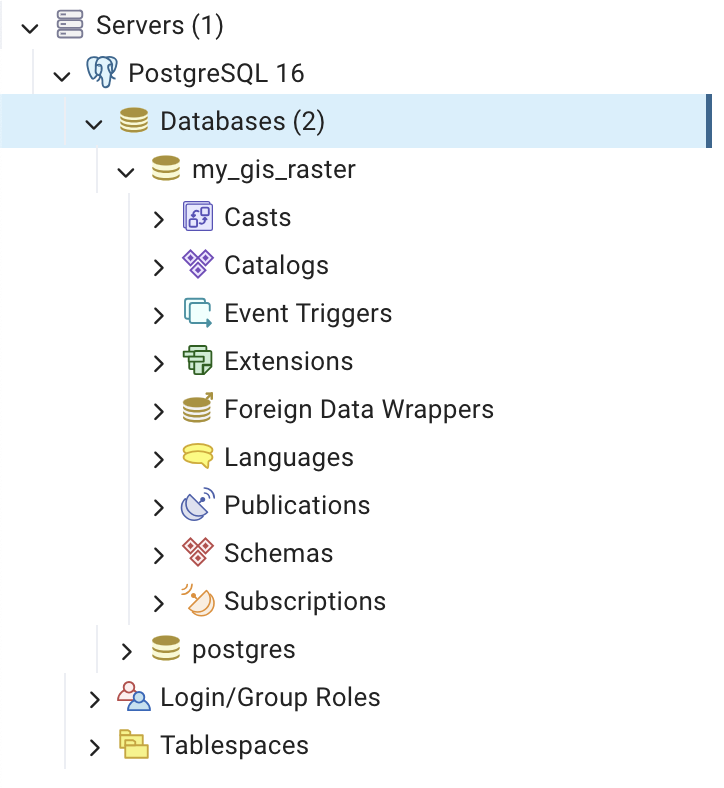
**SIGM – AULA PRÁTICA 6** **– GRUPO 2**

**1. Criar e destruir base de dados (com “template” usando *psql*)**

a) Complete o *script* \scripts\00\_script\_CRIAR\_BD\_GIS\_RASTER.txt de modo a eliminar e construir a base de dados de nome my\_gis\_raster (que irá conter, para além do formato vetorial, também dados “raster”). Execute o *script* \scripts\\_go00.bat e use o “pgAdmin 4” para confirmar que a base de dados my\_gis\_rater foi correctamente construída.

Foi criada a base de dados “my\_gis\_raster” e foram adicionadas as extensões:

* “postgis”, como usualmente.
* “postgis\_raster”, porque o tipo “raster” não está diretamente disponível na nova versão do extensor “postgis”.
* “plpython3u”, apesar de não usada, mantemos esta alternativa disponível.



**2. *Formato Vetorial* – Povoar a base de dados com “Lisboa”**

Vamos considerar o conceito de “*Volunteered Geographic Information*” (VGI) e a seguinte definição:

“*Volunteered Geographic Information (VGI) is he widespread engagement of large numbers of private citizens, often with little in the way of formal qualifications, in the creation of geographic information, a function that for centuries has been reserved to official agencies.*

*They are largely untrained and their actions are almost always voluntary, and the results may or may not be accurate.*

*But, collectively, they represent a dramatic innovation that will certainly have profound impacts on geographic information systems (GIS) and more generally on the discipline of geography and its relationship to the general public*.” [Goodchild, 2007]

O projeto OpenStreetMap (http://www.openstreetmap.org/) é um dos VGI de maior relevo; a informação aí disponível é usada por organizações como a Apple e a Google. Vamos portanto usar também a informação aí disponível para povoar a nossa base de dados!

a) Considerar a aplicação “osm2pgsql” que carrega dados em formato “osm” (de “OpenStreetMap”) numa base de dados PostGIS mantendo toda a informação do OpenStreetMap.

A aplicação esta disponível na pasta com os conteúdos desta aula (osm2pgsql.zip; versão Windows, 32bit). Original disponível em “http://wiki.openstreetmap.org/wiki/Osm2pgsql”. *Nota*: o “osm2pgsql” tem uma nova versão disponível na aula; no entanto mantem-se a versão antiga caso exista alguma incompatibilidade; ambas versões testadas e a funcionar Windows10.

Para instalar, a sugestão é que descompacte na pasta desta aula prática. Em alternativa pode colocar noutro local mas depois precisa de alterar o “path” nos ficheiros fornecidos nesta aula.

