

# OGC API – Routes

---

Winter School 2022



*Ms. Prajwalita J. Chavan*  
*IIT, Bombay*

# Overview

---

- About OGC API – Routes
- Operations
- Requirements classes defining resources
- Requirements classes defining routes
- Requirements classes defining conformance
- Get Route
- Delete Route
- Define Route
- Intermediate waypoint
- Route Mode
- Vehicle Height
- Vehicle Weight
- Obstacle
- Temporal Constraint
- Route Exchange Model

# OGC API - Routes

---

- **Publication Date:** not published
- **Submitter:** Clemens Portele
- Core standard specifies the fundamental API building blocks for interacting with the **"route" resources**.
- **Routing** is one of the most widely used geospatial operations in the world today.
- Accordingly, any routing API will accept **two waypoints, the start and end position**. APIs are free to **find routes within a single or across multiple environments**, support **additional waypoints between the start and end positions**, and be **extensible to support parameters** required by diverse implementation communities.

# Introduction

---

- A Web API that allows applications to request routes in a manner independent of the underlying routing data set, routing engine or algorithm.

This standard defines modular API building blocks to:

- compute **new routes**,
- specify **additional**, commonly used **routing parameters** in addition to the **start and end point of the route**,
- **communicate routes synchronously or asynchronously** to clients,
- support **callbacks**,
- **delete routes** stored on the server,
- request **different profiles of the route** (complete, overview, by segment).



# Operations

---

The bare minimum requirement for a Routing API is to **compute a route** based on a start and end point. The resulting route is encoded using the **Route Exchange Model (REM)**.

Besides this basic requirement, **additional optional requirements** to the API are: **Storage and management of computed routes**

Additionally, a Routing API may support the following optional routing parameters:

- Capability to **add additional intermediate waypoints**.
- Select a **transport mode** for the route computation.
- Apply a **height restriction** to the route computation.
- Apply a **weight restriction** to the route computation.
- Ability to define **obstacles** that have to be avoided.
- Specify **departure or arrival times**.

# Requirements classes defining resources

---

Resource	Path	HTTP method	Request media type	Response media type	Document reference
Landing page	/	GET	n/a	application/json	<a href="#">API landing page</a>
Conformance declaration	/conformance	GET	n/a	application/json	<a href="#">Declaration of conformance classes</a>
Routes	/routes	POST	application/json	application/geo+json	<a href="#">Compute a new route</a>
		GET	n/a	application/json	<a href="#">Fetch routes</a>
Route	/routes/{routeId}	GET	n/a	application/geo+json	<a href="#">Fetch a route</a>
		DELETE	n/a	n/a	<a href="#">Delete a route</a>
Route definition	/routes/{routeId}/definition	GET	n/a	application/json	<a href="#">Fetch the definition of a route</a>

# Landing page

---

Requirement 1	<code>/req/core/landing-page-link</code>
A	The content of the landing page response SHALL include a link to the Routes resource at path <code>/routes</code> (link relation type <code>http://www.opengis.net/def/rel/ogc/1.0/routes</code> ).

# Requirements classes defining routes

Requirement 2	<b>/req/core/compute-route-op</b>
A	The server SHALL support the HTTP POST operation at the path <code>/routes</code> .
B	<p>The server SHALL accept a route definition in the content of the request based upon the following OpenAPI 3.0 schema:</p> <pre data-bbox="682 508 2275 1400">type: object required:   - inputs properties:   inputs:     type: object     required:       - waypoints     properties:       name:         type: string       waypoints:         type: object         required:           - value         properties:           value:             type: object             required:               - type               - coordinates             properties:               type:                 type: string                 enum:                   - MultiPoint               coordinates:                 type: array                 minItems: 2</pre>

# Requirements classes conformance

Requirement 3	<code>/req/core/conformance-values</code>
A	<p>The content of the conformance declaration response at path <code>/conformance</code> SHALL list all values that the <code>preference</code> parameter supports, based upon the following OpenAPI 3.0 schema:</p> <pre data-bbox="667 491 2137 1290">type: object required:   - properties properties:   properties:     type: object     required:       - http://www.opengis.net/spec/ogcapi-routes-1/1.0.0-draft.1/conf/core     properties:       http://www.opengis.net/spec/ogcapi-routes-1/1.0.0-draft.1/conf/core:         type: object         required:           - preferences         properties:           preferences:             type: array             items:               minItems: 1               type: string</pre> <p style="text-align: right;">YAML</p>

# Get Routes

---

<b>Requirement 7</b>	<b>/req/manage-routes/get-routes-op</b>
A	The server SHALL support the HTTP GET operation at the path <code>/routes</code> .
<b>Requirement 8</b>	<b>/req/manage-routes/get-routes-success</b>
A	A successful execution of the operation SHALL be reported as a response with a HTTP status code <code>200</code> .
B	<p>The content of that response SHALL be based upon the following OpenAPI 3.0 schema:</p> <pre>type: object properties:   links:     type: array     items:       type: object       required:         - rel         - href       properties:         href:           type: string         rel:           type: string         type:           type: string         hreflang:</pre> <span style="float: right;">YAML</span>



# Delete Route

---

<b>Requirement 11</b>	<b>/req/manage-routes/delete-route-op</b>
A	The server SHALL support the HTTP DELETE operation at the path <code>/routes/{routeId}</code> for each route referenced from the Routes resource at <code>/routes</code> .
<b>Requirement 12</b>	<b>/req/manage-routes/delete-route-success</b>
A	A successful execution of the operation SHALL be reported as a response with a HTTP status code <code>200</code> or <code>204</code> .
B	If the operation is not executed immediately, but is added to a processing queue, the response SHALL have a HTTP status code <code>202</code> .

# Define Route

---

Requirement 13	<b><code>/req/manage-routes/route-definition-op</code></b>
A	The server SHALL support the HTTP GET operation at the path <code>/routes/{routeId}/definition</code> for each route referenced from the Routes resource at <code>/routes</code> .
Requirement 14	<b><code>/req/manage-routes/route-definition-success</code></b>
A	A successful execution of the operation SHALL be reported as a response with a HTTP status code <code>200</code> .
B	The content of that response SHALL be identical to the content of the POST request to <code>/routes</code> when the route was created.



# Intermediate Waypoints

---

<b>Requirement 15</b>	<b><i>/req/intermediate-waypoints/input</i></b>
A	The server SHALL support at least five points in the member with the name "waypoints" in the route definition in a HTTP POST request to the path <code>/routes</code> (i.e. <code>maxItems</code> may be removed from the schema definition or increased to a value larger than '4').
<b>Requirement 16</b>	<b><i>/req/intermediate-waypoints/success</i></b>
A	The computed route SHALL pass through all waypoints in the order in which they have been provided. "Pass through" means that the route overview line string geometry passes through the position or a position on the route network that is close to the waypoint.

# Define Route mode

---

- On Road
- Off Road

Requirement 17	<code>/req/mode/input</code>
A	<p>The server SHALL support a member at the JSON Pointer <code>/inputs/mode</code> in the route definition in a HTTP POST request to the path <code>/routes</code> based on the following schema:</p> <pre>type: string</pre> <p style="text-align: right;">YAML</p>

# Specify Vehicle Height

---

Requirement 19	<code>/req/height/input</code>
A	<p>The server SHALL support a member at the JSON Pointer <code>/inputs/height</code> in the route definition in a HTTP POST request to the path <code>/routes</code> based on the following schema:</p> <pre>type: number minimum: 0</pre>

# Specify Vehicle Weight

---

Requirement 21	<code>/req/weight/input</code>
A	<p>The server SHALL support a member at the JSON Pointer <code>/inputs/weight</code> in the route definition in a HTTP POST request to the path <code>/routes</code> based on the following schema:</p> <pre>type: number minimum: 0</pre> <p>YAML</p>

# Specify Obstacles

Requirement 23	/req/obstacles/input
A	<p>The server SHALL support a member at the JSON Pointer <code>/inputs/obstacles</code> in the route definition in a HTTP POST request to the path <code>/routes</code> based on the following schema (a GeoJSON MultiPolygon, wrapped into a "value" member):</p> <pre data-bbox="665 472 2127 1349">type: object required:   - value properties:   value:     type: object     required:       - type       - coordinates     properties:       type:         type: string         enum:           - MultiPolygon       coordinates:         type: array         items:           type: array           items:             type: array             minItems: 4             items:               type: array               minItems: 2               items:                 type: number</pre>

# Specify Temporal constraints

Requirement 25	<code>/req/time/input</code>
A	<p>The server SHALL support a member at the JSON Pointer <code>/inputs/when</code> in the route definition in a HTTP POST request to the path <code>/routes</code> based on the following schema (an time constraint object, wrapped into a "value" member):</p> <pre data-bbox="682 494 2175 1248">type: object required:   - value properties:   value:     type: object     required:       - timestamp     properties:       timestamp:         type: string         format: date-time     type:       type: string       default: departure       enum:         - departure         - arrival</pre>
B	<p>The <code>timestamp</code> value SHALL be a <code>date-time</code> string value according to <a href="#">RFC 3339, 5.6</a> in UTC (time zone "Z").</p>

# OGC Route Exchange Model

---

The Route model is driven by three routing scenarios.

These were:

- **Online** - fully connected with stability
- **Intermittent** - unreliable connection
- **Offline** - no connectivity



# Online - fully connected with stability

---

- In the Online scenario, an operator uses a routing client to request a route from a Routing API provider, which in turn retrieves the route from an **online routing engine**.
- In this scenario **all components have consistent connections** between them and out to the wider internet.
- This scenario uses OGC API Routes **for all request and response handling** between the client and the routing infrastructure.



# Intermittent - unreliable connection

---

- The intermittent scenario is where the components have **connectivity**, but it is **not necessarily consistent, stable, reliable, or high-speed**.
- Therefore, the **network** cannot be relied upon to provide connectivity on demand and compensation actions are likely when connectivity is not available.
- Intermittent connectivity is **unpredictable** and it maybe that in the real-world, **decisions** are made to treat intermittent connectivity as **no connectivity**, as it is the only sensible course of action, especially if the **scenario involves threat to life**.

# Intermittent - unreliable connection

---

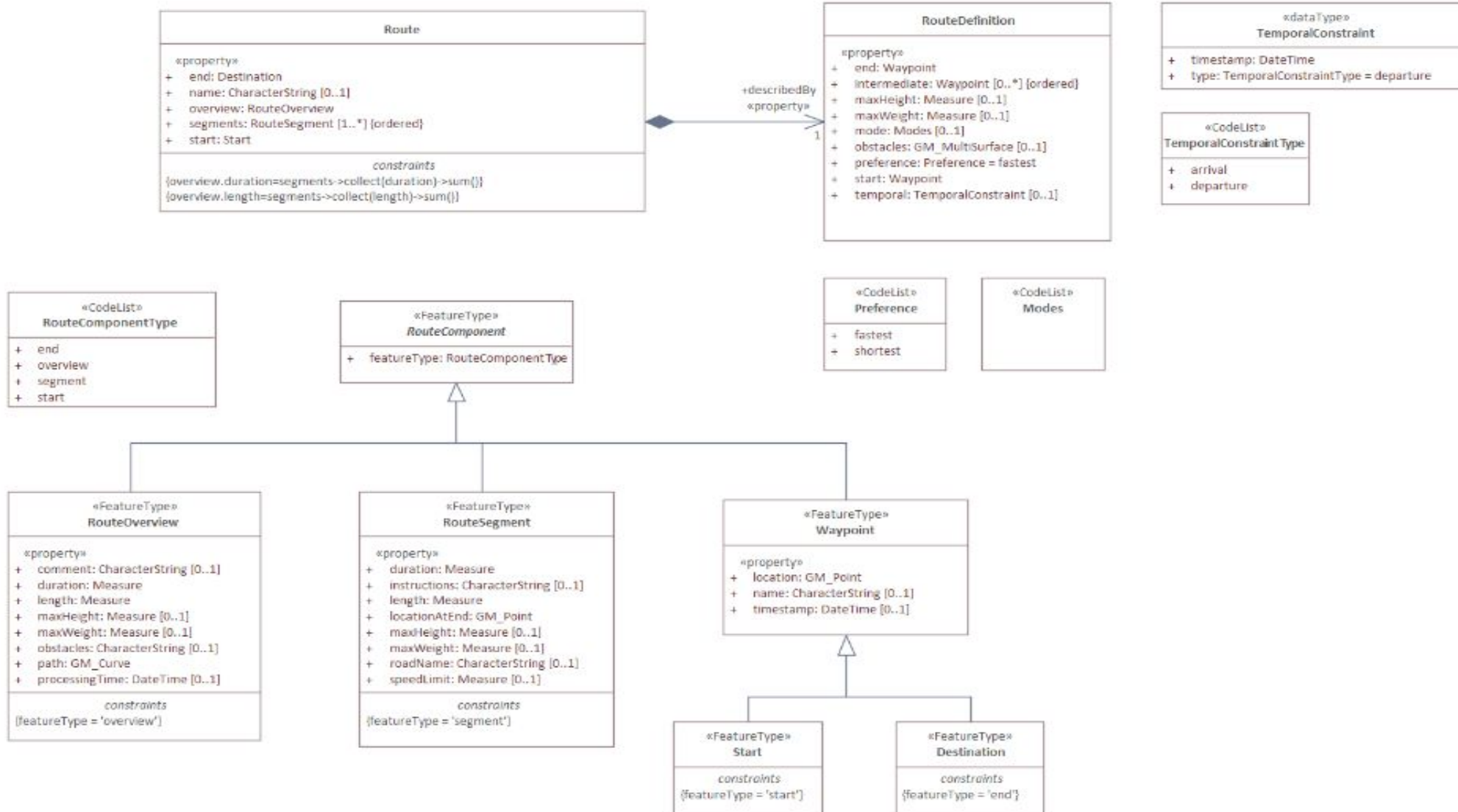
- For example, **only one of the clients had access to a routing engine.** Therefore, the connected client had the ability to create routes, but other clients cannot. If the clients are able to communicate with each other via some other means (**Bluetooth** or some other peer-to-peer communication, for example), the clients could still **share pre-defined routes**, that is, the routing operation has been completed when a **connection to the routing engine was established, but has now been lost.**
- Another approach to support in particular low-bandwidth situations is to **not transmit the complete route definition to the client, but to return route information segment by segment as the vehicle moves along the route.** This approach is not supported by this standard and might be added in a future extension or revision.

