# Renown Health Developer API GuidelinesProvider Directory Query API

Updated: Monday, November 13, 2023

Contents

[Renown Health Developer API Guidelines Provider Directory Query API 1](#_Toc150766998)

[Summary 2](#_Toc150766999)

[API Access 2](#_Toc150767000)

[GraphQL Schema’s for the Hometown Health Provider Directory 2](#_Toc150767001)

[GraphQL Query/Response 5](#_Toc150767002)

[Query: 5](#_Toc150767003)

[JSON Response: 6](#_Toc150767004)

|  |
| --- |
| SummaryThe purpose of this document is to provide an overview of data access via the Renown developer service API’s. This document will cover access into the Hometown Health Provider Directory data. All the public API’s are RESTful micro-service patterns accessed via GraphQL query http POST requests to a web API with a single endpoint of access. Using GraphQL abstraction provides a layer with many queries via the one endpoint. API AccessThe Provider Directory API endpoint is located on Renown/Hometown Health web servers with 7/24 availability. URI: <https://apps.hometownhealth.com/Services/_V3.0/ProviderDirectory/gql>Access to the GraphQL endpoints is done with a http POST request to the endpoint. Application like Postman® or Altair® client can be used for testing the queries. The response data returned by the query is a JSON file. GraphQL information can be found here <https://graphql.org/learn/> and by searching the internet for how to query a GraphQL endpoint.GraphQL Schema’s for the Hometown Health Provider DirectoryAvailable queries* getPlans:
	+ [PlanGraphType] **model available fields**
		- planName: String!
		- PlanDescription: String!
		- planCode: String!
* getSpecialties:
	+ [SpecialtyGraphType] **model available fields**
		- specialtyCode: String!
		- specialtyName: String!
* getSearchResults
	+ input arguments
		- (SearchText: String
		- searchTextLocation: String
		- searchLatitude: String
		- searchLongitude: String
		- zipcode: String
		- plan: String
		- providerType: String
		- acceptNewpatient: String
		- county: String
		- specialty: String
		- language: String
		- city: String
		- providerLastName: String
		- sortBy: String
		- maxDistance: String
		- directoryNames: String
		- providerSource: String
		- showLevenResults: String
		- wca: String
		- gender: String
		- removePharma: String
		- skip: Inttake: Int):
	+ [SearchResultGraphType] ] **model available fields**
		- providerID: String!
		- lastName: String!
		- firstName: String
		- mI: String
		- title: String
		- name: String!
		- nameLFMT: String!
		- gender: String
		- status: String!
		- acceptNewPatient: String!
		- specialties: [String]
		- locations: [String]
		- institutions: [String]
		- languages: [String]
		- nearestLatitude: Float
		- nearestLongitude: Float
		- nearestDistance: Decimal
		- weight: Float
		- imageUri: String!
		- additionalComments: [String]
		- isConciergeMedicineProvider: String
		- isHomeVisitsOnlyProvider: String!
		- totalRecordCount: Int!
		- effectiveDate: Date
		- terminationDate: Date
		- addresses: [String]
		- groupID: String
		- distanceToNearest: Decimal
		- phones: [String]
		- formattedPhones: [String]
		- isWCA: Boolean
		- mSOWImageurl: String
		- mSOWPhyID: String
		- mSOWFellowship: String
		- mSOWUniversityName: String
		- mSOWDegreeEarned: String
		- mSOWResidency: String
		- mSOWLanguagesSpoken: String
		- mSOWBio: String
		- mSOWVideoLink: String
		- mSOWEmployed: String
		- mSOWAreaofExpertise: String
		- mSOWGoverningboard: String
		- mSOWMemberofWCA: String
		- mSOWMemberofUNRMedSchool: String
		- showLevenResult: String
		- showBehavioralHealth: String
* getProviderDetails
	+ Input arguments
		- (id: String):
	+ [ProviderDetailGraphType]
		- providerID: String!
		- firstName: String
		- lastName: String!
		- name: String!
		- status: String!
		- specialties: [String!]!
		- languages: [String!]!
		- providerTypes: [String!]!
		- additionalComments: [String!]!
		- isConciergeMedicineProvider: String!
		- isHomeVisitsOnlyProvider: String!
		- groupID: String
		- facilities(id: Int):
			* [FacilityDetailGraphType]
				+ id: String!
				+ name: String
				+ address1: String
				+ address2: String
				+ city: String
				+ state: String
				+ zipCode: String
				+ county: String
				+ phone: String
				+ fax: String
				+ cityStateZip: String
				+ formattedPhone: String
				+ formattedFax: String
				+ webSite: String
				+ latitude: Float
				+ longitude: Float
				+ showPanelStatus: Boolean!
				+ hoursSunday: String
				+ hoursMonday: String
				+ hoursTuesday: String
				+ hoursWednesday: String
				+ hoursThursday: String
				+ hoursFriday: String
				+ hoursSaturday: String
				+ effectiveDate: Date
				+ terminationDate: Date
				+ facilityPlans(id: Int):