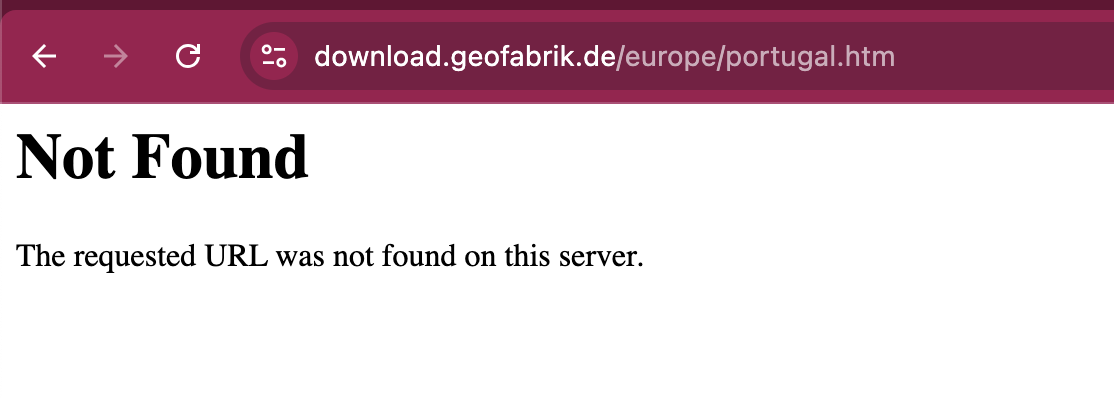
**Windows:** O “osm2pgsql” foi instalado de acordo com as indicações do guião.

**MAC OS:** Utilizou-se o comando “brew install oms2pgsql”.

b) No ficheiro lisbon\_portugal.osm.pbf (aula prática) tem a cidade de Lisboa. Para o obter foi usado o serviço em “https://mapzen.com/data/metro-extracts/metro/lisbon\_portugal/” que, à data da preparação desta aula, não estava disponível. Esta é uma organização que simplifica a extração, do OpenStreeMap, de representações de cidades. Note que em “https://mapzen.com/products/” é indicado “https://mapzen.com/data/metro-extracts/” para versão mais atualizada dos mapas; este site obriga a autenticação mas, à data da preparação desta aula, não estava disponível.

De momento vamos trabalhar apenas com o mapa de Lisboa. No entanto pode ter acesso ao de Portugal em https://download.geofabrik.de/europe/portugal.htm ou definir uma qualquer outra zona interativamente em http://www.openstreetmap.org \ Export \ Manually select a diferente area.

O site a vermelho não está disponível:



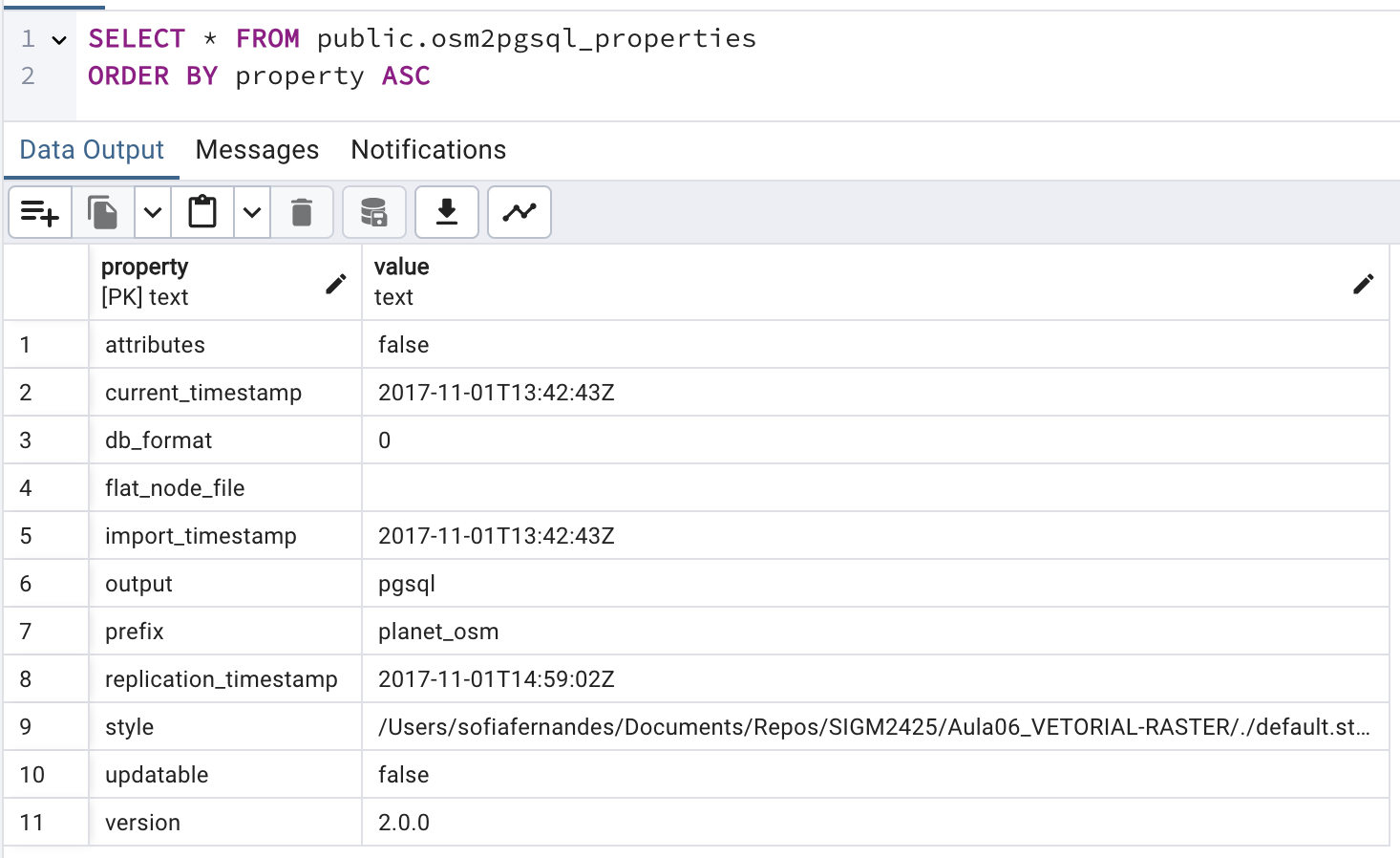
c) Neste momento está tudo preparado para “carregar Lisboa” na base de dados! Para isso analise as instruções em “\_01\_comando-osm2pgsql.txt” e execute-as; note que a “password” de acesso à base de dados precisa de ser registada em variável de ambiente. Todas as instruções estão disponíveis em “\_01\_comando-osm2pgsql.txt”.