# Offline - no connectivity

---

- The Offline scenario assumes that there is **no connectivity** outside of a device's local network, this could be a desktop computer, mobile device or a mesh. In the real-world the scenario is modeling an instance where there is no connectivity and there is not going to be any connectivity for the duration of an operation.
- An operator uses the **routing functionality provided by the client to create a route**. The operator then shares this route with other local clients using the **route exchange model**. To enable the required functionality, all of the capability has to be tightly coupled in a single location. Practically, this **involves installing all of the components on the same machine** to remove communication dependencies with the wider network.

# Route Model



# Route Model

---

- A single encoding of the Route model described in the previous clause for the creation and transfer of route information between a variety of routing components ("**Route Exchange Model**").
- The encoding **uses GeoJSON** to represent routes according to the model described above and is defined in the next clause.
- A GeoJSON feature collection is used to **represent the Route**.
- Each **RouteComponent** is represented by a GeoJSON feature in the feature collection of the route.
- **GeoJSON is used for the following reasons:**
  - **lightweight**
  - **extensible**
  - **widely supported** in libraries and tools
  - **consistent** with the emerging OGC API standards
  - **an open standard**

# Requirement class for Route Model Exchange: Features

---

Requirement 1	<code>/req/rem/geojson</code>
A	The representation of a route SHALL be a valid GeoJSON feature collection.
Requirement 2	<code>/req/rem/features</code>
A	The feature collection SHALL contain the following features, identified by their property <code>featureType</code> : <ul style="list-style-type: none"><li>• a route overview ("overview")</li><li>• the start location of the route ("start")</li><li>• the end location of the route ("end")</li><li>• one or more segments ("segment")</li></ul>
B	The sequence of the segments SHALL be in their order along the route.

# Route Model: Start

---

Requirement 3	<code>/req/rem/start</code>
A	The start location of the route (property <code>featureType</code> is "start") SHALL be a GeoJSON feature with a Point geometry.
B	The point geometry of the feature SHALL be identical to the first point of the route overview.
C	If the feature has a property <code>timestamp</code> , it SHALL be a date-time string value according to <a href="#">RFC 3339, 5.6</a> in UTC (time zone "Z"). The value indicates the estimated departure time.



# Route Model: End

---

Requirement 4	<code>/req/rem/end</code>
A	The start location of the route (property <code>featureType</code> is "end") SHALL be a GeoJSON feature with a Point geometry.
B	The point geometry of the feature SHALL be identical to the last point of the route overview.
C	If the feature has a property <code>timestamp</code> , it SHALL be a <code>date-time</code> string value according to <a href="#">RFC 3339, 5.6</a> in UTC (time zone "Z"). The value indicates the estimated arrival time.



# Route Model: Overview

---

Requirement 5	<i>/req/rem/overview</i>
A	The route overview (property <code>featureType</code> is "overview") SHALL be a GeoJSON feature with a <code>LineString</code> geometry.
B	The line string geometry of the route overview SHALL be the path from the start point to the end point of the route.
C	The route overview SHALL have a property <code>length_m</code> (type: <code>number</code> ) with the length of the segment in meters.
D	The value of a property <code>length_m</code> SHALL be identical to the sum of all route segment properties with the same name.
E	If the route overview has a property <code>duration_s</code> , the value SHALL be of type <code>number</code> with the estimated amount of time required to travel the segment in seconds.

# Hands-on: 1

---

## PMT Project: Routing

# Hands-on: 1

The screenshot shows the QGIS desktop application with the following components:

- Menu Bar:** Project, Edit, View, Layer, Settings, Plugins, Vector, Raster, Database, Web, Mesh, HCGMIS, MMQGIS, Processing, Help.
- Toolbars:** Standard QGIS toolbars and a custom toolbar with icons for various processing tools.
- Browser Panel:** Shows data sources like Mapzen Global Terrain, OpenStreetMap, WCS, WFS / OGC API - Features, ArcGIS REST Servers, and GeoNode.
- Layers Panel:** Lists several layers including 'Zone\_Population\_2018', 'Mode\_OUT', 'Mode\_IN', and 'PuneCMP\_BaseYear\_PTNetwork'. The 'Lines to points' layer is currently selected and visible on the map.
- Map Canvas:** Displays a map of Pune, India, with a network of roads highlighted in yellow and orange. Red dots represent the extracted vertices from the road network.
- Toolbox:** The 'Vector' menu is open, and the 'Geometry Tools' sub-menu is selected. The 'Extract Vertices...' option is highlighted.
- Status Bar:** Shows the current coordinate (18.5130, 73.7083), scale (1:41927), magnifier (100%), rotation (0.0°), and projection (EPSG:4326).
- System Tray:** Displays system information such as temperature (30°C), weather (Haze), and the date/time (17-10-2022, 11:29).



# Hands-on:

**Extract Vertices**

Parameters Log

Input layer  
PuneCMP\_BaseYear\_PTNetwork [EPSI]

Selected features only

Vertices  
MT) Project/To Client/PTNetwork\_Lines\_to\_Points.shp

Open output file after running algorithm

0%

Advanced Run as Batch Process... Run Close Help

**Extract vertices**

This algorithm takes a line or polygon layer and generates a point layer with points representing the vertices in the input lines or polygons. The attributes associated to each point are the same ones associated to the line or polygon that the point belongs to.

Additional fields are added to the point indicating the vertex index (beginning at 0), the vertex's part and its index within the part (as well as its ring for polygons), distance along original geometry and bisector angle of vertex for original geometry.

Map layers:  
Zone\_Population\_2018  
Mode\_OUT  
Mode\_IN  
PT\_OUT\_div\_TOTAL\_OUT  
PT\_IN\_div\_TOTAL\_IN  
PT\_IN\_vs\_PT\_OUT  
PT\_IN  
PT\_OUT  
MODE\_TOTAL\_OUT\_per1000  
MODE\_TOTAL\_IN\_per1000  
MODE\_TOTAL\_IN  
MODE\_TOTAL\_OUT  
Population\_perZone  
buses\_per1000  
buses\_per\_zone  
Pune\_roads\_clip  
PuneCMP\_BaseYear\_PTNetwork  
PuneCMP\_TAZ  
PTNetwork\_Lines\_to\_points  
OSM Standard

Coordinate 18.4565,73.8026 Scale 1:41927 Magnifier 100% Rotation 0.0° Render EPSG:4326

