Daten, der Aktualisierungsintervalle und der Schnittstellen für den Datenabgriff
- Anlage /2: IP-Adressen des Datennutzers mit denen der Datennutzer den Datenabgriff an den in Anlage /1 definierten Schnittstellen durchführt
- Anlage /3: Darstellung der Quellenangabe ÖAMTC in den Ausgabemedien des Datennutzers (screenshot)

- (2) Bei Widersprüchen gilt die gegenständliche Vereinbarung vorrangig vor den Anhängen.

III. Vertragsgegenstand

- (1) Folgende ÖAMTC-Daten werden dem Datennutzer zu den unter ./IV beschriebenen Bedingungen ausschließlich für die ebenfalls dort definierten Verwendungszwecke zur Verfügung gestellt, wobei sich der Vertragspartner verpflichtet, im Gegenzug die unter ./V genannten Leistungen für den ÖAMTC zu erbringen:

- Offstreet-Parkings: Parkgaragen und Park & Ride-Anlagen österreichweit
- Kurzparkzonen Österreich (60 Städte)
[Auswahl wird je nach Bestellung angepasst]

- (2) Die Daten werden in dem in Anlage /.1 beschriebenen Format über die in Anlage /.1 beschriebenen Schnittstellen bereitgestellt.

IV. Vertragsbedingungen

Der Datennutzer verpflichtet sich zur Erfüllung folgender Bedingungen:

- (1) Der Datennutzer verfügt über keine Exklusivrechte für die Nutzung der ÖAMTC-Daten.
- (2) Der Datennutzer ist verpflichtet, die IP-Adresse(n) mit denen seine Systeme auf die ÖAMTC-Daten zugreifen, dem ÖAMTC bekannt zu geben. Diese sind im Anlage /.2 vom Datennutzer einzutragen. Änderungen bei den IP-Adressen, die sich im Laufe der Kooperation ergeben und / oder zusätzliche IP-Adressen, die vom Datennutzer verwendet werden, sind dem ÖAMTC unverzüglich, spätestens aber nach 10 Werktagen ab Aktivsetzung per e-Mail (staumelder@oeamtc.at) dem ÖAMTC bekanntzugeben.
- (3) Eine Weitergabe der ÖAMTC-Daten durch den Datennutzer an Dritte ist grundsätzlich untersagt.
- (4) Der Datennutzer verwendet die ÖAMTC-Daten ausschließlich für folgende Zwecke:
- [Beschreibung Verwendung der ÖAMTC-Daten durch den Datennutzer, vom Datennutzer zu ergänzen]
 - Die Archivierung einzelner ÖAMTC-Datensätze oder von Gesamt-Beständen ist nicht gestattet. Einmal abgegriffene Daten dürfen nur bis zum nächsten Daten-Update ausschließlich für den oben genannten Zweck in den Systemen des Datennutzers gespeichert

werden. Sobald aktuellere Daten vom Datennutzer abgegriffen werden, sind von diesem sämtliche älteren ÖAMTC-Daten unverzüglich zu löschen.

- (5) Die Anzeige der ÖAMTC-Daten in den Applikationen des Datennutzers erfolgt unter Angabe der Quelle ÖAMTC. Art der Darstellung der Quellenangabe ist in Anlage /3 dargestellt.

V. Entgelt und Zahlungsbedingungen

- (1) Die jährlichen Lizenzkosten sind wertgesichert und unterliegen folgender Anpassungen: Für das erste volle Nutzungsjahr erfolgt keine Wertanpassung. Ab Vollendung des ersten Nutzungsjahres werden die Lizenzkosten entsprechend der Preissteigerung des jeweiligen Verbraucherpreisindex (VPI) oder eines an seine Stelle tretenden Index angepasst. Verglichen werden jeweils die für Jänner des Vorjahres veröffentlichten Indexzahlen mit den entsprechenden für Jänner des Vorvorjahres veröffentlichten Indexzahlen.

Die Wertanpassung erfolgt einmal jährlich, mit Wirksamkeit zum 1. Jänner eines jeden Kalenderjahres. Die Möglichkeit zur Wertanpassung besteht demnach erstmals zum **[01.01.2027.]**

Die Wertanpassung erfolgt nur, wenn sich die Indexzahlen in gewichteter Summe um mehr als 2 % verändert haben. Bei Überschreiten der 2 %-Schwelle kommt die gesamte Veränderung zur Anwendung, nicht also nur der über 2 % liegende Teil. (Kommt es mangels Überschreitung der 2 %-Schwelle zu keiner Anpassung, bleiben die Indexzahlen des vorvorigen Jahres für die nächste Wertanpassung maßgeblich.)

- (2) Die jährlichen Lizenzkosten für das erste Nutzungsjahr betragen EUR **[Betrag je nach Datenart(en) einzufügen, Kosten orientieren sich am Verwendungszweck des Datennutzers]** zzgl. Mehrwertsteuer.
- (3) Die Rechnungslegung erfolgt jährlich im Vorhinein mit Jahresanfang (spätestens bis 15. Februar) auf Basis einer vom ÖAMTC dem Umsatzsteuergesetz entsprechenden Rechnung. Die darin ausgewiesenen Beträge sind innerhalb von 30 Tagen nach Rechnungserhalt auf das in der Rechnung angegebene ÖAMTC-Konto zu überweisen.
- (4) Für die Einrichtung der Schnittstelle zum ÖAMTC Geo-Daten-Server inkl. technischem Support wird zusätzlich einmalig: EUR 1.020,00 (zzgl. 20 % MWSt.) verrechnet. Der Betrag ist nach Bestätigung des funktionierenden Datenstroms durch den Datennutzer (per mail an den ÖAMTC) fällig und wird in der ersten Jahresrechnung ausgewiesen.
- (5) Sämtliche bei der Vertragsausführung entstehenden Nebenkosten und Auslagen (bspw. Reisespesen, Aufenthaltskosten, Aufwandsentschädigungen etc.) sind von dem jeweiligen Vertragspartner vollumfänglich selbst zu tragen.

- (6) Leistungen der beiden Vertragspartner, die über jene, die in /.IV beschrieben sind, hinausgehen, müssen gesondert schriftlich vereinbart werden. Ohne eine solche Vereinbarung erbrachte Leistungen lösen keinerlei Verpflichtung des jeweils anderen Vertragspartners aus.

VI. Gewährleistung und Leistungsstörung

- (1) Der ÖAMTC haftet dem Datennutzer gegenüber für den Ausfall der technischen Systeme zur Datenbereitstellung nur bei grober Fahrlässigkeit. Der ÖAMTC ist in jedem Fall bemüht, den Ausfall so rasch wie möglich zu beheben und den Datenfluss wiederherzustellen.
- (2) Der ÖAMTC ist im Rahmen dieser Vereinbarung als Aggregator und Distributor der unter /.I genannten Daten tätig. Der ÖAMTC ist im Zuge seiner Tätigkeit als Aggregator und Distributor selbst auf Richtigkeit und Aktualität der Daten seiner Informationsquellen (z.B.: Parkhaus-Betreiber, Kommunen etc.) angewiesen. Der ÖAMTC haftet aus diesem Grunde nicht für die Richtigkeit und Aktualität der Daten.
- (3) Der ÖAMTC garantiert keinen Service-Level, bestätigt aber, dass die Daten-Verfügbarkeit in den vergangenen Jahren bei nahezu 100 Prozent lag.
- (4) Der Ersatz von Folgeschäden, Vermögensschäden, entgangenem Gewinn und von Schäden aus Ansprüchen Dritter ist jedenfalls ausgeschlossen.