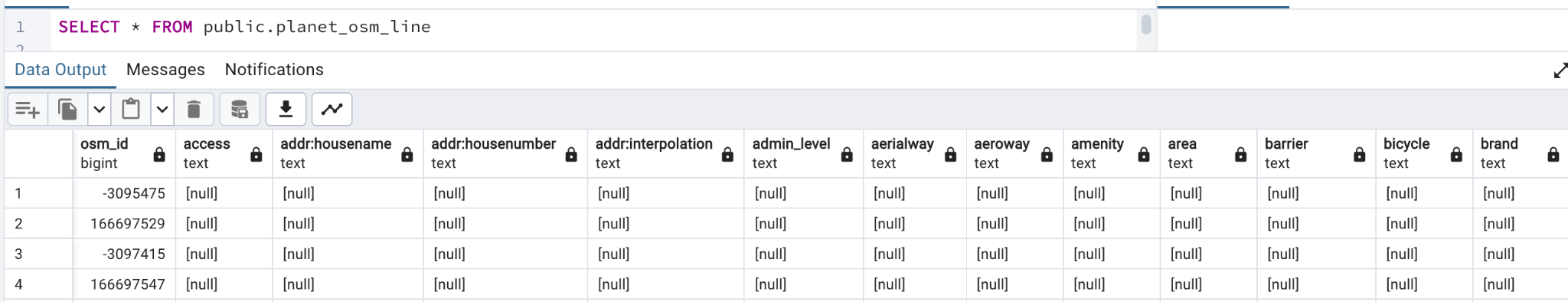
**Windows:** De acordo com o script fornecido.

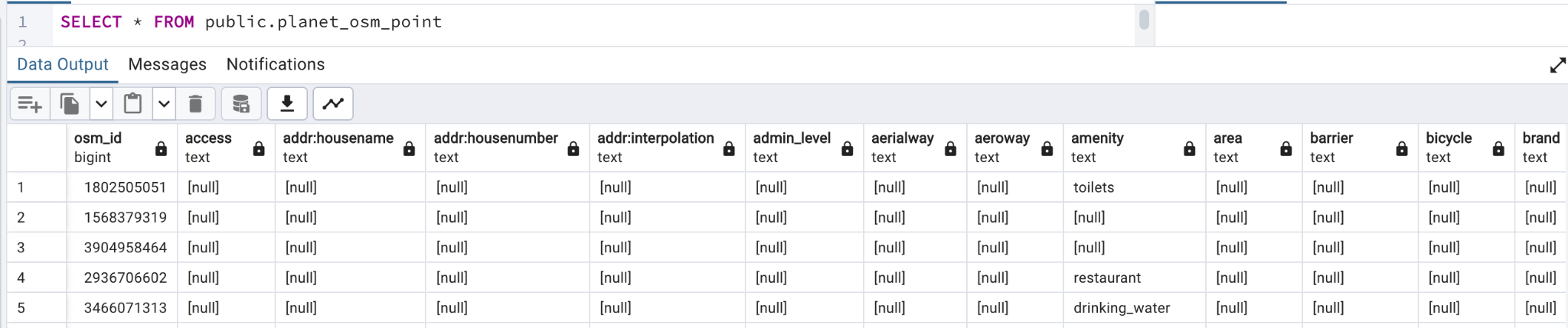
**MAC OS:** Utilizou-se o seguinte comando, colocando de seguida a password.