30°C Haze 11:30 17-10-2022

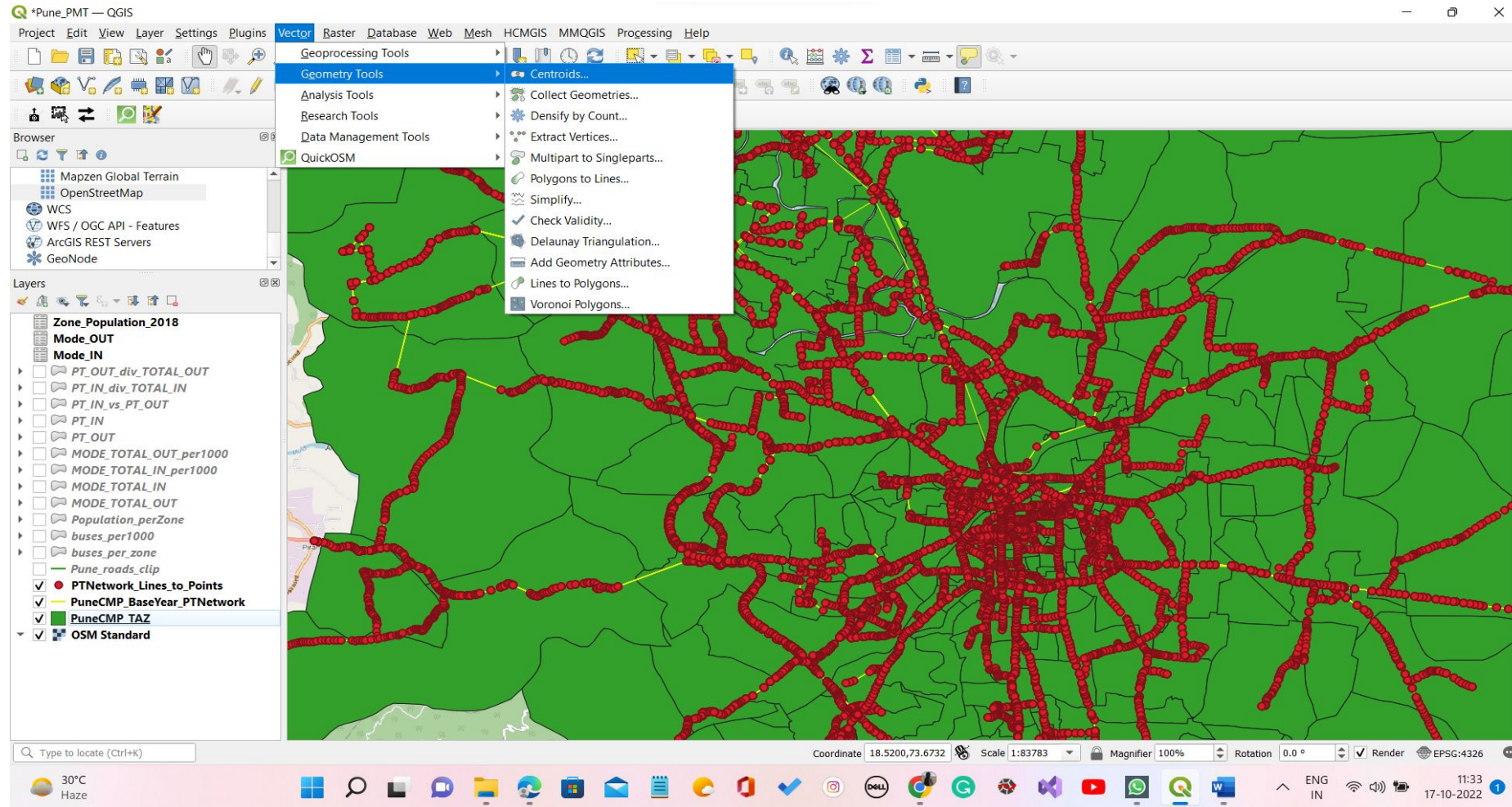


# Hands-on:

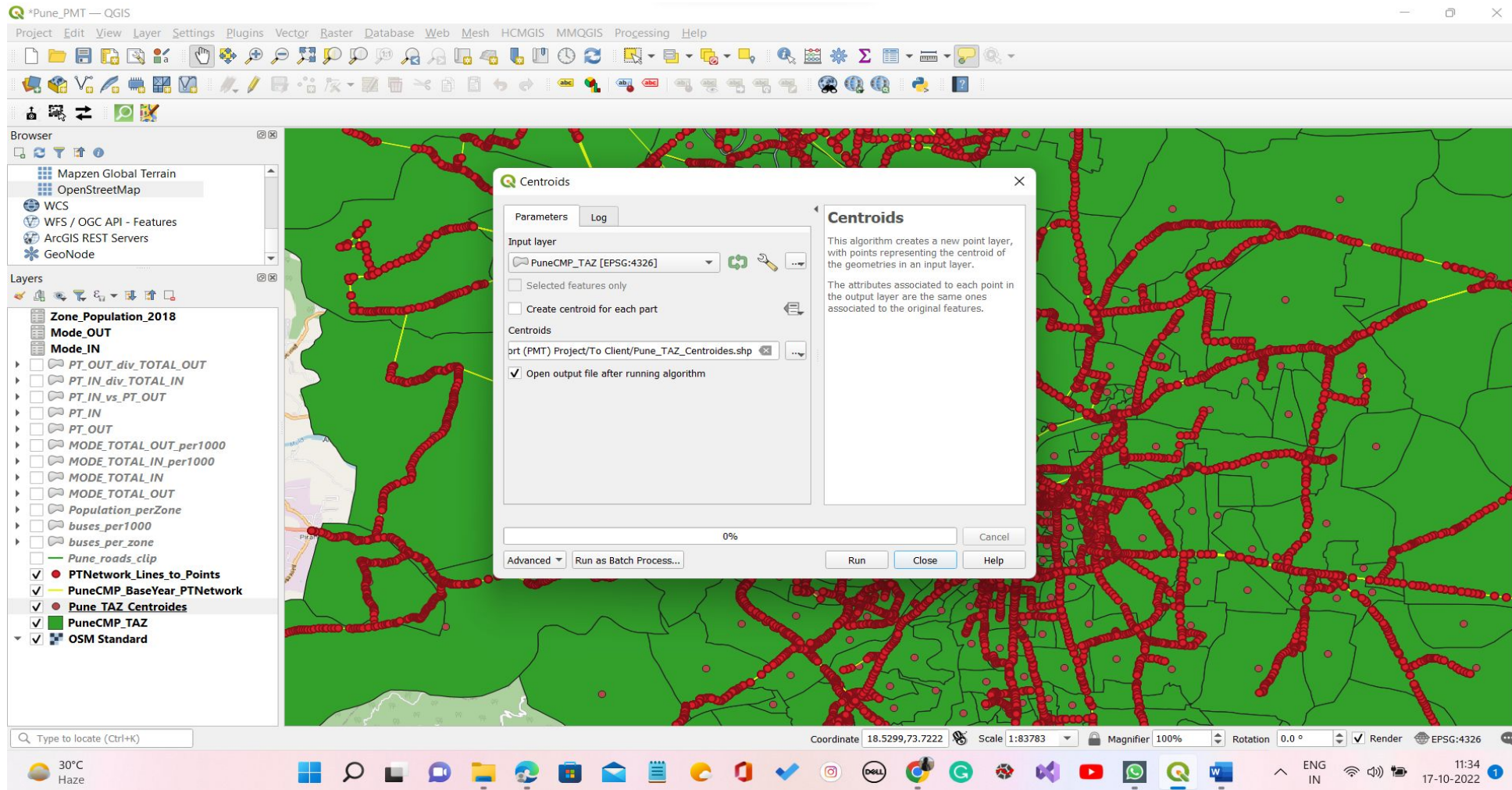
The screenshot displays the QGIS desktop application window titled '\*Pune\_PMT — QGIS'. The interface includes a menu bar (Project, Edit, View, Layer, Settings, Plugins, Vector, Raster, Database, Web, Mesh, HCMGIS, MMQGIS, Processing, Help) and a toolbar with various GIS tools. On the left, the 'Browser' panel shows data sources like 'Mapzen Global Terrain', 'OpenStreetMap', 'WCS', 'WFS / OGC API - Features', 'ArcGIS REST Servers', and 'GeoNode'. The 'Layers' panel lists several layers, with 'PTNetwork Lines to Points' and 'OSM Standard' checked. The main map area shows a detailed view of Pune, India, with a dense network of red lines representing the PT network overlaid on a satellite-style map. The status bar at the bottom indicates the current coordinate (18.5646, 73.8513), scale (1:83783), magnification (100%), rotation (0.0°), and projection (EPSG:4326). The Windows taskbar at the very bottom shows the system tray with a temperature of 30°C, date (17-10-2022), and time (11:32).



# Hands-on:



# Hands-on:





# Hands-on:

The screenshot displays the QGIS desktop application window titled "Pune\_PMT — QGIS". The interface includes a menu bar (Project, Edit, View, Layer, Settings, Plugins, Vector, Raster, Database, Web, Mesh, HCMGIS, MMQGIS, Processing, Help), a toolbar with various GIS tools, and a main map area. On the left, there are two panels: "Browser" and "Layers".

The "Browser" panel shows the following layers:

- Mapzen Global Terrain
- OpenStreetMap
- WCS
- WFS / OGC API - Features
- ArcGIS REST Servers
- GeoNode

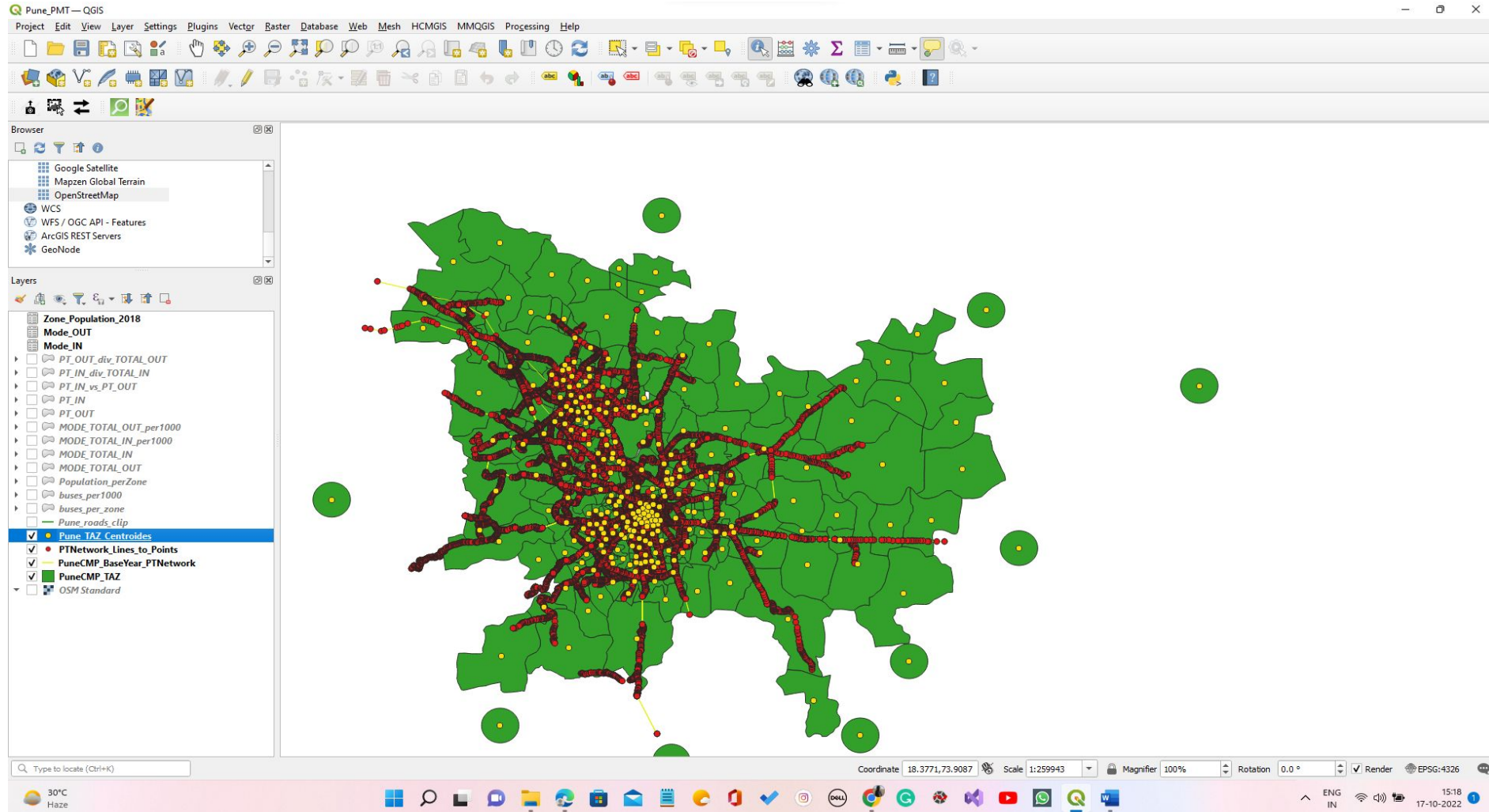
The "Layers" panel shows a list of layers with checkboxes:

- Zone\_Population\_2018
- Mode\_OUT
- Mode\_IN
- PT\_OUT\_div\_TOTAL\_OUT
- PT\_IN\_div\_TOTAL\_IN
- PT\_IN\_vs\_PT\_OUT
- PT\_IN
- PT\_OUT
- MODE\_TOTAL\_OUT\_per1000
- MODE\_TOTAL\_IN\_per1000
- MODE\_TOTAL\_IN
- MODE\_TOTAL\_OUT
- Population\_perZone
- buses\_per1000
- buses\_per\_zone
- Pune\_roads\_clip
- PTNetwork\_Lines\_to\_Points
- PuneCMP\_BaseYear\_PTNetwork
- Pune\_TAZ\_Centroides
- PuneCMP\_TAZ
- OSM Standard

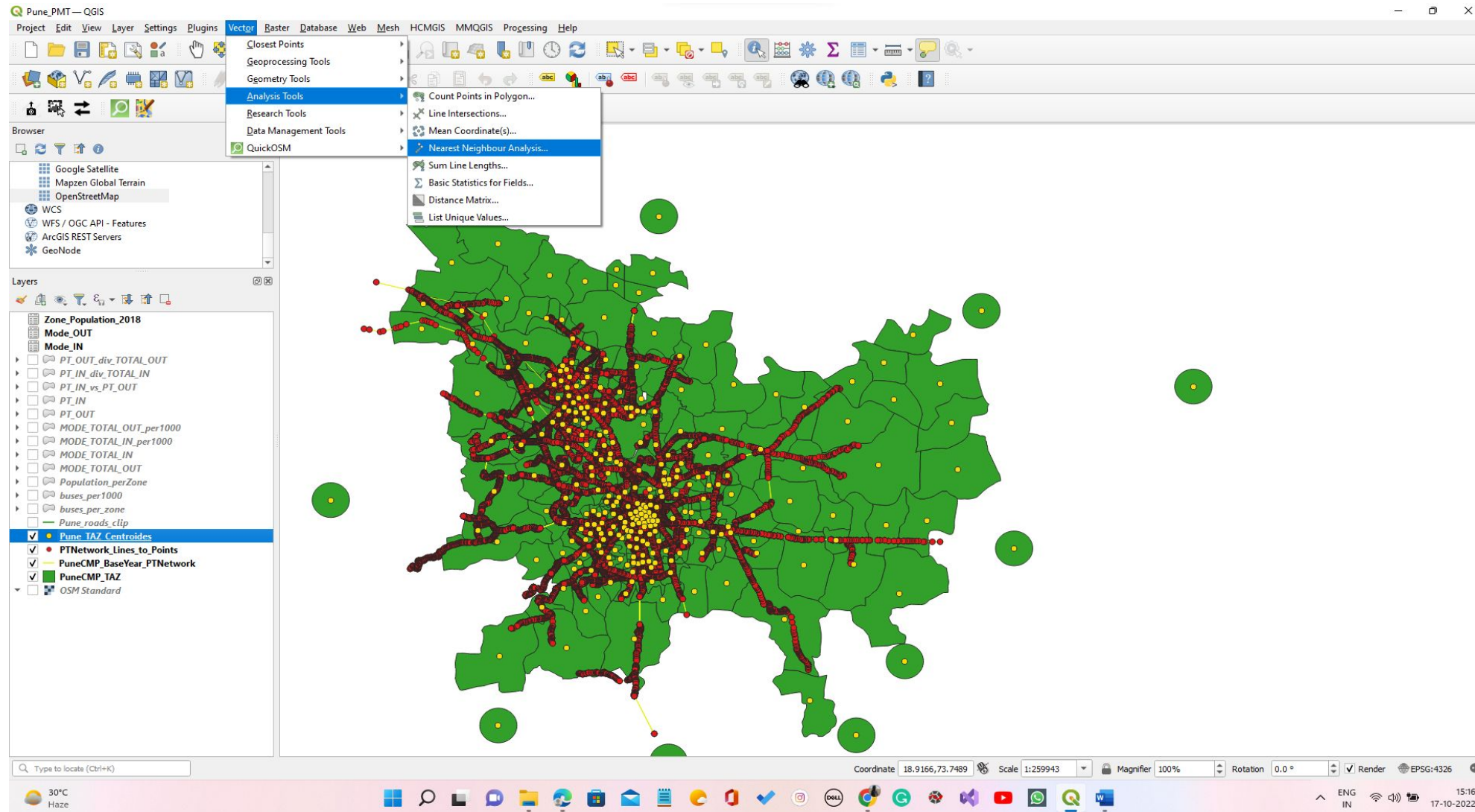
The main map area shows a green-colored map of Pune with a network of yellow lines and orange dots. A search bar at the bottom left contains the text "Type to locate (Ctrl+K)". The status bar at the bottom displays the saved project path: "C:\Prajwalita\IITB\QGIS\Pune Municipal Transport (PMT) Project\Pune\_PMT.qgz", the coordinate "18.5719,73.8614", the scale "1:167516", the magnifier "100%", the rotation "0.0 °", the render status "Render", and the EPSG code "EPSG:4326". The Windows taskbar at the bottom shows the system tray with the date "17-10-2022" and time "11:35".