VII. Marketing

- (1) Sämtliche Marketing- sowie PR-Aktionen vom Datennutzer, bei denen der ÖAMTC genannt werden soll – wie bspw. die Präsentation des ÖAMTC als Partner für Verkehrsinformationen auf der Website des Datennutzers– bedürfen jeweils der vorherigen Abstimmung sowie der schriftlichen Zustimmung des ÖAMTC.

VIII. Verschwiegenheit

- (1) Beide Vertragspartner verpflichten sich während und auch nach Beendigung dieses Vertragsverhältnisses zur Geheimhaltung von Geschäfts- und Betriebsgeheimnissen bzw. von vertraulichen Informationen der anderen Partei.
- (2) Der Datennutzer verpflichtet sich auch zur Verschwiegenheit über alle ihm im Rahmen dieses Vertragsverhältnisses bekannt gewordenen, den ÖAMTC betreffenden und nicht allgemein bekannten Tatsachen.
- (3) Der Datennutzer wird auch seine Mitarbeiter und Partner entsprechend zur Geheimhaltung verpflichten.

- (4) Darüber hinaus verpflichten sich beide Vertragspartner, insbesondere die Bestimmungen der DSGVO (EU/2016/679) sowie des DSG2018 (BGBl. I Nr. 120/2017) einzuhalten und diese auf ihre im Rahmen dieser Vereinbarung tätig werdenden Mitarbeiter und Partner zu übertragen und alle notwendigen Datensicherheitsmaßnahmen zu ergreifen.
- (5) Die Geheimhaltungsverpflichtung gilt auch für die Zeit nach Beendigung dieses Vertragsverhältnisses unbefristet weiter.

IX. Laufzeit und Kündigung

- (1) Die Kooperationsvereinbarung tritt mit Unterzeichnung durch beide Vertragsparteien in Kraft und gilt vorerst für die Dauer von drei Jahren, somit bis zum [Datum einsetzen]. Sollte keiner der beiden Vertragspartner die Vereinbarung vorzeitig auflösen (kündigen), wird die Geltungsdauer automatisch unbefristet verlängert.
- (2) Der Vertrag kann von jeder der Vertragsparteien unter Einhaltung einer Kündigungsfrist von drei Monaten jeweils zum Quartalsende [sohin zum 31. März, 30. Juni, 30. September oder 31. Dezember] aufgekündigt werden. Die Aufkündigung muss schriftlich eingeschrieben erfolgen.
- (3) Darüber hinaus kann der Vertrag von jeder der Vertragsparteien nur aus wichtigem Grund mit sofortiger Wirkung aufgelöst werden. Die Auflösung aus wichtigem Grund ist schriftlich eingeschrieben zu erklären. Als wichtiger Grund gilt für den ÖAMTC insbesondere,
 - wenn ein Antrag auf Eröffnung eines Insolvenzverfahrens über das Vermögen des Vertragspartners mangels Vermögens abgewiesen wird oder die Voraussetzungen für eine solche Abweisung vorliegen;
 - der Datennutzer eine wesentliche Verpflichtung aus dieser Vereinbarung verletzt und trotz Mahnung und der Setzung einer angemessenen, zumindest einwöchigen Nachfrist zur Herstellung des vertragskonformen Zustandes, die Verletzung nicht einstellt oder behebt;
 - bei einer Verletzung der Datenschutzbestimmungen, Vertraulichkeitsverpflichtung oder der Regelung zum Schutz von IP-Rechten des ÖAMTC kann die Auflösung des Vertrages auch ohne Nachfristsetzung erklärt werden.

X. Salvatorische Klausel

- (1) Sollten einzelne Bestimmungen dieses Vertrages nichtig, undurchführbar oder ungültig sein oder werden, so hat dies nicht die Nichtigkeit zur Folge. Die Vertragsteile verpflichten sich für diesen Fall, anstelle der nichtigen, undurchführbaren oder ungültigen Bestimmungen eine Regelung zu

vereinbaren, die dem mit der nichtigen, undurchführbaren oder ungültigen Regelung verfolgten Zweck am wirtschaftlich nächsten kommt.

XI. Anzuwendendes Recht

- (1) Auf dieses Vertragsverhältnis ist das materielle österreichische Recht unter Ausschluss der Verweisungsnormen des österreichischen internationalen Privatrechts und des UN-Kaufrechtsübereinkommens anzuwenden.
- (2) Für den Fall von Streitigkeiten welcher Art immer aus diesem Vertrag oder in Zusammenhang mit diesem Vertrag, sowie über das Zustandekommen dieses Vertrages und seine Rechtswirksamkeit wird die ausschließliche Zuständigkeit des sachlich zuständigen Gerichts für Wien Innere Stadt vereinbart.

XII. Schlussbestimmungen

- (1) Änderungen oder Ergänzungen dieser Vereinbarung bedürfen, um rechtswirksam zu werden, der Schriftform sofern nicht nach den gesetzlichen Bestimmungen eine andere Form zwingend vorgeschrieben ist. Die Schriftform ist auch für ein Abgehen von dieser Schriftformklausel erforderlich.
- (2) Der Vertrag wird in 2-facher Ausfertigung erstellt, wovon jede Vertragspartei eine Ausfertigung erhält.

Wien, am _____

_____, am _____

ÖAMTC

Datennutzer

Anlage 1

Beschreibung des ÖAMTC-Datenformats und der Schnittstellen für den Datenabgriff

Datenbereitstellung

[wird vom ÖAMTC ergänzt Bezugsmethode: REST-Service / OAuth2]

Ressource: [wird vom ÖAMTC ergänzt]

Methode: [wird vom ÖAMTC ergänzt]

Authentifizierung: [wird vom ÖAMTC ergänzt]

Datenstruktur / Datendokumentation

Aktualisierungsintervalle

- Statische Daten: 2 x / Jahr in fixen Intervallen, zusätzlich bei Bedarf (z.B. Neu-Errichtung einer Parkgarage / Kurzparkzone, Ausbau / Erweiterung der Anlage / Kurzparkzone etc.)
- Dynamische Auslastungsdaten der Park & Ride-Anlagen und Parkgaragen (= Belegungsdaten im Knoten utilization): alle 10 Min

Anlage 2

IP-Adressen, mit denen der Datennutzer den Datenabgriff durchführt

[vom Datennutzer zu ergänzen]

Anlage 3

Darstellung der Quellenangabe ÖAMTC in den Ausgabemedien des Datennutzers

[Screenshot vom Datennutzer in Abstimmung mit dem ÖAMTC zu ergänze

ÖAMTC shorttermparkings API Documentation

Vers. 2025-09-30

Content

Shorttermparking Entity	1
Containers by vehicle	9
Containers by season	9
Comments	10
GIS Entity	10
Wrapped single entity	10
GIS Lists	21
Concepts	30
Includes	30
Paging and Sorting	30
Facets	30
Filtering	33
Query	40
Get a list of shorttermparkings	40
Retrieve shorttermparking by id	41
Search in shorttermparkings	42
Translations	50
Translate enums and boolean flags	50