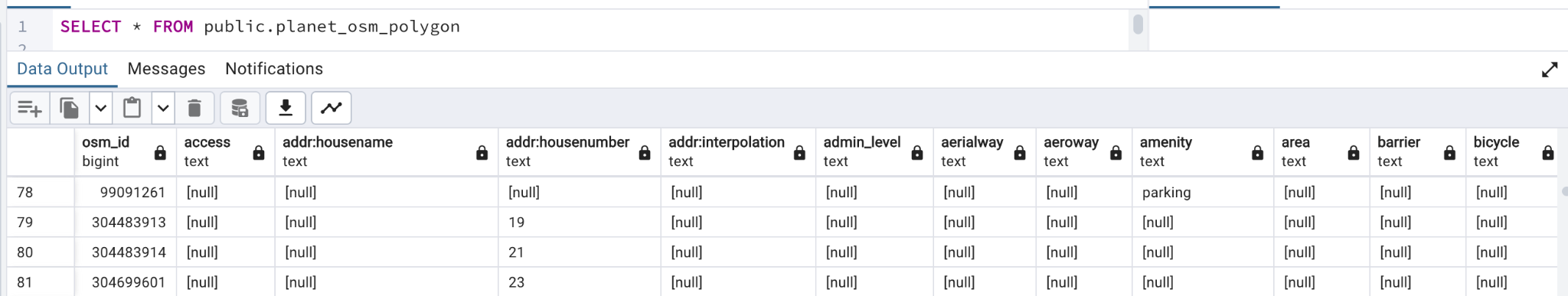
osm2pgsql -c -d my\_gis\_raster -U postgres -W -H localhost -S ./default.style ./lisbon\_portugal.osm.pbf

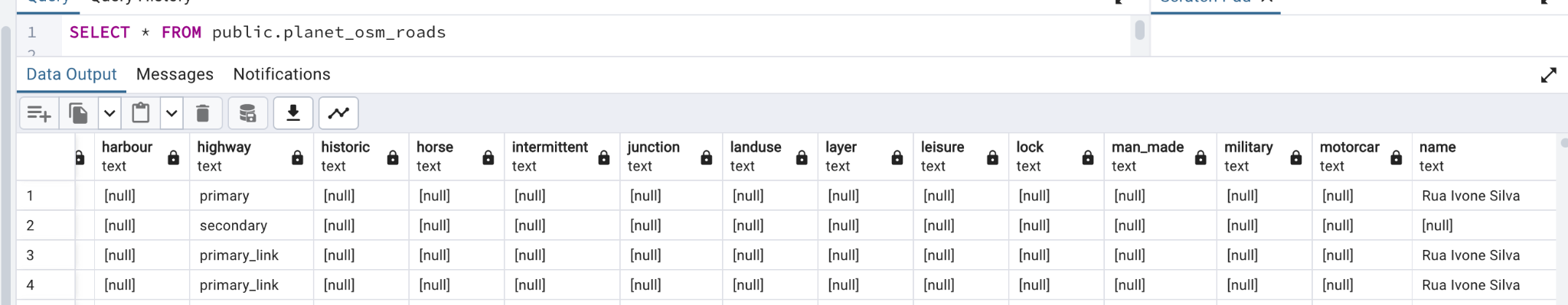
d) Depois de executar (com sucesso) o “carregamento de Lisboa” analise (via pgAdmin 4) o modelo de dados construído.