# Hands-on:



# Hands-on:



# Hands-on:

The screenshot shows the QGIS interface with the 'Closest Points' plugin window open. The map displays a network of roads (red lines) and points (yellow dots) overlaid on a green terrain. The 'Closest Points' plugin window is open, showing its description and settings.

**Closest Points**

get the closest points for one layer to another

This plugin extracts the closest points for all features of a given layer to another layer using the euclidean distance.

☆☆☆☆☆ 2 rating vote(s), 8791 downloads

**Category** Vector

**Tags** vector, point, points, closest, nearest

**More info** [homepage](#) [bug tracker](#) [code repository](#)

**Author** [Christoph Jung](#)

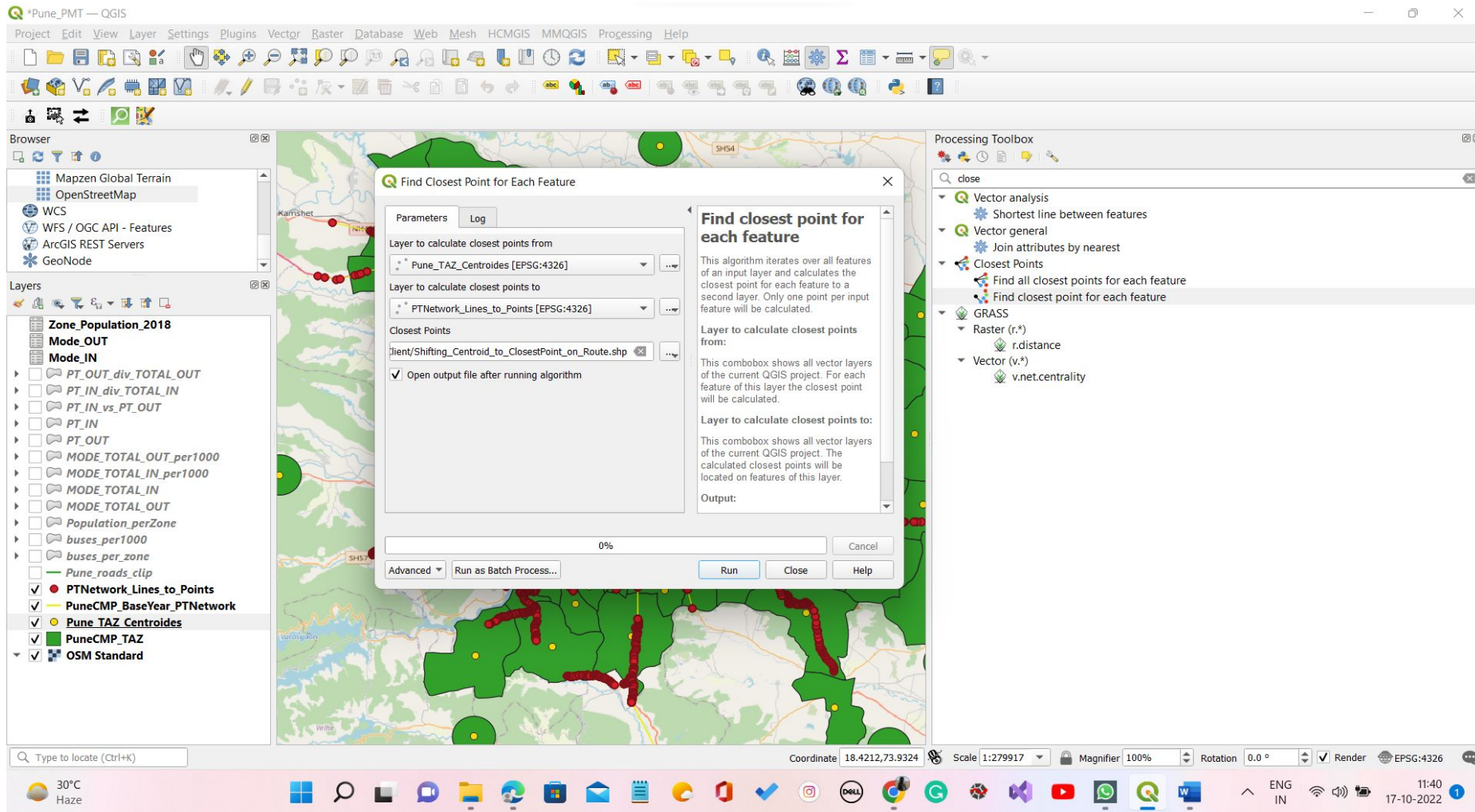
**Installed version** 1.0.1

**Available version (stable)** 1.0.1 updated at Tue Jul 14 13:41:06 2020

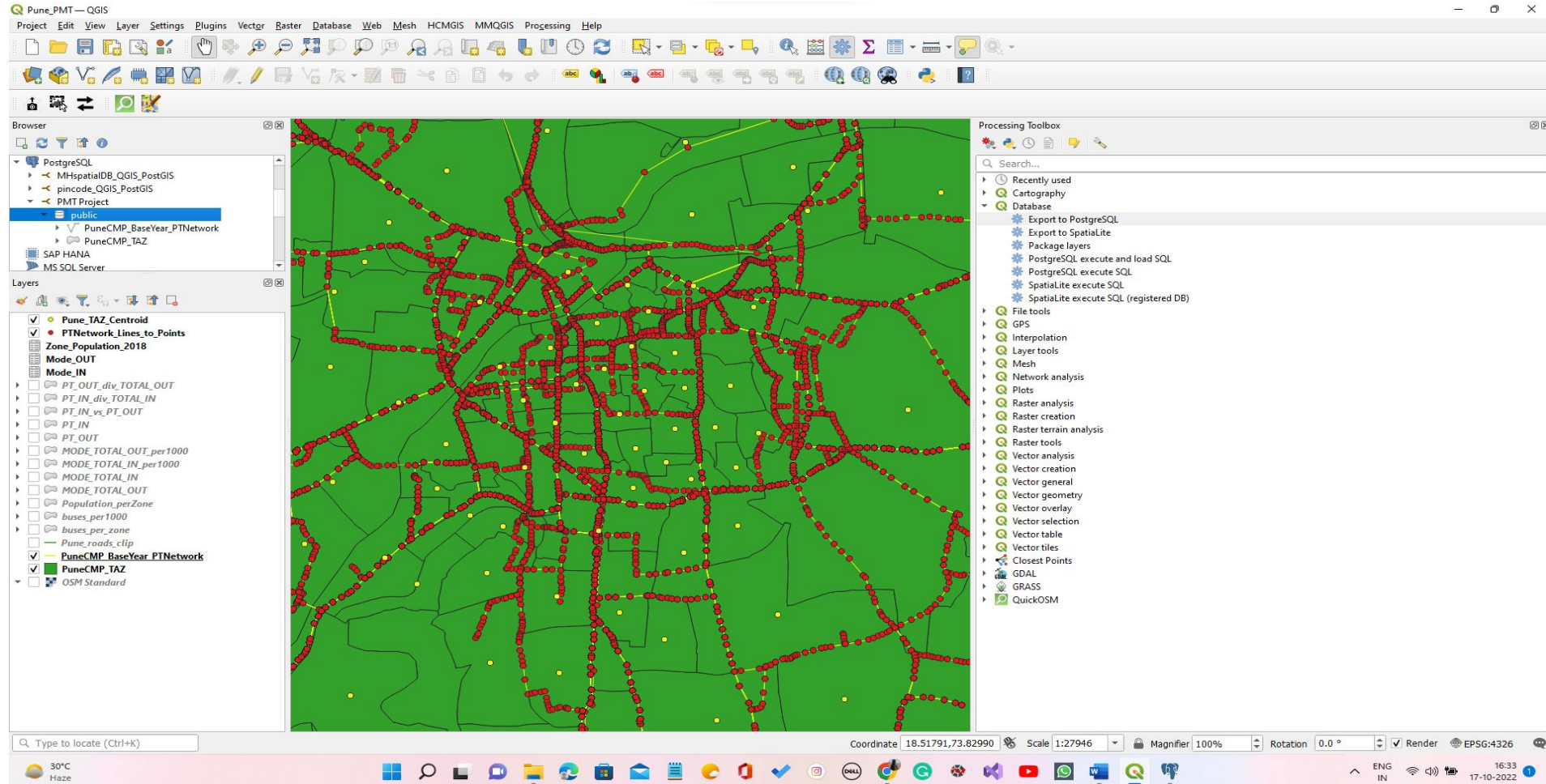
Upgrade All Uninstall Plugin Reinstall Plugin Close Help



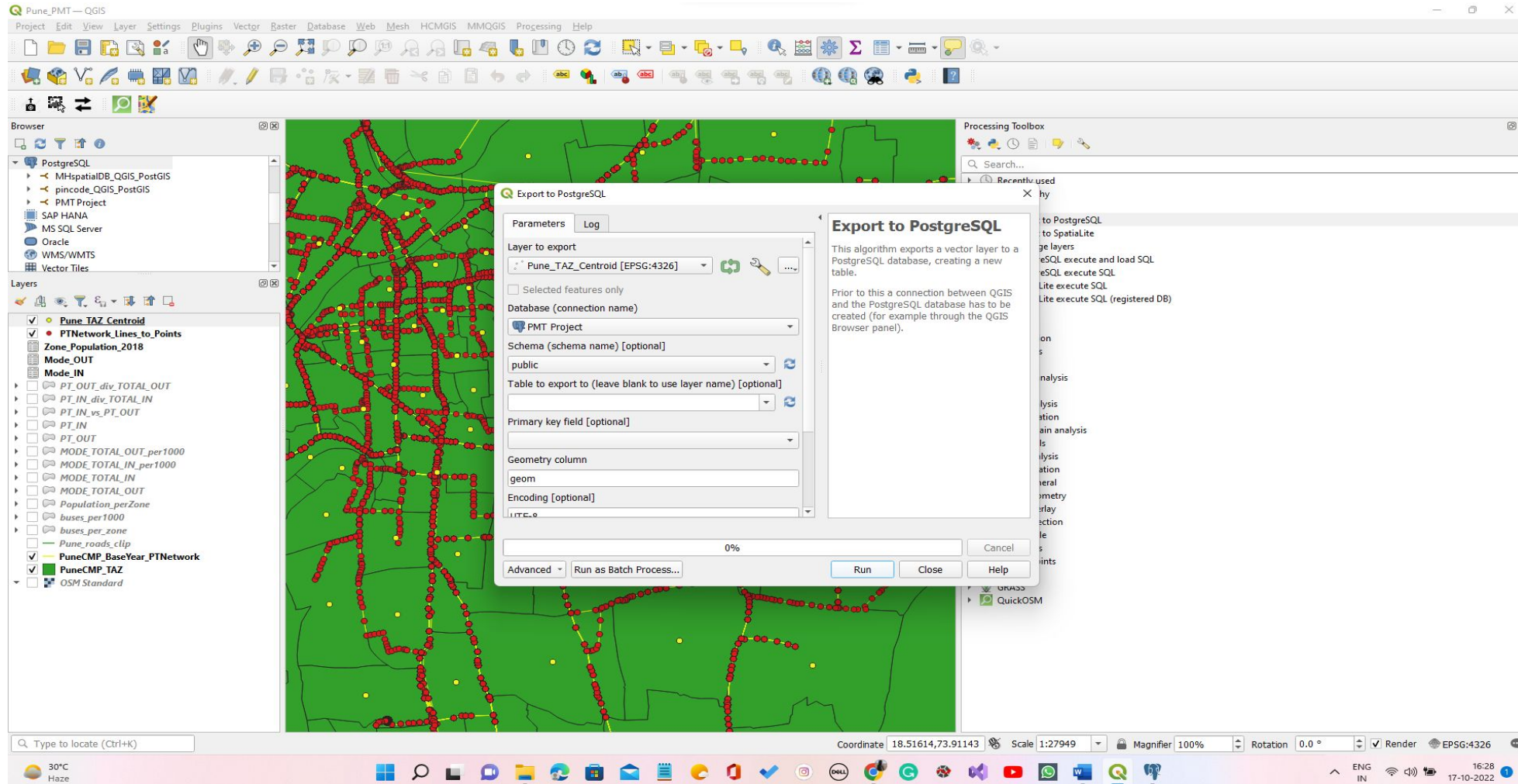
# Hands-on:



# Hands-on:



# Hands-on:





# Hands-on:

The screenshot displays the QGIS interface with the 'Shortest Path (Point to Layer)' dialog box open. The dialog box contains the following parameters:

- Parameters** tab is active.
- Vector layer representing network:** PuneCMP\_BaseYear\_PTNetwork [E]
- Path type to calculate:** Shortest
- Start point:** (Empty text box)
- Vector layer with end points:** Pune\_TAZ\_Centroid [EPSG:4326]
- Advanced Parameters:**
  - Shortest path:** [Create temporary layer]

The Processing Toolbox on the right shows the 'Shortest path (point to layer)' tool selected under the 'Network analysis' category. The Identify Results panel on the far right shows the following data:

Feature	Value
<b>Shortest path</b>	
Details	Village
(Derived)	
(Actions)	
SHAPE_Le...	0.1647164897
SHAPE_Ar...	0.0013232109
Details	Village
Areas	Sus (N.V.), Mai
Taluk	Mulshi
TAZWARDS	233
start	73.8704, 18.44
end	73.750001097
cost	0.1959137430



# Hands-on:

The screenshot displays the QGIS interface for a project named 'Pune\_PMT'. The main map area shows a green-shaded region representing the Pune Metropolitan Region (PMR), overlaid with a network of roads and transit lines. A red line highlights a specific shortest path between two zones. The interface includes a top menu bar, a toolbar, and several panels: a Browser panel on the left showing project structure, a Layers panel listing various data layers, and an Identify Results panel on the right showing details for a selected feature.

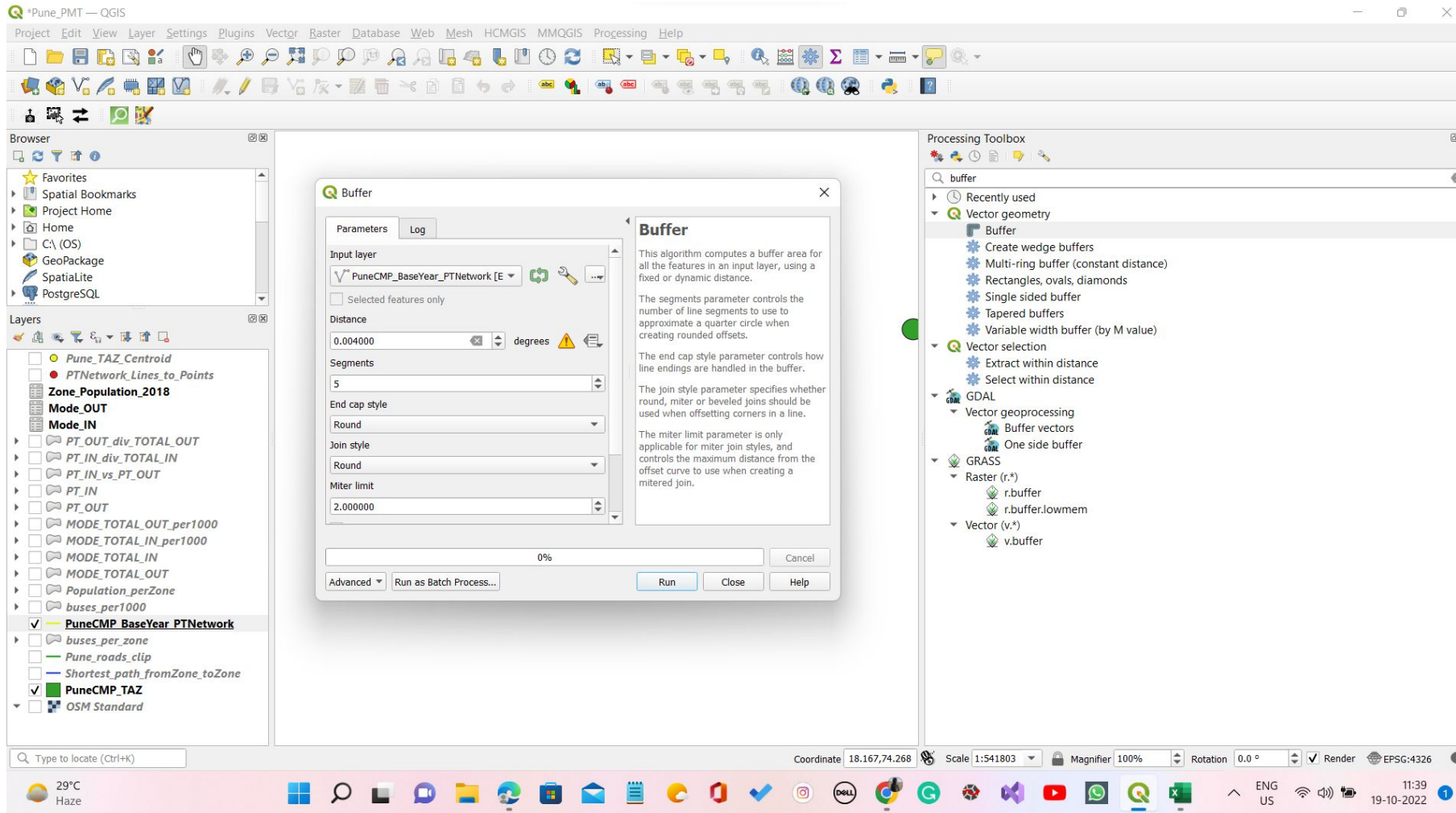
**Layers Panel:**

- ✓ Pune\_TAZ\_Centroid
- PTNetwork\_Lines\_to\_Points
- Zone\_Population\_2018
- Mode\_OUT
- Mode\_IN
  - PT\_OUT\_div\_TOTAL\_OUT
  - PT\_IN\_div\_TOTAL\_IN
  - PT\_IN\_vs\_PT\_OUT
  - PT\_IN
  - PT\_OUT
  - MODE\_TOTAL\_OUT\_per1000
  - MODE\_TOTAL\_IN\_per1000
  - MODE\_TOTAL\_IN
  - MODE\_TOTAL\_OUT
  - Population\_perZone
  - buses\_per1000
- PuneCMP\_BaseYear\_PTNetwork
- buses\_per\_zone
- Pune\_roads\_clip
- ✓ Shortest\_path\_fromZone\_toZone
- ✓ PuneCMP\_TAZ
- ✓ OSM Standard

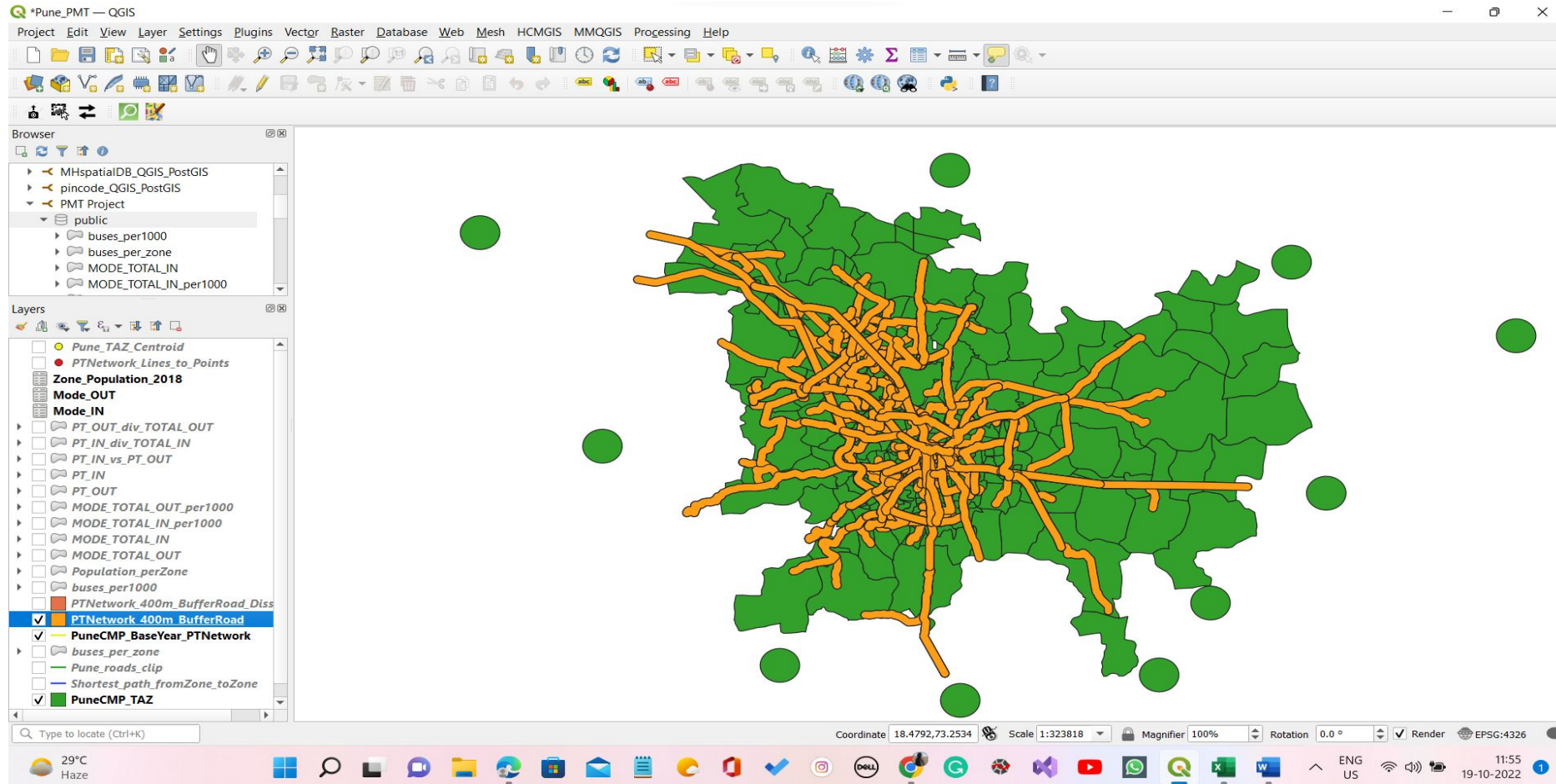
**Identify Results Panel:**

Feature	Value
Shortest_path_fromZone_toZone	Village
(Derived)	
(Actions)	
SHAPE_Le...	0.31231396533
SHAPE_Ar...	0.00446435082
Details	Village
Areas	Ranjane, Khamgaon, Kondgaon, Ambed, Wardad...
Taluk	Haveli
TAZWARDS	236
start	73.8704, 18.443767
end	73.7144980202, 18.3678128488
cost	0.198840251629662