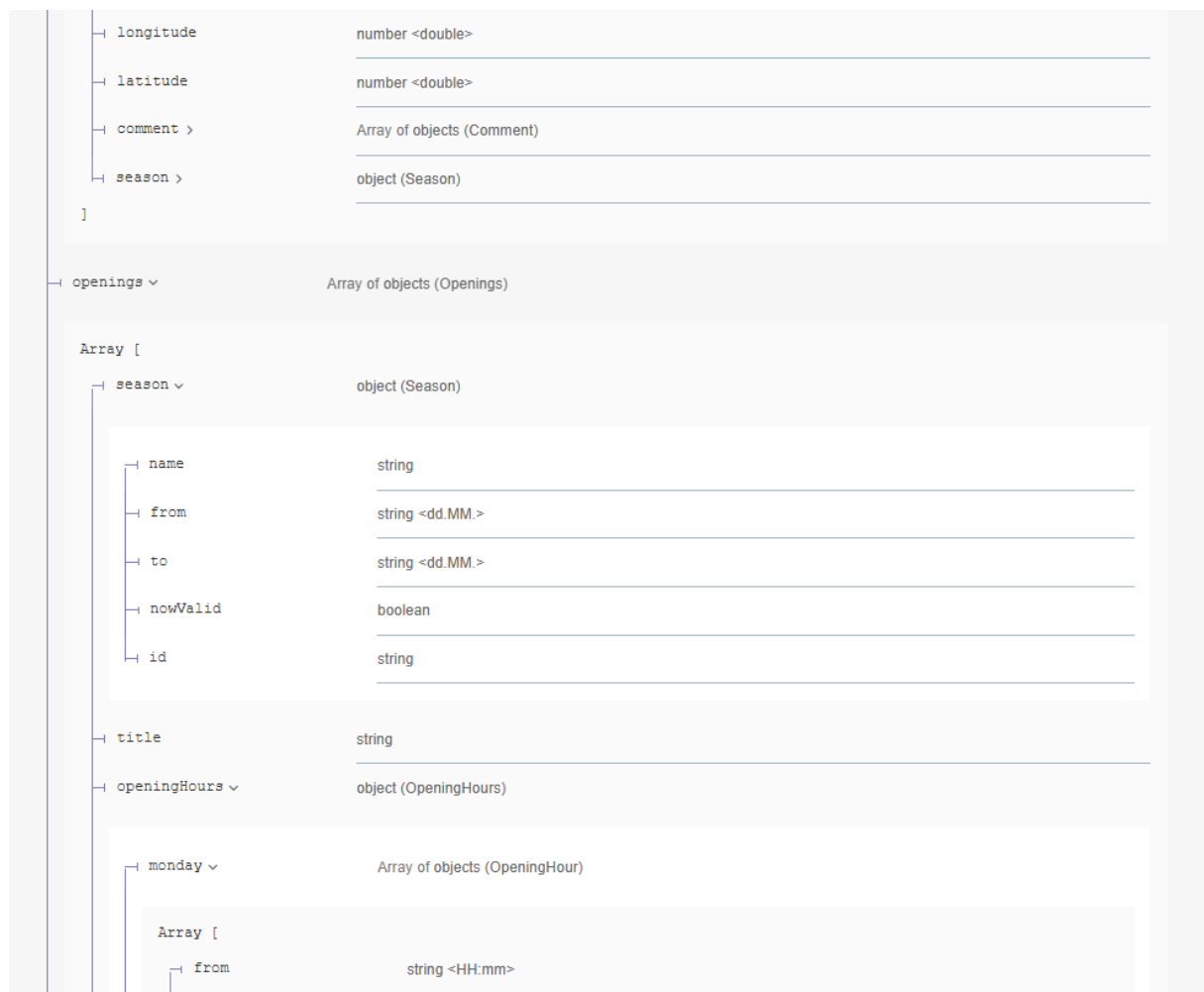
Shorttermparking entity

The shorttermparking entity describes a raw data item as imported from cms. This data type is reused in all usecases (import, detail result, list result).It groups content in containers:

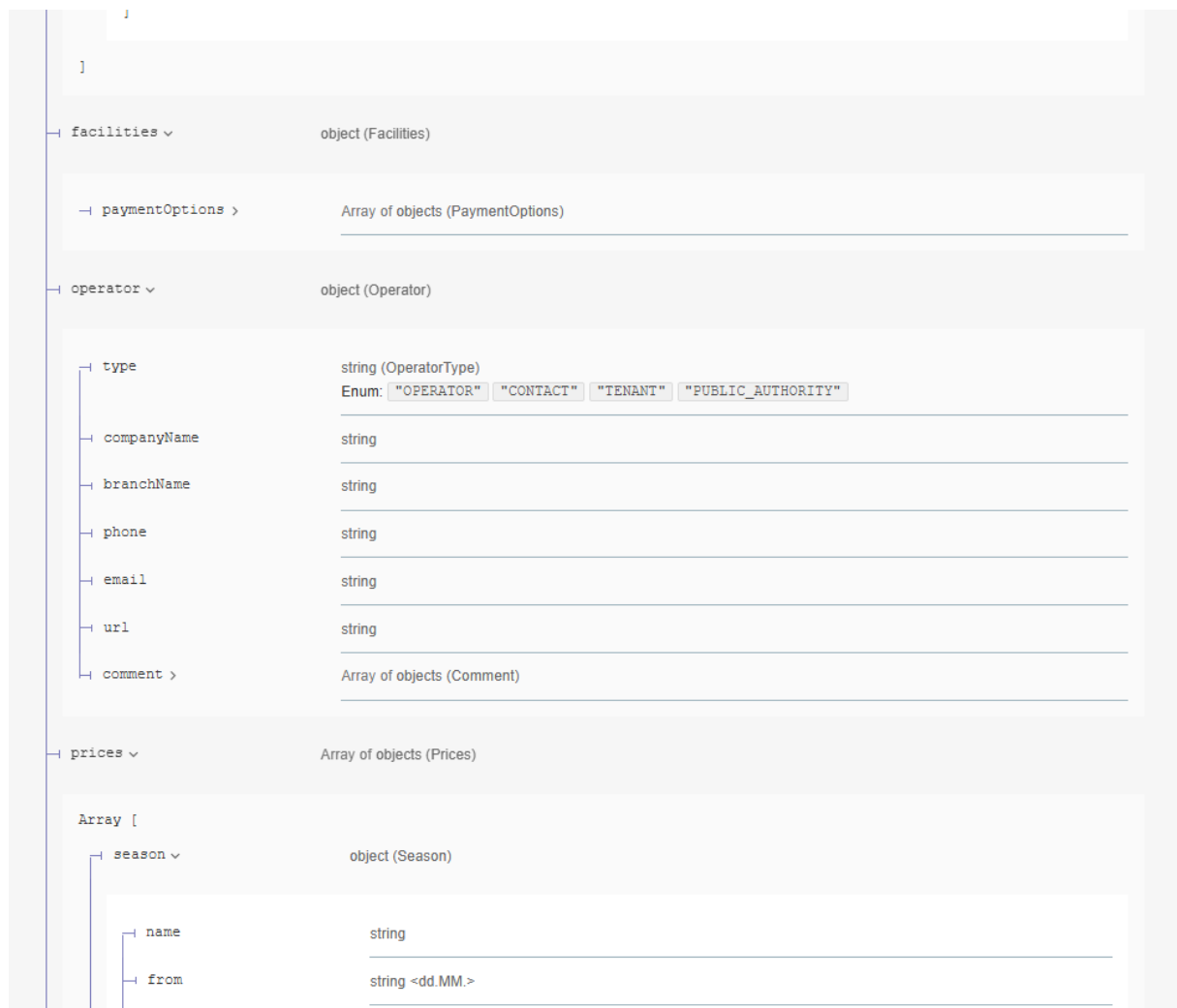
header ▾	object (Header)
id	string
lastUpdated	string <date-time>
resource	string (AResource) Value: "SHORTTERMPARKING"
categories	Array of strings Items Enum: "PUBLIC" "NON_PUBLIC"
names ▾	Array of objects (Name)
Array [type value]	string Enum: "NAME" "ALIAS1" "ALIAS2" "GROUP_NAME" string
quality ▾	object (Quality)
manuallyApproved	boolean
logos ▾	Array of objects (Picture)
Array [	