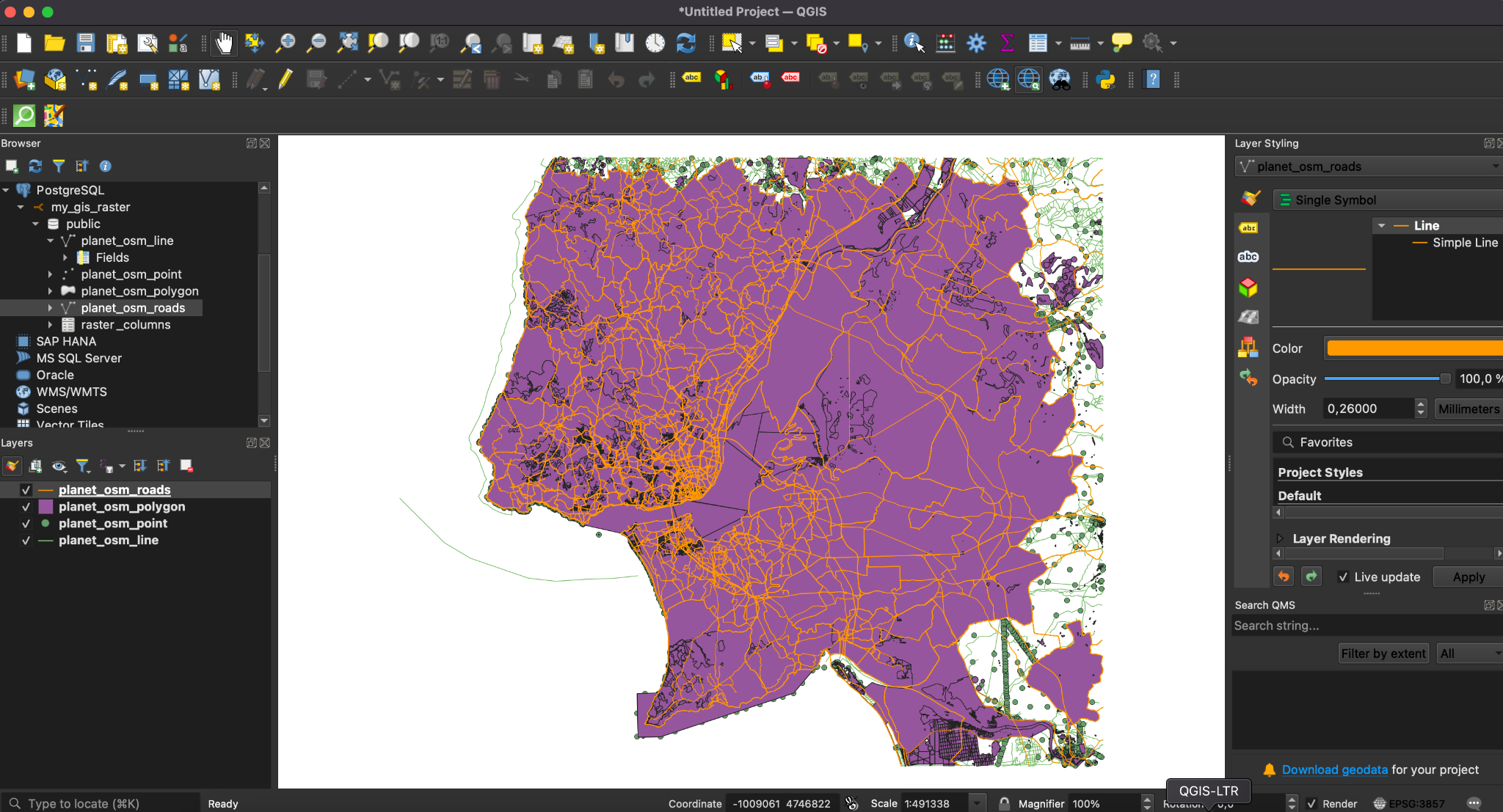




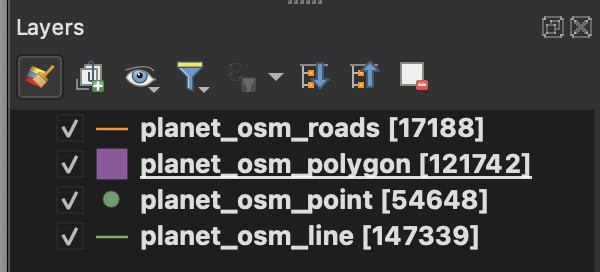


**3. … visualizar Lisboa (via QuantumGIS)**

a) Inicie o QuantumGIS e construa conexão com a base de dados “my\_gis\_raster” (Layer \ Add PostGIS Layer \ Edit \ [my\_gis\_raster | localhost | 5432 | my\_gis\_raster | postgres | *password*]).  
   