# Hands-on:



# Hands-on:





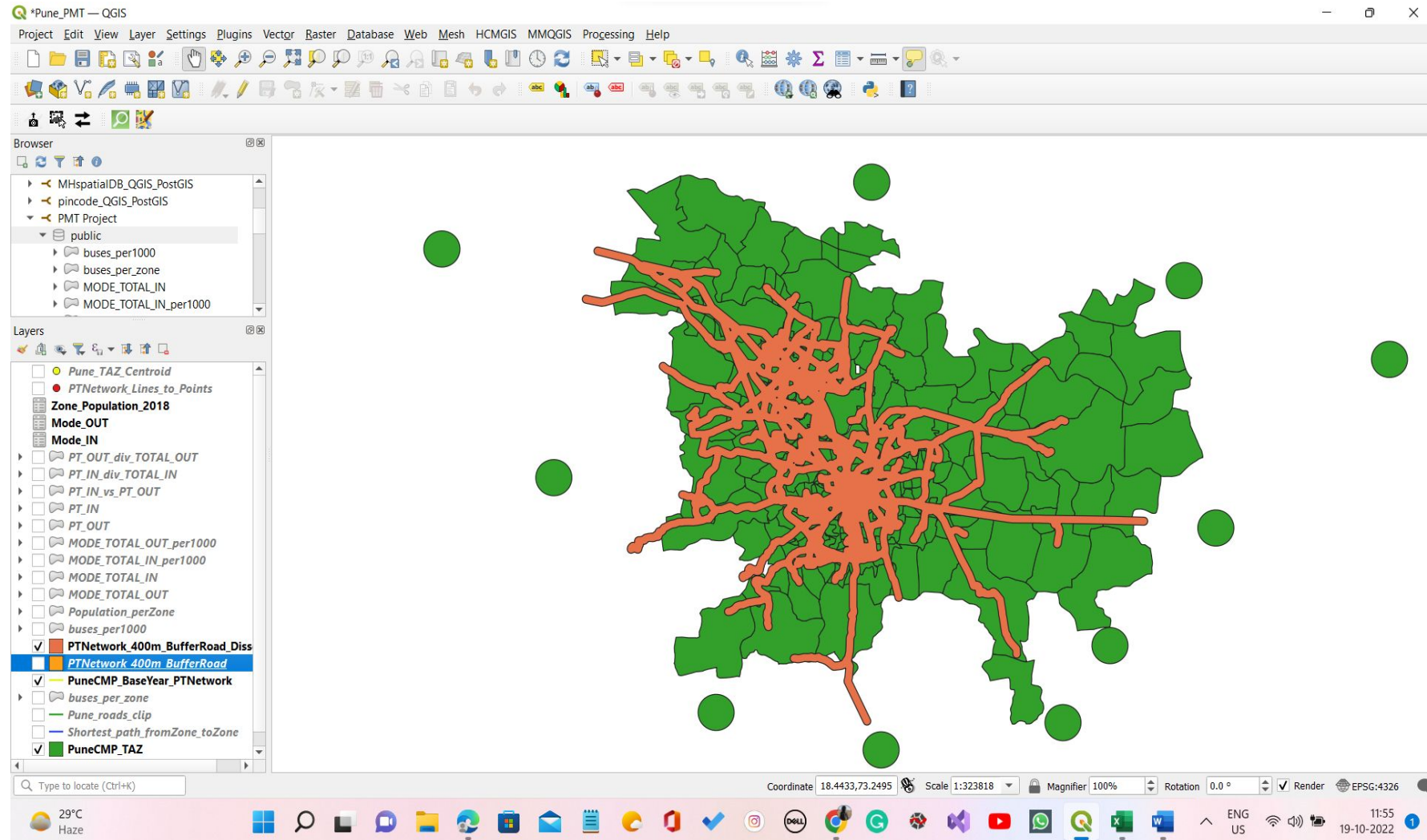
# Hands-on:

The screenshot displays the QGIS interface with the Dissolve tool dialog box open. The dialog box is titled "Dissolve" and has two tabs: "Parameters" and "Log". The "Parameters" tab is active, showing the following settings:

- Input layer:** PTNetwork\_400m\_BufferRoad [EPSG:...]
- Selected features only:**
- Dissolve field(s) [optional]:** 0 field(s) selected
- Advanced Parameters:**
  - Dissolved:** To Client/PTNetwork
  - Open output file:**  Python identifier: OUTPUT

The "Processing Toolbox" on the right side of the interface shows the "Dissolve" tool selected under the "Vector geometry" category. The "Layers" panel on the left side shows a list of layers, with "PTNetwork\_400m\_BufferRoad" and "PuneCMP\_TAZ" checked. The status bar at the bottom of the window displays the following information: Coordinate: 18.724,74.310, Scale: 1:541803, Magnifier: 100%, Rotation: 0.0 °, Render, and EPSG:4326.