id	string										
source	string										
comment ▾	Array of objects (Comment)										
Array [<table> <tr> <td>lang</td> <td>string Enum: "GERMAN" "ENGLISH"</td> </tr> <tr> <td>mimetype</td> <td>string Enum: "text/plain" "text/html"</td> </tr> <tr> <td>content</td> <td>string</td> </tr> </table>]		lang	string Enum: "GERMAN" "ENGLISH"	mimetype	string Enum: "text/plain" "text/html"	content	string				
lang	string Enum: "GERMAN" "ENGLISH"										
mimetype	string Enum: "text/plain" "text/html"										
content	string										
text	string										
url	string										
]											
pictures ▾ Array of objects (Picture)											
Array [<table> <tr> <td>id</td> <td>string</td> </tr> <tr> <td>source</td> <td>string</td> </tr> <tr> <td>comment ▾</td> <td>Array of objects (Comment)</td> </tr> <tr> <td colspan="2"> Array [<table> <tr> <td>lang</td> <td>string Enum: "GERMAN" "ENGLISH"</td> </tr> </table>] </td> </tr> </table>]		id	string	source	string	comment ▾	Array of objects (Comment)	Array [<table> <tr> <td>lang</td> <td>string Enum: "GERMAN" "ENGLISH"</td> </tr> </table>]		lang	string Enum: "GERMAN" "ENGLISH"
id	string										
source	string										
comment ▾	Array of objects (Comment)										
Array [<table> <tr> <td>lang</td> <td>string Enum: "GERMAN" "ENGLISH"</td> </tr> </table>]		lang	string Enum: "GERMAN" "ENGLISH"								
lang	string Enum: "GERMAN" "ENGLISH"										

	mimetype	string Enum: "text/plain" "text/html"
	content	string
]		
	text	string
	url	string
]		
deepLinks >		Array of objects (DeepLink)
regions		Array of strings
addresses v		Array of objects (Addresses)
Array [		
	type	string Enum: "LOCATION" "POSTAL_ADDRESS" "ENTRANCE" "EXIT"
	name	string
	street	string
	houseNumber	string
	postalCode	string
	city	string
	county	string
	country	string



to string <HH:mm>	
]	
tuesday >	Array of objects (OpeningHour)
wednesday >	Array of objects (OpeningHour)
thursday >	Array of objects (OpeningHour)
friday >	Array of objects (OpeningHour)
saturday >	Array of objects (OpeningHour)
sunday >	Array of objects (OpeningHour)
holiday v	Array of objects (OpeningHour)
Array [from string <HH:mm> to string <HH:mm>]	
comment v	Array of objects (Comment)
Array [lang string Enum: "GERMAN" "ENGLISH" mimetype string Enum: "text/plain" "text/html" content string]	



to	string <dd.MM.>
nowValid	boolean
id	string
prices	Array of objects (Price)
<pre> Array [vehicle > unit price comment > freeMinutes] </pre>	<pre> object (Vehicle) string (PriceUnit) Enum: "PER_HOUR" "PER_HALFHOUR" number Array of objects (Comment) number </pre>
mobileParkings	Array of objects (MobileParkings)
<pre> Array [type isAvailable isPartner isCoopPartner url] </pre>	<pre> string Enum: "HANDY_PARKEN" "EASY_PARK" "PARK_NOW" "PARK_AND_MORE" "WIENERNEUSTADTMOBILESPARKEN" boolean boolean boolean string </pre>

comment ▾

Array of objects (Comment)

Array [

lang

string

Enum: "GERMAN" "ENGLISH"

mimetype

string

Enum: "text/plain" "text/html"

content

string

]

]

geoData ▾

object (Geodata)

wkt

string

coordinates ▾

Array of objects (Coordinate)

Array [

longitude

number <double>

latitude

number <double>

]

maxDuration

integer <int32>

displayColor

string

Containers by vehicle

Some container data is available multiple times, grouped by a vehicle property indicating different values for different vehicle types. A vehicle is defined like this:

type	string (VehicleType) Enum: "CAR" "ELECTRIC_CAR" "TRUCK" "MOTORCYCLE" "ELECTRIC_MOTORCYCLE" "BUS" "CAMPER" "TRAILER" "BIKE" "ELECTRIC_BIKE"
maxLength	number <double> Maximal length in meters
maxHeight	number <double> Maximal height in meters
maxWidth	number <double> Maximal width in meters
maxWeight	number <double> Maximal weight in tons

Containers by season

Some container data is available multiple times, grouped by a season property to list items describing the validity as date range during the year. A season is defined like this:

name	string
from	string <dd.MM.>
to	string <dd.MM.>
nowValid	boolean
id	string

Comments

Optional comments throughout the data may be defined per language and mime type.

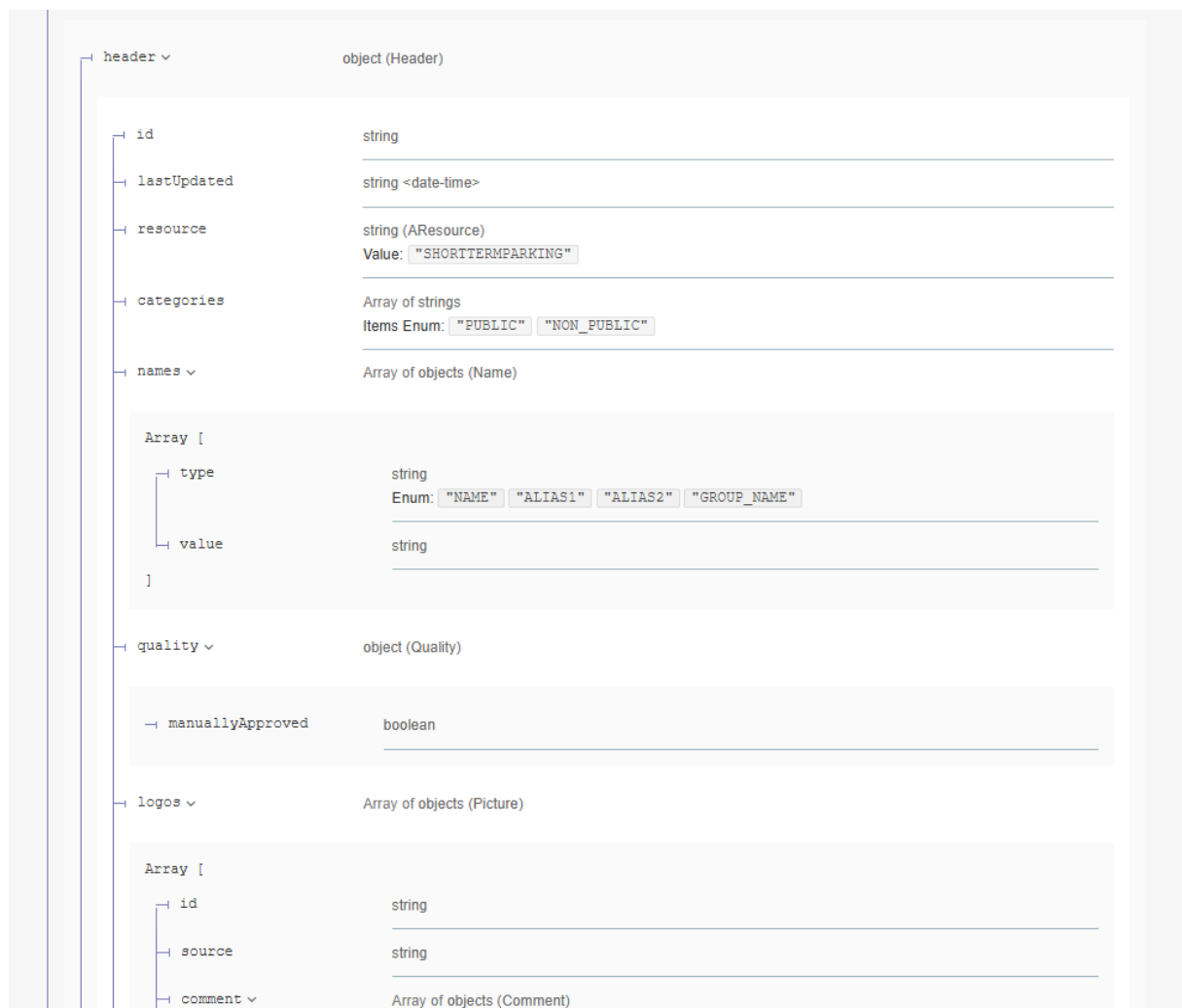
lang	string Enum: "GERMAN" "ENGLISH"
mimetype	string Enum: "text/plain" "text/html"
content	string