Carregue todos os “layers” disponíveis (Layer \ Add PostGIS Layer \ Connect [*selecionar-todos*]). Caso necessite de dar indicação sobre o sistema de referência (CRS – “Coordinate Reference  
   
System”) indique o: WGS84 (EPGS4326).



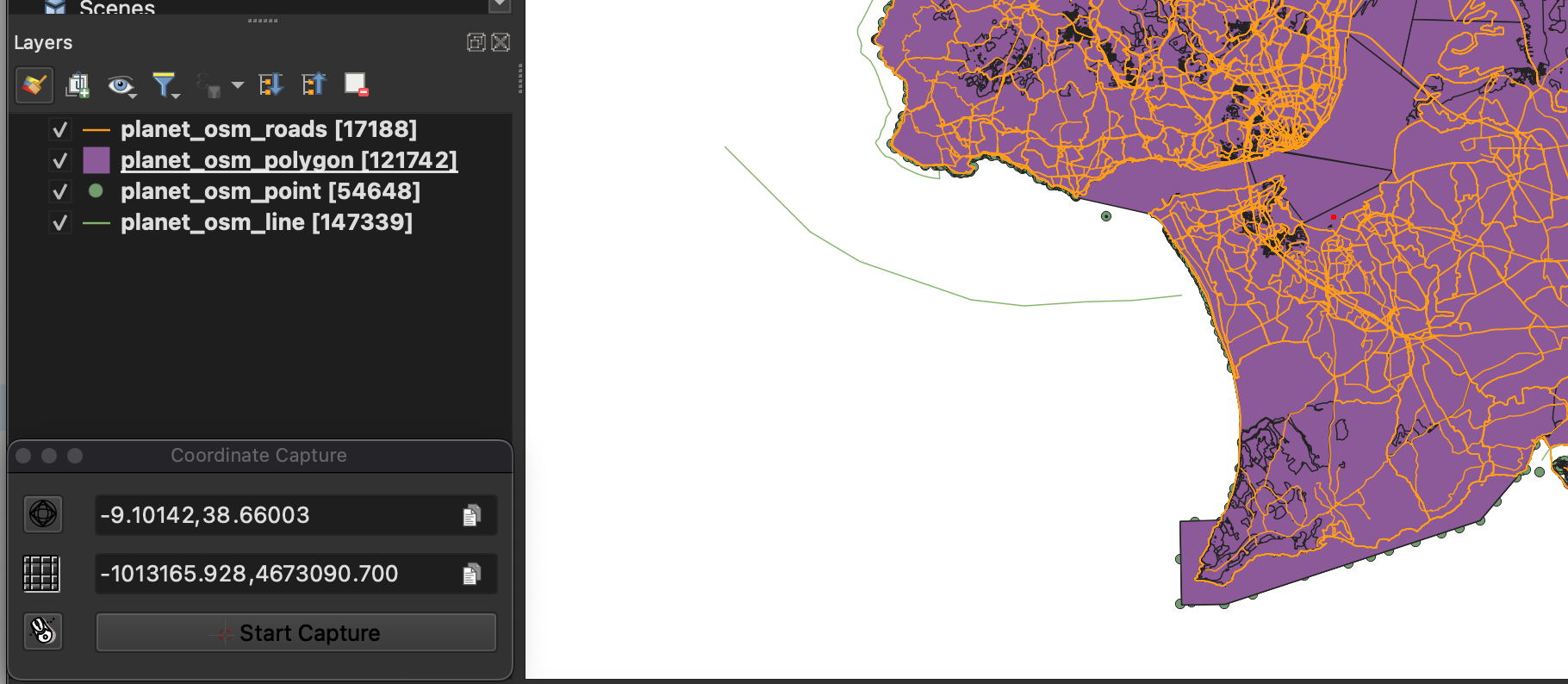
b) Ative o painel que permite escolher os “layers” a apresentar (View \ Panels \ [Layers]).

c) Apresente o número de objetos representados em cada “layer” (*selecionar-nome-do-layer* \ *botão- direito-rato* \ [Show Feature Count]).

d) Fique também como uma ideia global dos atributos alfanuméricos associados a cada “layer” (*selecionar-nome-do-layer* \ *botão-direito-rato* \ [Open Attribute Table]).