# Hands-on:

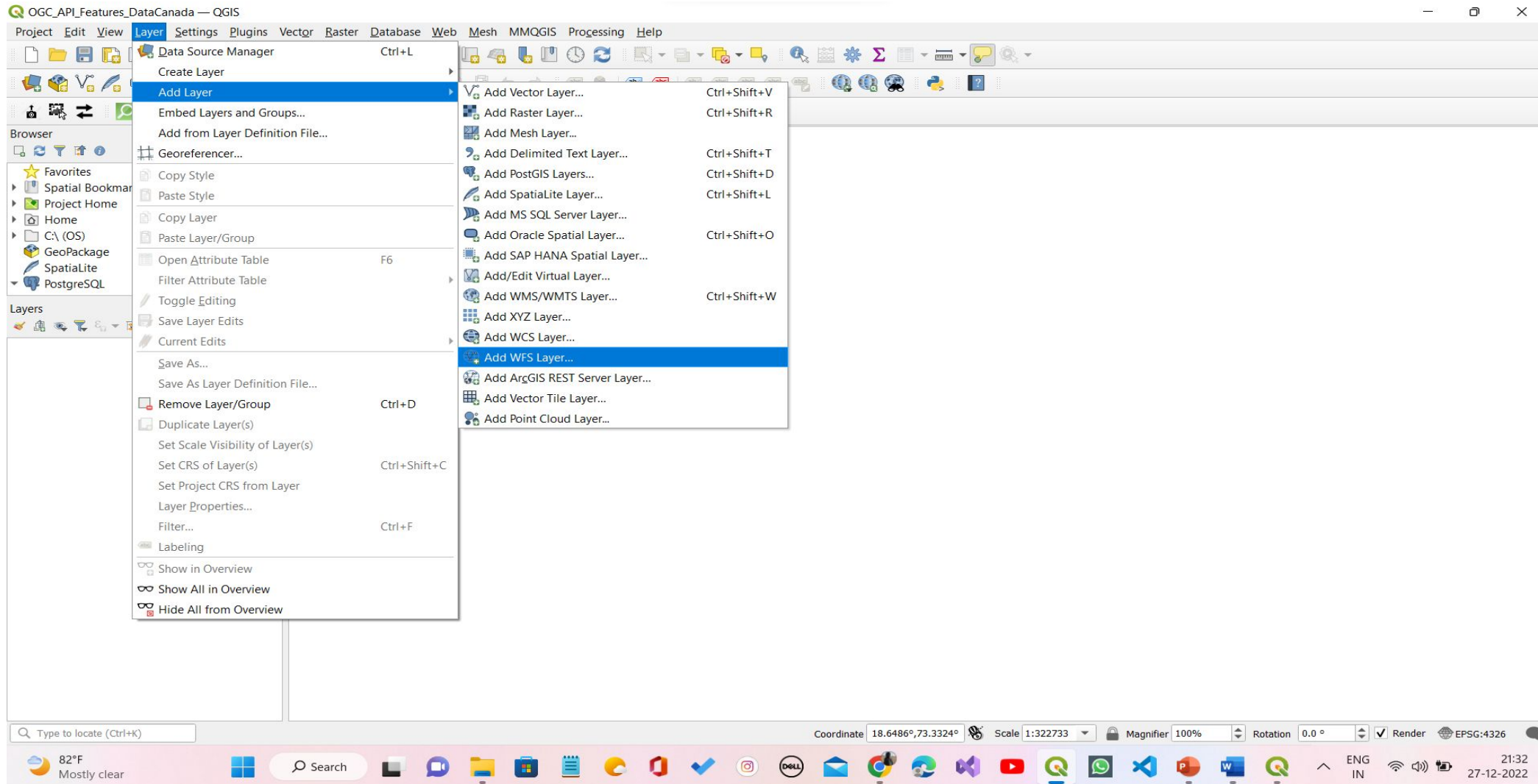


# Hands-on: 2

---

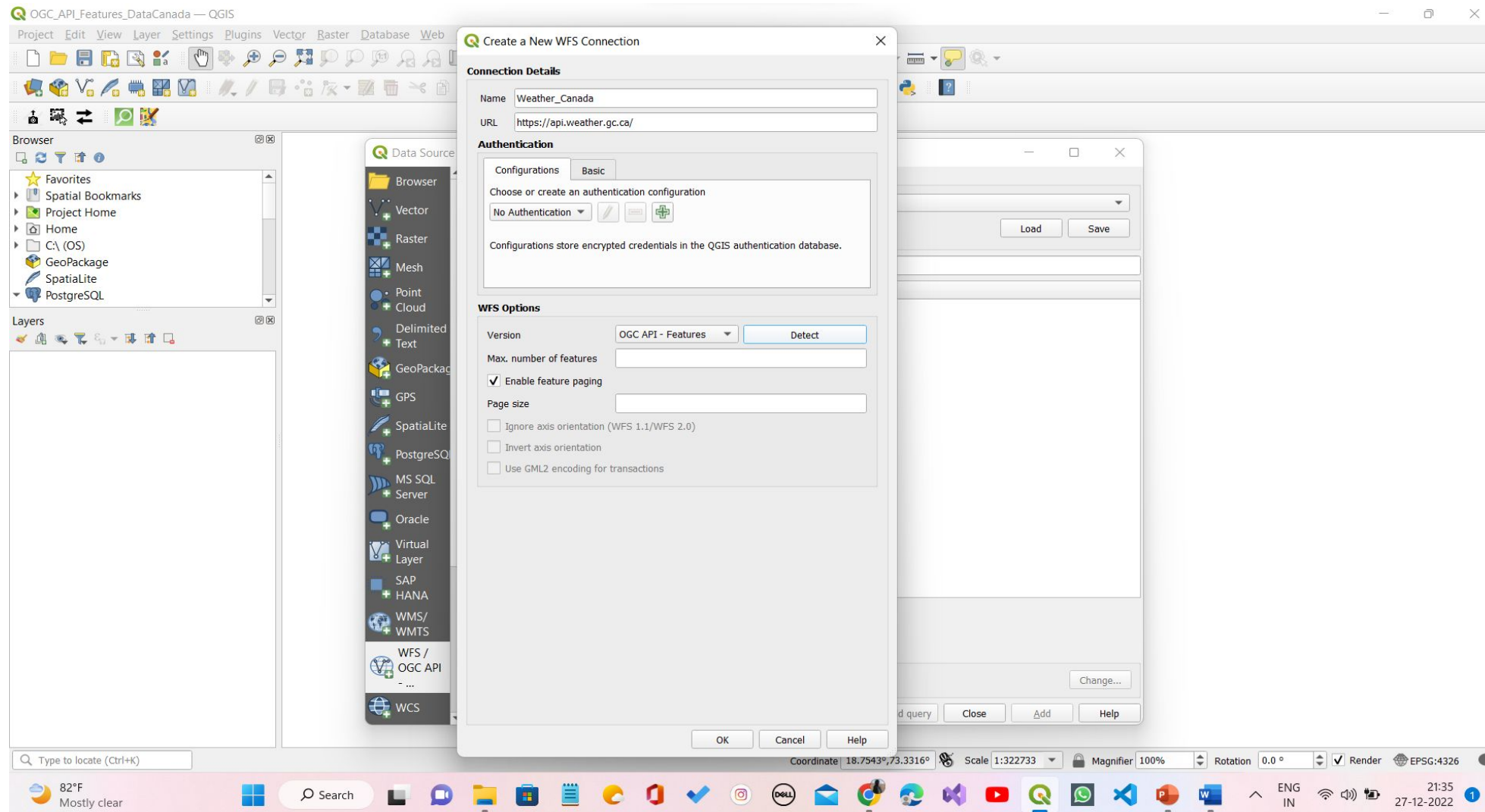
## Canada Weather Data- OGC API

# Hands-on: 2

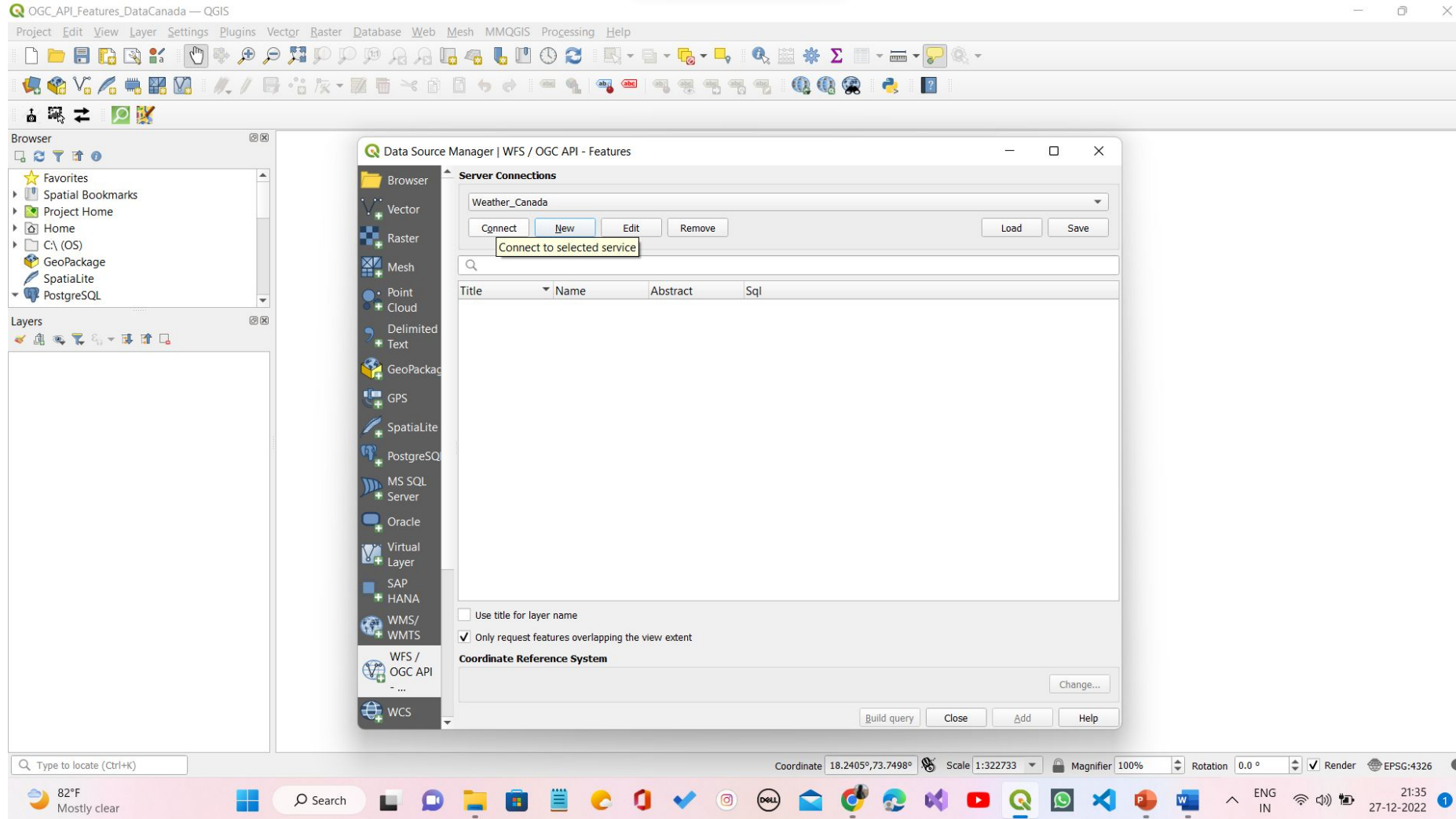




# Hands-on: 2



# Hands-on: 2



# Hands-on: 2

OGC\_API\_Features\_DataCanada — QGIS

Project Edit View Layer Settings Plugins Vector Raster Database Web Mesh MMQGIS Processing Help

Browser

- Favorites
- Spatial Bookmarks
- Project Home
- Home
- C:\ (OS)
- GeoPackage
- SpatialLite
- PostgreSQL

Layers

Data Source Manager | WFS / OGC API - Features

Server Connections

Weather\_Canada

Connect New Edit Remove Load Save

Title	Name	Abstract	Sql
WMO WIS2 notification messages (exp...	wis2-notification-messages	WMO WIS2 notifica...	
WMO WIS2 discovery metadata (exper...	wis2-discovery-metadata	WMO WIS2 discove...	
Virtual Climate Stations (LTCE)	ltce-stations	A Virtual Climate st...	
Surface Weather Observations	swob-realtime	Surface Observatio...	
Regional Deterministic Precipitation A...	weatherrdpa:15km:6f	The Regional Deter...	
Regional Deterministic Precipitation A...	weatherrdpa:10km:6p	The Regional Deter...	
Regional Deterministic Precipitation A...	weatherrdpa:10km:6f	The Regional Deter...	
Regional Deterministic Precipitation A...	weatherrdpa:15km:24f	The Regional Deter...	
Regional Deterministic Precipitation A...	weatherrdpa:10km:24p	The Regional Deter...	
Regional Deterministic Precipitation A...	weatherrdpa:10km:24f	The Regional Deter...	
Real-time meteorological bulletins	bulletins-realtime	Real-time meteorol...	
Real-time hydrometric data	hydrometric-realtime	Real-time water lev...	
Projected SPEI-3	climate:spei-3:projected	The Standardized P...	
Projected SPEI-12	climate:spei-12:projected	The Standardized P...	
Projected SPEI-1	climate:spei-1:projected	The Standardized P...	
Projected seasonal DCS	climate:dcs:projected:seasonal:absolute	The statistically do...	
Projected seasonal CMIP5	climate:cmip5:projected:seasonal:absolute	The Global climate ...	

Use title for layer name

Only request features overlapping the view extent

Coordinate Reference System

Change...

Build query Close Add Help

Coordinate 18.2367°, 73.986° Scale 1:32273 Magnifier 100% Rotation 0.0 ° Render EPSG:4326

82°F Mostly clear Search 27-12-2022 21:35

# Hands-on: 2

OGC\_API\_Features\_DataCanada — QGIS

Project Edit View Layer Settings Plugins Vector Raster Database Web Mesh MMQGIS Processing Help

Browser

- Favorites
- Spatial Bookmarks
- Project Home
- Home
- C:\ (OS)
- GeoPackage
- SpatialLite
- PostgreSQL

Layers

Data Source Manager | WFS / OGC API - Features

Server Connections

Weather\_Canada

Connect New Edit Remove Load Save

Title	Name	Abstract	Sql
Historical indices	climate:indices:historical	High-resolution stat...	
Historical annual DCS	climate:dcs:historical:annual:absolute	The statistically do...	
Historical annual CMIP5	climate:cmip5:historical:annual:absolute	The Global climate ...	
Historical annual anomaly DCS	climate:dcs:historical:annual:anomaly	The statistically do...	
Historical annual anomaly CMIP5	climate:cmip5:historical:annual:anomaly	The Global climate ...	
Daily Mean of Water Level or Flow	hydrometric-daily-mean	The daily mean is th...	
Daily Extremes of Records (LTCE) – Te...	ltce-temperature	Anomalous weather...	
Daily Extremes of Records (LTCE) – Sn...	ltce-snowfall	Anomalous weather...	
Daily Extremes of Records (LTCE) – Pre...	ltce-precipitation	Anomalous weather...	
Daily Climate Observations	climate-daily	Daily climate obser...	
Climate Stations	climate-stations	Climate observatio...	
CanGRD historical seasonal trend	climate:cangrd:historical:seasonal:trend	CANGRD data are i...	
CanGRD historical seasonal anomaly	climate:cangrd:historical:seasonal:anom...	Gridded seasonal m...	
CanGRD historical monthly anomaly	climate:cangrd:historical:monthly:anomaly	Gridded monthly m...	
CanGRD historical annual trend	climate:cangrd:historical:annual:trend	CANGRD data are i...	
CanGRD historical annual anomaly	climate:cangrd:historical:annual:anomaly	Gridded annual me...	
Canadian Seasonal to Inter-annual Pre...	weather:cansips:250km:forecast:members	The Canadian Seas...	

Use title for layer name

Only request features overlapping the view extent

Coordinate Reference System

Change...

Build query Close Add Help

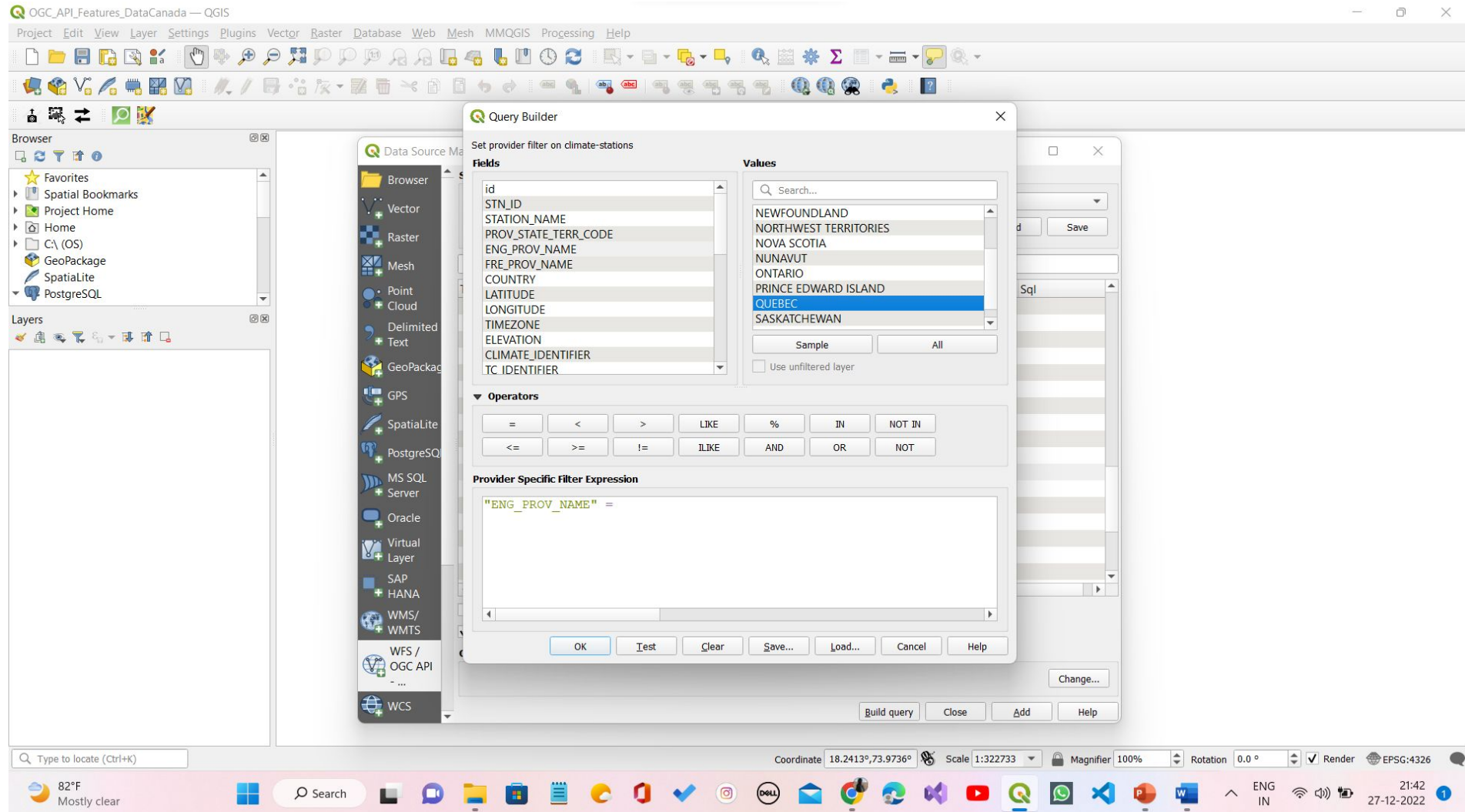
Build query

Coordinate 18.2413°,73.9845° Scale 1:322733 Magnifier 100% Rotation 0.0° Render EPSG:4326