GIS Entity

Wrapped single entity

A GIS detail (single) result wraps a raw shorttermparking model in `data` and adds two GIS specific top level objects `header` and `computed`:

header	object (ShorttermparkingResultHeader)
lastImported	string <date-time>
lastImportedHuman	string
etag	string
url	string
data	object (Shorttermparking)



Array [lang string Enum: "GERMAN" "ENGLISH" mimetype string Enum: "text/plain" "text/html" content string	
]	
text	string
url	string
]	
pictures	Array of objects (Picture)
Array [id string source string comment Array of objects (Comment) text string url string	
]	
deepLinks	Array of objects (DeepLink)
Array [title string	

url

string

]

regions

Array of strings

addresses

Array of objects (Addresses)

Array [

type

string

Enum: "LOCATION" "POSTAL_ADDRESS" "ENTRANCE" "EXIT"

name

string

street

string

houseNumber

string

postalCode

string

city

string

county

string

country

string

longitude

number <double>

latitude

number <double>

comment

Array of objects (Comment)

Array [

lang

string

Enum: "GERMAN" "ENGLISH"

mimetype

string

Enum: "text/plain" "text/html"

openingHours ▾	object (OpeningHours)
monday ▾ Array [from to]	Array of objects (OpeningHour) string <HH:mm> string <HH:mm>
tuesday >	Array of objects (OpeningHour)
wednesday >	Array of objects (OpeningHour)
thursday >	Array of objects (OpeningHour)
friday >	Array of objects (OpeningHour)
saturday >	Array of objects (OpeningHour)
sunday >	Array of objects (OpeningHour)
holiday ▾	Array of objects (OpeningHour)
Array [from to]	string <HH:mm> string <HH:mm>
comment >	Array of objects (Comment)

1

season ▾

object (Season)

name

string

from

string <dd.MM.>

to

string <dd.MM.>

nowValid

boolean

id

string

prices ▾

Array of objects (Price)

Array [

vehicle >

object (Vehicle)

unit

string (PriceUnit)

Enum: "PER_HOUR" "PER_HALFHOUR"

price

number

comment >

Array of objects (Comment)

freeMinutes

number

]

mobileParkings ▾

Array of objects (MobileParkings)

Array [

type

string

Enum: "HANDY_PARKEN" "EASY_PARK" "PARK_NOW" "PARK_AND_MORE"

		"WIENERNEUSTADTMOBILES PARKEN"
isAvailable	boolean	
isPartner	boolean	
isCoopPartner	boolean	
url	string	
comment >	Array of objects (Comment)	
}		
geoData v	object (Geodata)	
wkt	string	
coordinates >	Array of objects (Coordinate)	
maxDuration	integer <int32>	
displayColor	string	
computed v	object (ShorttermparkingResultComputed)	
displayAddress v	object (Address)	
type	string	Enum: "LOCATION" "POSTAL_ADDRESS" "ENTRANCE" "EXIT"
name	string	
street	string	
houseNumber	string	

postalCode	string
city	string
county	string
country	string
longitude	number <double>
latitude	number <double>
comment	Array of objects (Comment)
Array [lang string Enum: "GERMAN" "ENGLISH" mime-type string Enum: "text/plain" "text/html" content string	
season	object (Season)
name string from string <dd.MM.> to string <dd.MM.> nowValid boolean id string	

maxOpeningHours ▾

object (OpeningHours)

monday ▾

Array of objects (OpeningHour)

Array [

from

string <HH:mm>

to

string <HH:mm>

]

tuesday >

Array of objects (OpeningHour)

wednesday >

Array of objects (OpeningHour)

thursday >

Array of objects (OpeningHour)

friday >

Array of objects (OpeningHour)

saturday >

Array of objects (OpeningHour)

sunday >

Array of objects (OpeningHour)

holiday ▾

Array of objects (OpeningHour)

Array [

from

string <HH:mm>

to

string <HH:mm>

]

openingInformation ▾

object (OpeningInformation)

Information about opening status based on current time.

openAllDay	boolean	This Poi is open 0-24h today.
openNextHour	boolean	This Poi is min 60 minutes from now.
minutesLeftOpen	number	Exact minutes left open in current opening slot.
minutesLeftClosed	number	Exact minutes left closed until next opening slot.
centerPoint	object (Coordinate)	
longitude	number <double>	
latitude	number <double>	
distanceToUserCoordinates	number <double>	Distance to supplied user coordinates in km
lastImported	string <date-time>	Last imported timestamp.

GIS Lists

A GIS list result represents an array of ShorttermparkingResults in `results` and adds two list specific top level objects `header` and `computed`:

header	object (ShorttermparkingResultListHeader)	
lastImported	string <date-time>	
etag	string	
start	integer	
count	integer	
totalCount	integer	
sortField	string	
sortOrder	string	
includeFacets	boolean	
results	Array of objects (ShorttermparkingResult)	
Array [		
header	object (ShorttermparkingResultHeader)	
lastImported	string <date-time>	
lastImportedHuman	string	
etag	string	

url	string
data	object (Shorttermparking)
header	object (Header)
id	string
lastUpdated	string <date-time>
resource	string (AResource) Value: "SHORTTERMPARKING"
categories	Array of strings Items Enum: "PUBLIC" "NON_PUBLIC"
names	Array of objects (Name)
Array [	
type	string Enum: "NAME" "ALIAS1" "ALIAS2" "GROUP_NAME"
value	string
]	
quality	object (Quality)
manuallyApproved	boolean
logos	Array of objects (Picture)
Array [	

id

string

source

string

comment ▾

Array of objects (Comment)

Array [

lang

string

Enum: "GERMAN" "ENGLISH"

mimetype

string

Enum: "text/plain" "text/html"

content

string

text

string

url

string

pictures ▾

Array of objects (Picture)

Array [

id

string

source

string

comment ▾

Array of objects (Comment)

Array [

lang

string

Enum: "GERMAN" "ENGLISH"

mimetype

string

Enum: "text/plain" "text/html"

content

string

text

string

url

string

deepLinks

Array of objects (DeepLink)

Array [

title

string

url

string

regions

Array of strings

addresses

Array of objects (Addresses)