**4. ...mais interação com Lisboa**

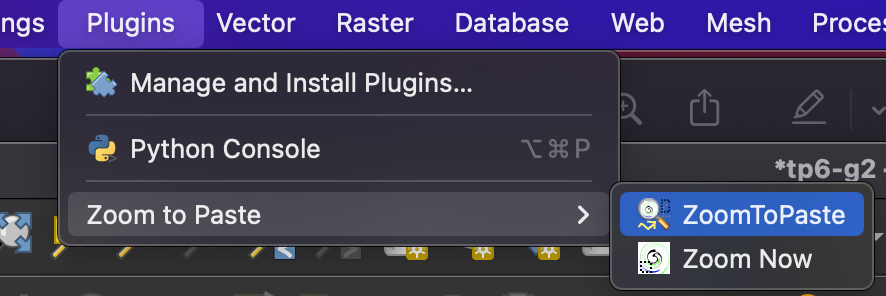
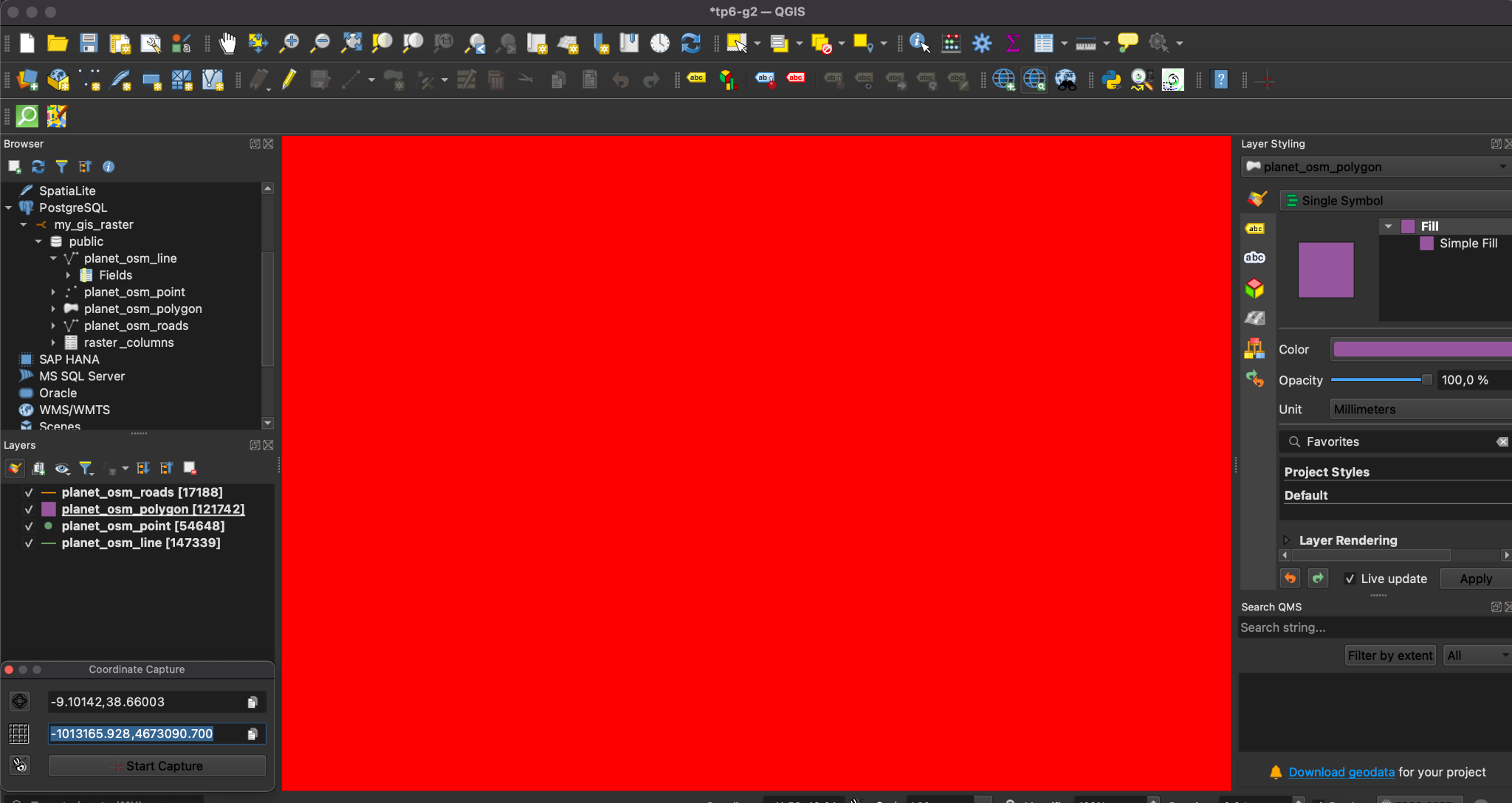
a) Ative a captura de coordenadas (Plugins \ Manage and Install Plugins... [Installed] e selecione “Coordinate Capture” e depois Vector \ Coordinate Capture) e carregue no botão que permite fazer o “mousse drag” e depois em <Start capture>. Arraste o rato (pressionando o botão) sobre Lisboa e escolha determinada coordenada.



a) Instale o “plug-in” que permite fazer “zoom” de determinada coordenada; em Plugins \ Manage and Install Plugins... [Not installed] e escolha “ZoomToPaste” carregue em <Install plugin>.

b) Depois de instalar vá a Plugins \ ZoomToPaste e use as coordenadas que capturou para fazer fazer “pan” para esse local.