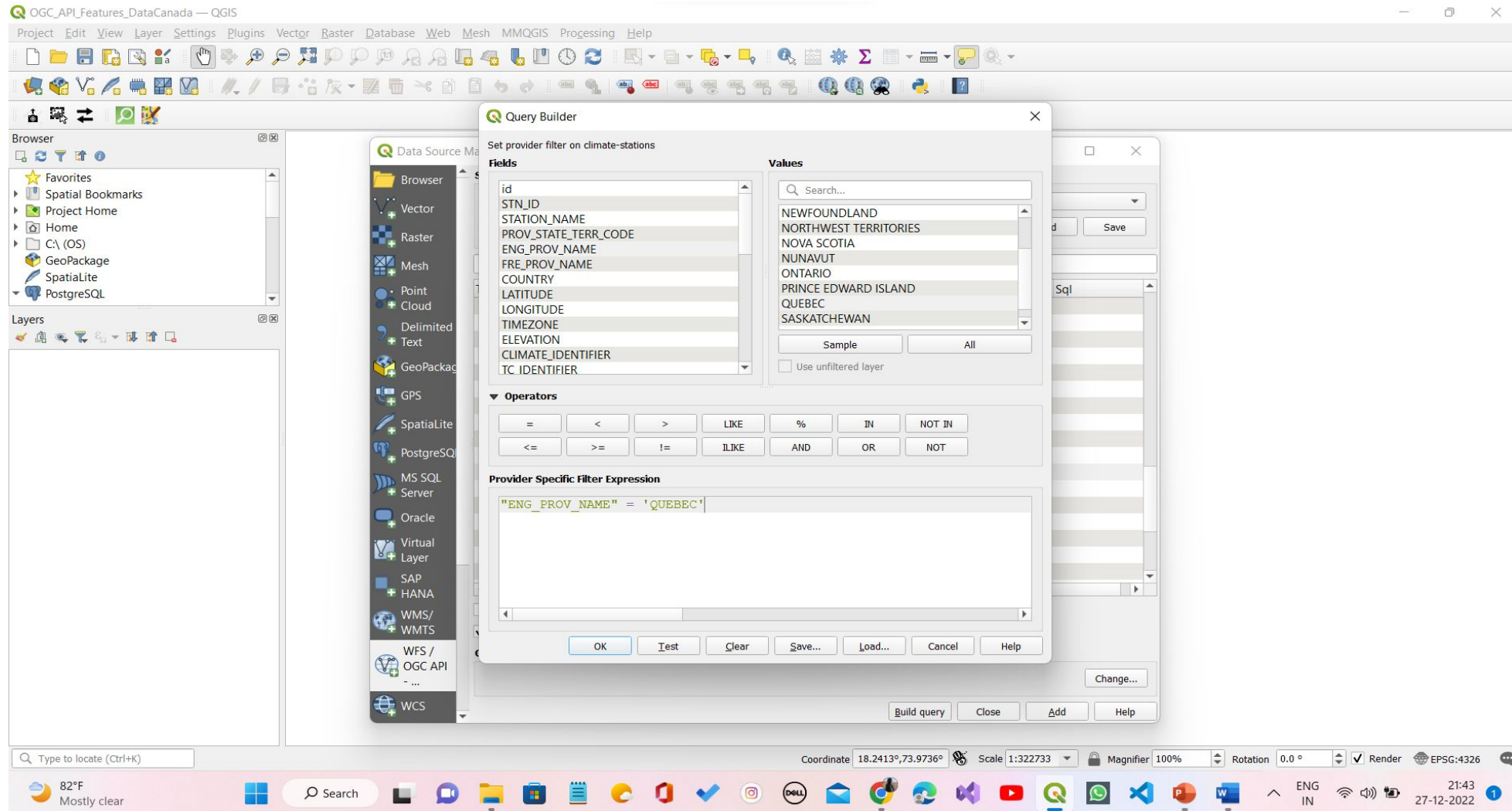
82°F Mostly clear Search 21:37 27-12-2022



# Hands-on: 2



# Hands-on: 2



# Hands-on: 2

The screenshot shows the QGIS Desktop interface with the Data Source Manager dialog box open. The dialog is titled "Data Source Manager | WFS / OGC API - Features" and shows a list of server connections for "Weather\_Canada".

**Server Connections**

Weather\_Canada

Connect New Edit Remove Load Save

Title	Name	Abstract	Sql
Historical indices	climate:indices:historical	High-resolution stat...	
Historical annual DCS	climate:dcs:historical:annual:absolute	The statistically do...	
Historical annual CMIP5	climate:cmip5:historical:annual:absolute	The Global climate ...	
Historical annual anomaly DCS	climate:dcs:historical:annual:anomaly	The statistically do...	
Historical annual anomaly CMIP5	climate:cmip5:historical:annual:anomaly	The Global climate ...	
Daily Mean of Water Level or Flow	hydrometric-daily-mean	The daily mean is th...	
Daily Extremes of Records (LTCE) - Te...	ltce-temperature	Anomalous weather...	
Daily Extremes of Records (LTCE) - Sn...	ltce-snowfall	Anomalous weather...	
Daily Extremes of Records (LTCE) - Pre...	ltce-precipitation	Anomalous weather...	
Daily Climate Observations	climate-daily	Daily climate obser...	
Climate Stations	climate-stations	Climate observatio...	"ENG_PROV_NA...
CanGRD historical seasonal trend	climate:cangrd:historical:seasonal:trend	CANGRD data are i...	
CanGRD historical seasonal anomaly	climate:cangrd:historical:seasonal:anom...	Gridded seasonal m...	
CanGRD historical monthly anomaly	climate:cangrd:historical:monthly:anomaly	Gridded monthly m...	
CanGRD historical annual trend	climate:cangrd:historical:annual:trend	CANGRD data are i...	
CanGRD historical annual anomaly	climate:cangrd:historical:annual:anomaly	Gridded annual me...	
Canadian Seasonal to Inter-annual Pre...	weather:cansips:250km:forecast:members	The Canadian Seas...	

Use title for layer name  
 Only request features overlapping the view extent

**Coordinate Reference System**

Change...

Build query Close Add Help



# Hands-on: 2

The screenshot displays the QGIS interface with the Data Source Manager dialog box open. The dialog is titled "Data Source Manager | WFS / OGC API - Features" and shows a list of server connections under the "Weather\_Canada" server. A table lists various climate data layers with columns for Title, Name, Abstract, and Sql. A small QGIS dialog box is overlaid on the table, showing "Loading features for layer climate-stations" at 99% completion. The background map shows a scatter plot of data points over a geographical area.

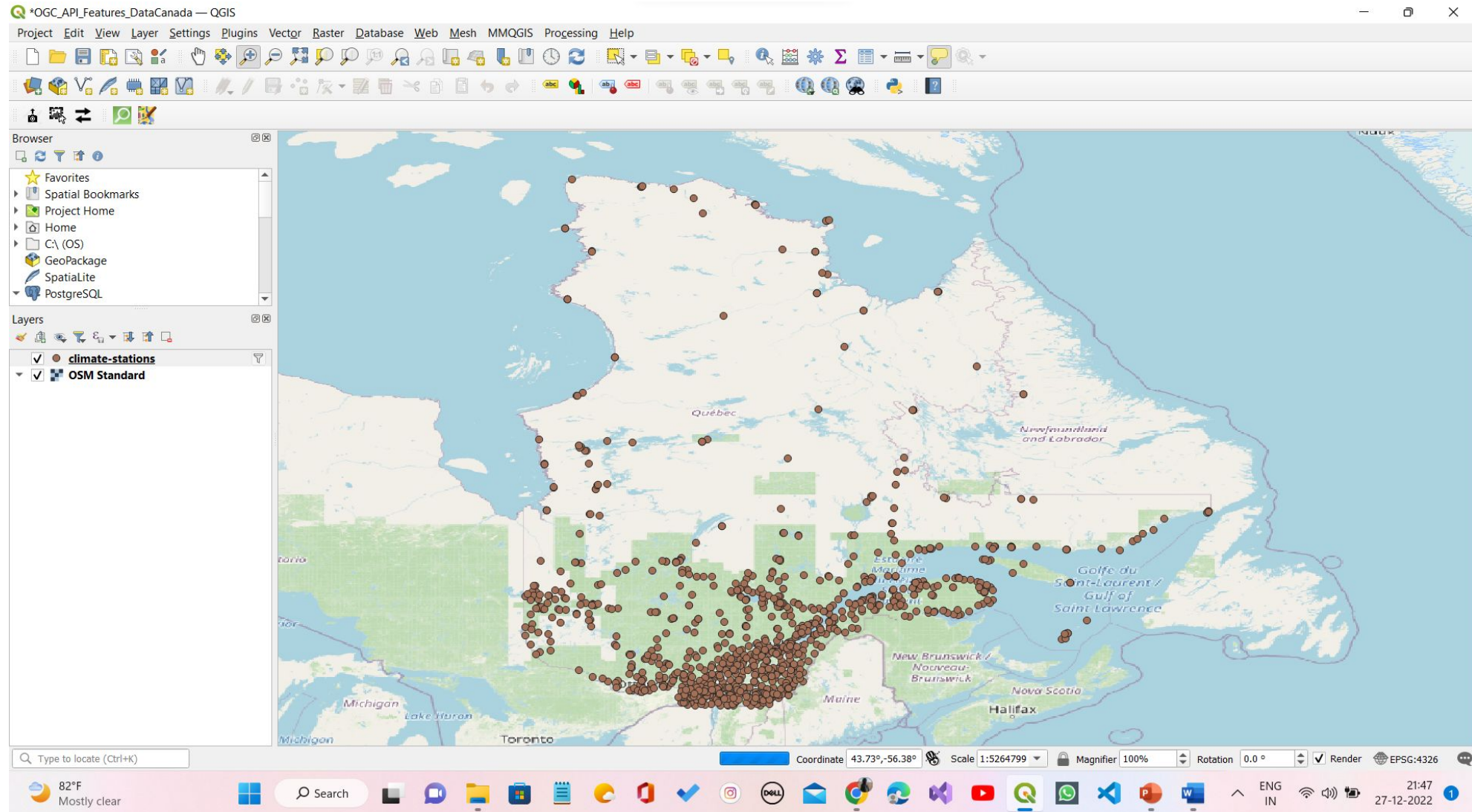
Title	Name	Abstract	Sql
Historical indices	climate:indices:historical	High-resolution stat...	
Historical annual DCS	climate:dcs:historical:annual:absolute	The statistically do...	
Historical annual CMIP5	climate:dcs:historical:annual:absolute	The Global climate ...	
Historical annual anomaly DCS	climate:dcs:historical:annual:anomaly	The statistically do...	
Historical annual anomaly CMIP5	climate:dcs:historical:annual:anomaly	The Global climate ...	
Daily Mean of Water Level or F...	climate-daily	The daily mean is th...	
Daily Extremes of Records (LTCE)	climate-daily	Anomalous weather...	
Daily Extremes of Records (LTCE)	climate-daily	Anomalous weather...	
Daily Extremes of Records (LTCE) - pre...	climate-daily	Anomalous weather...	
Daily Climate Observations	climate-daily	Daily climate obser...	
Climate Stations	climate-stations	Climate observatio...	"ENG_PROV_NA...
CanGRD historical seasonal trend	climate:cangrd:historical:seasonal:trend	CANGRD data are i...	
CanGRD historical seasonal anomaly	climate:cangrd:historical:seasonal:anom...	Gridded seasonal m...	
CanGRD historical monthly anomaly	climate:cangrd:historical:monthly:anomaly	Gridded monthly m...	
CanGRD historical annual trend	climate:cangrd:historical:annual:trend	CANGRD data are i...	
CanGRD historical annual anomaly	climate:cangrd:historical:annual:anomaly	Gridded annual me...	
Canadian Seasonal to Inter-annual Pre...	weather:cansips:250km:forecast:members	The Canadian Seas...	

# Hands-on: 2

The screenshot displays the QGIS desktop application interface. The title bar reads '\*OGC\_API\_Features\_DataCanada — QGIS'. The menu bar includes Project, Edit, View, Layer, Settings, Plugins, Vector, Raster, Database, Web, Mesh, MMQGIS, Processing, and Help. The toolbar contains various icons for file operations, navigation, and processing. On the left, the 'Browser' panel shows a file tree with 'Favorites', 'Spatial Bookmarks', 'Project Home', 'Home', 'C:\ (OS)', 'GeoPackage', 'SpatialLite', and 'PostgreSQL'. The 'Layers' panel shows two active layers: 'climate-stations' (checked) and 'OSM Standard' (checked). The main map area shows a map of Canada with a grid overlay. A dense cluster of brown circular markers is visible in the eastern part of the country, representing climate stations. The map also shows geographical features like rivers and lakes. At the bottom, the status bar displays the coordinate '82.93°, -132.97°', scale '1:9526598', magnifier '100%', rotation '0.0 °', and projection 'EPSG:4326'. The Windows taskbar at the very bottom shows the system tray with a weather widget (82°F, Mostly clear), search bar, and various application icons. The system clock shows 21:47 on 27-12-2022.



# Hands-on: 2



# THANK YOU!

---

prajwalita.chavan@gmail.com  
prajwalita@iitb.ac.in

#OGCAPI