Array [

type

string

Enum: "LOCATION" "POSTAL_ADDRESS" "ENTRANCE" "EXIT"

name

string

street

string

houseNumber

string

postalCode

string

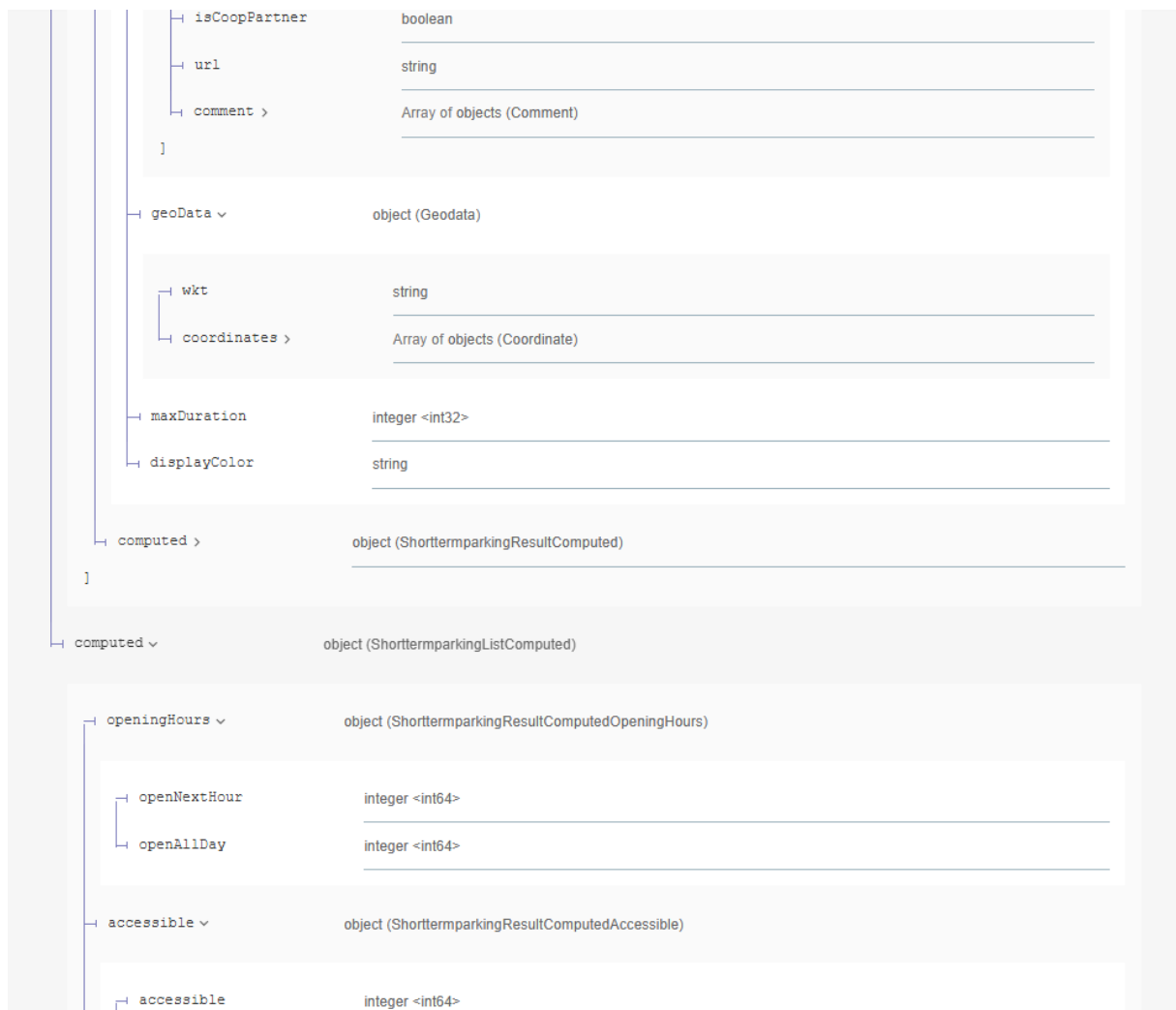
city

string

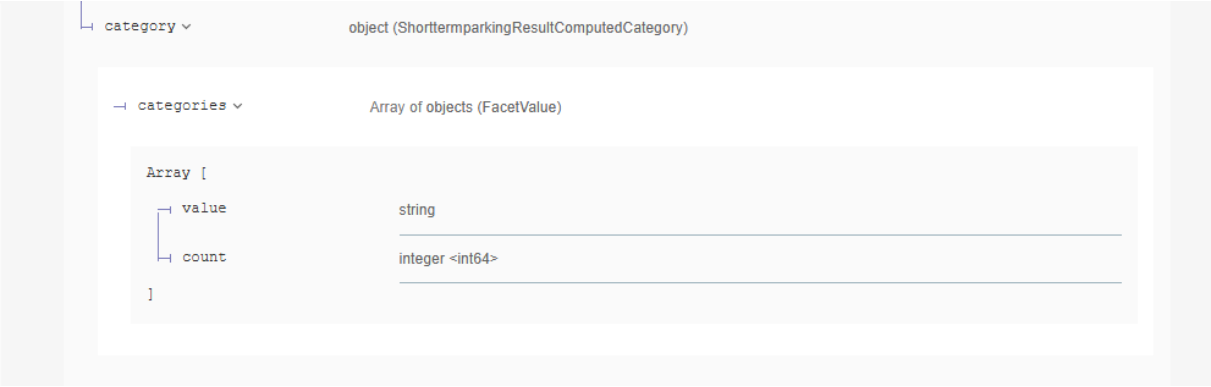
county	string
country	string
longitude	number <double>
latitude	number <double>
comment >	Array of objects (Comment)
season >	object (Season)
]	
openings v	Array of objects (Openings)
Array [	
season v	object (Season)
name	string
from	string <dd.MM.>
to	string <dd.MM.>
nowValid	boolean
id	string
title	string
openingHours v	object (OpeningHours)
monday v	Array of objects (OpeningHour)

Array [from string <HH:mm> to string <HH:mm>	
]	
tuesday >	Array of objects (OpeningHour)
wednesday >	Array of objects (OpeningHour)
thursday >	Array of objects (OpeningHour)
friday >	Array of objects (OpeningHour)
saturday >	Array of objects (OpeningHour)
sunday >	Array of objects (OpeningHour)
holiday v	Array of objects (OpeningHour)
Array [from string <HH:mm> to string <HH:mm>	
]	
comment >	Array of objects (Comment)
]	
facilities v	object (Facilities)
paymentOptions >	Array of objects (PaymentOptions)

operator ▾	object (Operator)
type string (OperatorType) Enum: "OPERATOR" "CONTACT" "TENANT" "PUBLIC_AUTHORITY"	
companyName	string
branchName	string
phone	string
email	string
url	string
comment >	Array of objects (Comment)
prices ▾	Array of objects (Prices)
Array [season > object (Season) prices > Array of objects (Price)]	
mobileParkings ▾	Array of objects (MobileParkings)
Array [type string Enum: "HANDY_PARKEN" "EASY_PARK" "PARK_NOW" "PARK_AND_MORE" "WIENERNEUSTADTMOBILESPARKEN" isAvailable boolean isPartner boolean	



handicappedParking	integer <int64>
locations >	object (ShorttermparkingResultComputedLocation)
prices v	object (ShorttermparkingResultComputedPrices)
ranges v	Array of objects (PriceRangeFacetValue)
<pre> Array [{ vehicle > unit from to count }]</pre>	<pre> object (Vehicle) string (PriceUnit) Enum: "PER_HOUR" "PER_HALFHOUR" number <double> number <double> integer <int64></pre>
operator v	object (ShorttermparkingResultComputedOperator)
companyName v	Array of objects (FacetValue)
<pre> Array [{ value count }]</pre>	<pre> string integer <int64></pre>