 [FacilityPlanGraphType]groupNumber: String!plan: String!acceptNewPatient: String* + - imageUri: String!
		- isWCA: Boolean
		- mSOWImageurl: String
		- mSOWPhyID: String
		- mSOWFellowship: String
		- mSOWUniversityName: String
		- mSOWDegreeEarned: String
		- mSOWResidency: String
		- mSOWLanguagesSpoken: String
		- mSOWBio: String
		- mSOWVideoLink: String
		- mSOWEmployed: String
		- mSOWAreaofExpertise: String
		- mSOWGoverningboard: String
		- mSOWMemberofWCA: String
		- mSOWMemberofUNRMedSchool: String

GraphQL Query/ResponseHere is an example Graphql query/response using the getSearchResults method.Query:{ results: getSearchResults(searchText: "smith", acceptNewpatient: "Open", sortBy: "Best Match", skip: 0, take: 12) { providerID groupID lastName firstName mI title nameLFMT specialties gender nearestLatitude nearestLongitude distanceToNearest imageUri totalRecordCount specialties locations institutions addresses status acceptNewPatient formattedPhones additionalComments mSOWImageurl mSOWPhyID mSOWFellowship mSOWUniversityName mSOWDegreeEarned mSOWResidency mSOWLanguagesSpoken mSOWBio mSOWVideoLink mSOWEmployed mSOWAreaofExpertise mSOWGoverningboard mSOWMemberofWCA mSOWMemberofUNRMedSchool showLevenResult } }JSON Response:{ "data": { "results": [ { "providerID": "082529", "groupID": "2466", "lastName": "Smith", "firstName": " David E", "mI": null, "title": "MD", "nameLFMT": "Smith, David E, MD", "specialties": [ "Cardiovascular Disease (Cardiology)" ], "gender": "M", "nearestLatitude": 39.5268636, "nearestLongitude": -119.7933571, "distanceToNearest": 0.00, "imageUri": "https://www.skillexpertiseandtechnology.com/providerMedia/images/Photos/David\_Smith\_\_280x280\_RGB.jpg", "totalRecordCount": 115, "locations": [ "RENO, NV 89502" ], "institutions": [ " Renown Institute for Heart And Vascular Health" ], "addresses": [ "1500 E 2ND ST #400" ], "status": "Active", "acceptNewPatient": "Accepting new patients", "formattedPhones": [ "775-982-2400" ], "additionalComments": [ "Western Clinical Alliance Provider" ], "mSOWImageurl": "https://www.skillexpertiseandtechnology.com/providerMedia/images/Photos/David\_Smith\_\_280x280\_RGB.jpg", "mSOWPhyID": "617", "mSOWFellowship": "Harbor-UCLA Medical Center", "mSOWUniversityName": "NYU School of Medicine", "mSOWDegreeEarned": "N", "mSOWResidency": "UC San Diego School of Medicine", "mSOWLanguagesSpoken": "English", "mSOWBio": "Specializing in electrophysiology, cardiac rhythm abnormalities, lead extractions, pacemakers and defibrillator placement.", "mSOWVideoLink": "", "mSOWEmployed": "Employed", "mSOWAreaofExpertise": "Cardiology", "mSOWGoverningboard": "American Board of Internal Medicine, American Board of Internal Medicine, American Board of Internal Medicine", "mSOWMemberofWCA": "", "mSOWMemberofUNRMedSchool": "", "showLevenResult": "" },… more looped data  |
|  |