Concepts

Includes

Per default, full result objects are returned on every request. To reduce the payload in single and list requests, add the `include` parameter to select only parts that are needed. for example:

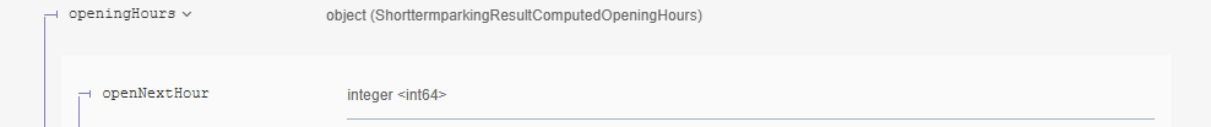
Request	Description
<code>/shorttermparking/detail/pg-000837DDC607B24?include=DATA_HEADER,DATA_RATINGS`</code>	return only <code>/data/header</code> and <code>/data/ratings</code>
<code>/shorttermparking/detail/pg-000837DDC607B24?include=DETAIL_HEADER,DATA`</code>	return only <code>/header</code> and <code>/data/*</code>
<code>/shorttermparking/list?include=LIST_COMPUTED`</code>	return only <code>/computed</code> facet section

Paging and Sorting

Both list and search request support paging using get paramaters `start`, `count` and `sort`. See sort enum for available sort options.

Facets

Both list and search responses automatically add computed facet values to the response. Facet values are computed over all possible matches, not only the ones returned in the list slice when using paging.



openAllDay	integer <int64>
accessible ▾	object (ShorttermparkingResultComputedAccessible)
accessible	integer <int64>
handicappedParking	integer <int64>
locations ▾	object (ShorttermparkingResultComputedLocation)
country ▾	Array of objects (FacetValue)
Array [	
value	string
count	integer <int64>
]	
country ▾	Array of objects (FacetValue)
Array [	
value	string
count	integer <int64>
]	
prices ▾	object (ShorttermparkingResultComputedPrices)

ranges	Array of objects (PriceRangeFacetValue)
<pre> Array [vehicle unit from to count] </pre>	<pre> object (Vehicle) string (PriceUnit) Enum: "PER_HOUR" "PER_HALFHOUR" number <double> number <double> integer <int64> </pre>
operator	object (ShorttermparkingResultComputedOperator)
companyName	Array of objects (FacetValue)
<pre> Array [value count] </pre>	<pre> string integer <int64> </pre>
category	object (ShorttermparkingResultComputedCategory)
categories	Array of objects (FacetValue)
<pre> Array [value </pre>	<pre> string </pre>

count	integer <int64>
-------	-----------------

Filtering

The search request inherits all features of list and adds the possibility to post one or more filters. See request body schema of search for available filters.

ids	object (IdsSelections)
list	Array of strings
geographic	object (GeographicSelections)
boundingBox	object (GeographicBoundingBoxSelection)
longitudeMin required	number <double> WGS84 longitude lower border.
longitudeMax required	number <double> WGS84 longitude upper border.
latitudeMin required	number <double> WGS84 latitude lower border.
latitudeMax required	number <double> WGS84 latitude upper border.

	required	WGS84 latitude lower border.
	latitudeMax required	number <double> WGS84 latitude upper border.
circle ▾		object (GeographicCircleSelection) Geographic nearby filter by center point and radius.
	centerLongitude required	number <double> WGS84 center longitude.
	centerLatitude required	number <double> WGS84 center latitude.
	radius required	number <double> Radius in km.
corridor ▾		object (GeographicCorridorSelection) Geographic filter items with max. distance to a wkt polyline.
	polylineWKT required	string Polyline (eg. Route) as WKT Linestring.
	radius required	number <double> Maximum normal distance from line in km.
location ▾		object (LocationSelections)
	city ▾	object (SelectionValueList)
	values	Array of strings

ignoredValues	Array of strings List of values to ignore.
operator	string Default: "OR" Enum: "OR" "AND"
country ▾	object (SelectionValueList)
values	Array of strings
ignoredValues	Array of strings List of values to ignore.
operator	string Default: "OR" Enum: "OR" "AND"
country ▾	object (SelectionValueList)
values	Array of strings
ignoredValues	Array of strings List of values to ignore.
operator	string Default: "OR" Enum: "OR" "AND"
postalCode ▾	object (SelectionValueList)
values	Array of strings

ignoredValues	Array of strings List of values to ignore.
operator	string Default: "OR" Enum: "OR" "AND"
region	object (SelectionValueList)
values	Array of strings
ignoredValues	Array of strings List of values to ignore.
operator	string Default: "OR" Enum: "OR" "AND"
category	object (SelectionValueList)
values	Array of strings
ignoredValues	Array of strings List of values to ignore.
operator	string Default: "OR" Enum: "OR" "AND"
operator	object (SelectionValueList)
values	Array of strings

ignoredValues	Array of strings List of values to ignore.
operator	string Default: OR Enum: OR AND

price ▾	object (PriceSelections)
---------	--------------------------

halfHour ▾	object (RangeSelection)
------------	-------------------------

from	number
to	number

hour ▾	object (RangeSelection)
--------	-------------------------

from	number
to	number

day ▾	object (RangeSelection)
-------	-------------------------

from	number
to	number

night ▾	object (RangeSelection)
---------	-------------------------

	from to	number number
night_halfHour ▾	object (RangeSelection)	
	from to	number number
night_hour ▾	object (RangeSelection)	
	from to	number number
weekend ▾	object (RangeSelection)	
	from to	number number
month ▾	object (RangeSelection)	
	from to	number number
month_night_only ▾	object (RangeSelection)	

from	number
to	number

season ▾	object (RangeSelection)
----------	-------------------------

from	number
to	number

year ▾	object (RangeSelection)
--------	-------------------------

from	number
to	number

vehicle ▾	object (RangeSelection)
-----------	-------------------------

from	number
to	number

openingHours ▾	object (OpeningHoursSelections)
----------------	---------------------------------

title	string
openNextHour	boolean
openAllDay	boolean
openHours >	object (TimeRangeSelection)

mobileParking ▾	object (SelectionValueList)
-----------------	-----------------------------

values	Array of strings
ignoredValues	Array of strings List of values to ignore.
operator	string Default: "OR" Enum: "OR" "AND"

optionalSeason ▾	object (OptionalSeasonSelections)
------------------	-----------------------------------

season	string
--------	--------

Query

Get a list of shorttermparkings

List shorttermparkings with GET request. Supports Paging, Includes and Facets

QUERY PARAMETERS

start	integer >= 0 Default: 0 Paging: start index
count	integer Default: 10 Paging: number of elements
sort	string Enum: "ID_ASC" "ID_DESC" "NAME_ASC" "NAME_DESC" "CITY_ASC" "CITY_DESC" "POSTAL_CODE_ASC" "POSTAL_CODE_DESC" "STREET_ASC" "STREET_DESC" "DISTANCE_ASC" Sorting
include	string Default: 'ALL' Enum: "ALL" "LIST_HEADER" "LIST_RESULTS" "LIST_COMPUTED" "DETAIL_HEADER" "DETAIL_COMPUTED" "DATA" "DATA_HEADER" "DATA_ADDRESS" "DATA_OPENINGS" "DATA_OPERATOR" "DATA_PRICES" "DATA_ACCESSIBILITY" "DATA_FACILITIES" "DATA_GEODATA" "DATA_UTILIZATION" "DATA_TRAFFIC_INFRASTRUCTURE" Includes the following top level fields: • ALL: All top level fields are included. • LIST_HEADER: Include header section for list - <code>header</code> • LIST_RESULTS: Include result (entities) in list - <code>results</code> • LIST_COMPUTED: Include computed section for list - <code>computed</code> • DETAIL_HEADER: Include header sections for entites - <code>results[].header</code> • DETAIL_COMPUTED: Include computed sections for entites - <code>results[].computed</code> • DATA: Include data (entities) - <code>results[].data</code> • DATA_HEADER: Include entity <code>data_header</code>

	• <code>DATA_ADDRESS</code> : Include entity <code>data.address</code> • <code>DATA_OPENINGS</code> : Include entity <code>data.openings</code> • <code>DATA_OPERATOR</code> : Include entity <code>data.operator</code> • <code>DATA_PRICES</code> : Include entity <code>data.prices</code> • <code>DATA_RATINGS</code> : Include entity <code>data.ratings</code> • <code>DDATA_ACCESSIBILITY</code> : Include entity <code>data.accessible</code> • <code>DATA_FACILITIES</code> : Include entity <code>data.facilities</code> • <code>DATA_GEODATA</code> : Include entity <code>data.geoData</code> • <code>DATA_TRAFFIC_INFRASTRUCTURE</code> : Include entity <code>data.trafficInfrastructure</code>
facets	boolean Default: <code>true</code> Add facet result in computed.
userCoordinates	string <wkt-coordinates> <code>POINT(-?(?:\d+ \d+(?:\d*)?) -?(?:\d+ \d+ </code> Show pattern User coordinates
HEADER PARAMETERS	
If-None-Match	string Use for ETag caching
Responses	
> 200 successful operation	
— 304 If header "If-None-Match" was supplied with an ETag value and the content for this request did not change	

PATH PARAMETERS

userCoordinates	string <wkt-coordinates> <code>POINT(-?(?:\d+ \d+(?:\d*)?)~?(?:\d+ \d+ </code> Show pattern User coordinates
HEADER PARAMETERS	
If-None-Match	string Use for ETag caching
Responses	
> 200 successful operation	
— 304 If header "If-None-Match" was supplied with an ETag value and the content for this request did not change	
— 400 Invalid ID	
— 404 Shorttermparking not found	
Search in shorttermparkings	
Search in shorttermparkings	
QUERY PARAMETERS	
start	integer 0 Default: 0 Paging: start index
count	integer Default: 10 Paging: number of elements

sort	string Enum: "ID_ASC" "ID_DESC" "NAME_ASC" "NAME_DESC" "CITY_ASC" "CITY_DESC" "POSTAL_CODE_ASC" "POSTAL_CODE_DESC" "STREET_ASC" "STREET_DESC" "DISTANCE_ASC" Sorting
include	string Default: "ALL" Enum: "ALL" "LIST_HEADER" "LIST_RESULTS" "LIST_COMPUTED" "DETAIL_HEADER" "DETAIL_COMPUTED" "DATA" "DATA_HEADER" "DATA_ADDRESS" "DATA_OPENINGS" "DATA_OPERATOR" "DATA_PRICES" "DATA_ACCESSIBILITY" "DATA_FACILITIES" "DATA_GEODATA" "DATA_UTILIZATION" "DATA_TRAFFIC_INFRASTRUCTURE" Includes the following top level fields: • ALL : All top level fields are included. • LIST_HEADER : Include header section for list - <code>header</code> • LIST_RESULTS : Include result (entities) in list - <code>results</code> • LIST_COMPUTED : Include computed section for list - <code>computed</code> • DETAIL_HEADER : Include header sections for entites - <code>results[].header</code> • DETAIL_COMPUTED : Include computed sections for entites - <code>results[].computed</code> • DATA : Include data (entities) - <code>results[].data</code> • DATA_HEADER : Include entity <code>data.header</code> • DATA_ADDRESS : Include entity <code>data.address</code> • DATA_OPENINGS : Include entity <code>data.openings</code> • DATA_OPERATOR : Include entity <code>data.operator</code> • DATA_PRICES : Include entity <code>data.prices</code> • DATA_RATINGS : Include entity <code>data.ratings</code> • DDATA_ACCESSIBILITY : Include entity <code>data.accessible</code> • DATA_FACILITIES : Include entity <code>data.facilities</code> • DATA_GEODATA : Include entity <code>data.geoData</code> • DATA_TRAFFIC_INFRASTRUCTURE : Include entity <code>data.trafficInfrastructure</code>
cluster	integer <code>>= 0</code> Default: <code>0</code> Cluster result in computed.
facets	boolean Default: <code>true</code> Add facet result in computed.

county ▾	object (SelectionValueList)
values	Array of strings
ignoredValues	Array of strings List of values to ignore.
operator	string Default: "OR" Enum: "OR" "AND"
country ▾	object (SelectionValueList)
values	Array of strings
ignoredValues	Array of strings List of values to ignore.
operator	string Default: "OR" Enum: "OR" "AND"
postalCode ▾	object (SelectionValueList)
values	Array of strings
ignoredValues	Array of strings List of values to ignore.
operator	string Default: "OR" Enum: "OR" "AND"

region ▾

object (SelectionValueList)

values

Array of strings

ignoredValues

Array of strings

List of values to ignore.

operator

string

Default: "OR"

Enum: "OR" "AND"

category ▾

object (SelectionValueList)

values

Array of strings

ignoredValues

Array of strings

List of values to ignore.

operator

string

Default: "OR"

Enum: "OR" "AND"

operator ▾

object (SelectionValueList)

values

Array of strings

ignoredValues

Array of strings

List of values to ignore.

operator

string

Default: "OR"

Enum: "OR" "AND"

price ▾	object (PriceSelections)	
halfHour ▾	object (RangeSelection)	
from	number	
to	number	
hour ▾	object (RangeSelection)	
from	number	
to	number	
day ▾	object (RangeSelection)	
from	number	
to	number	
night ▾	object (RangeSelection)	
from	number	
to	number	
night_halfHour ▾	object (RangeSelection)	
from	number	

	to	number
night_hour ▾	object (RangeSelection)	
	from	number
	to	number
weekend ▾	object (RangeSelection)	
	from	number
	to	number
month ▾	object (RangeSelection)	
	from	number
	to	number
month_night_only ▾	object (RangeSelection)	
	from	number
	to	number
season ▾	object (RangeSelection)	

from

number

to

number

year

object (RangeSelection)

from

number

to

number

vehicle

object (RangeSelection)

from

number

to

number

openingHours

object (OpeningHoursSelections)

title

string

openNextHour

boolean

openAllDay

boolean

openingHours

object (TimeRangeSelection)

from

string <HH:mm>

to

string <HH:mm>

mobileParking

object (SelectionValueList)

values

Array of strings

ignoredValues

Array of strings

List of values to ignore.

operator

string

Default: "OR"

Enum: "OR" "AND"

optionalSeason

object (OptionalSeasonSelections)

season

string

Responses

> 200 successful operation

— 400 Invalid selections object

Translations

Translate Enums and boolean flags

Get localized translations for enums and boolean flags

PATH PARAMETERS

→ language required	string Language for the internationalized enum values
------------------------	--

HEADER PARAMETERS

→ If-None-Match	string Use for ETag caching
-----------------	--------------------------------

Responses

> 200 successful operation

— 404 No internationalization defined for language