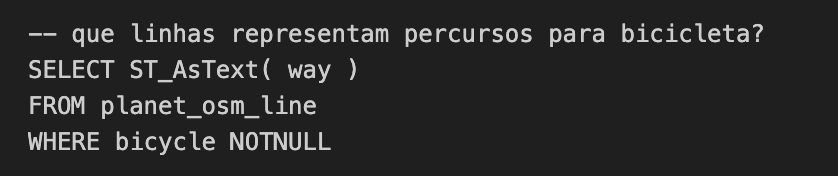
É necessário passar as coordenadas da 2ª linha, separando o X e o Y.

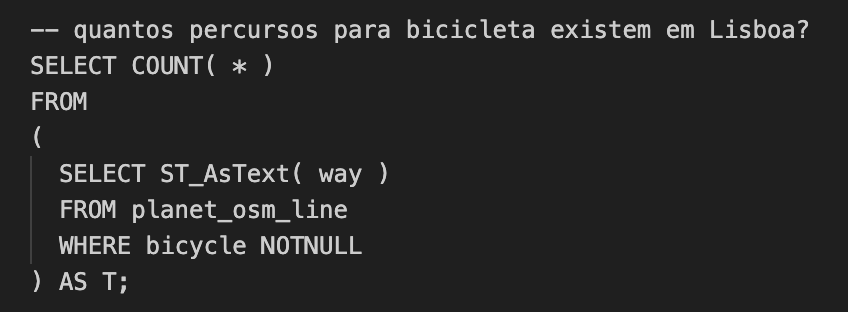
**5. ...interrogar Lisboa usando o PostGIS**

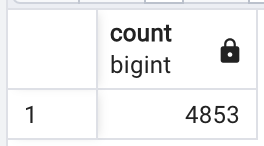
Inicie o pgAdmin e considere as diretivas SQL no ficheiro “\_02\_interrogacoes-sobre-Lisboa.txt”.

a) Obtenha as linhas que representam percursos para bicicleta.

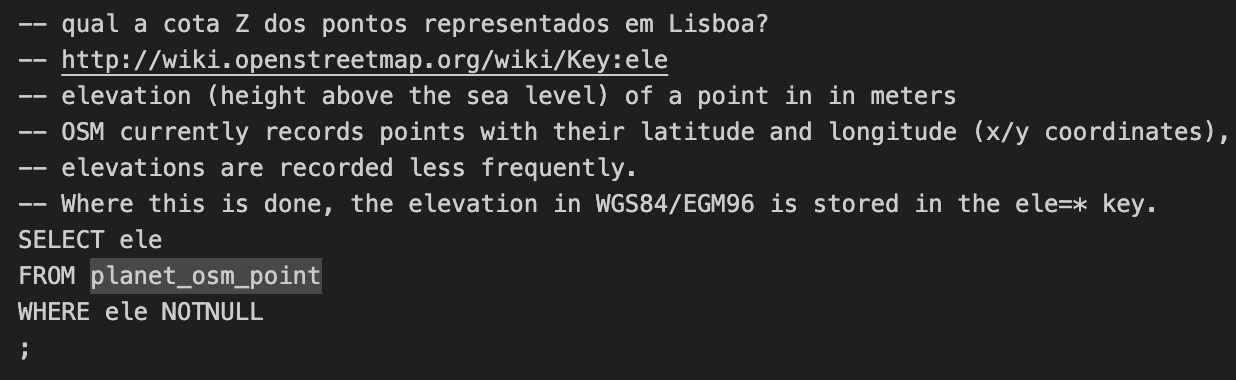
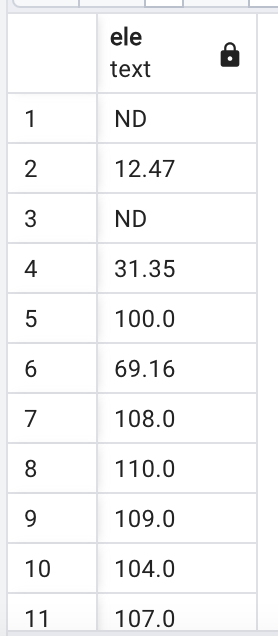




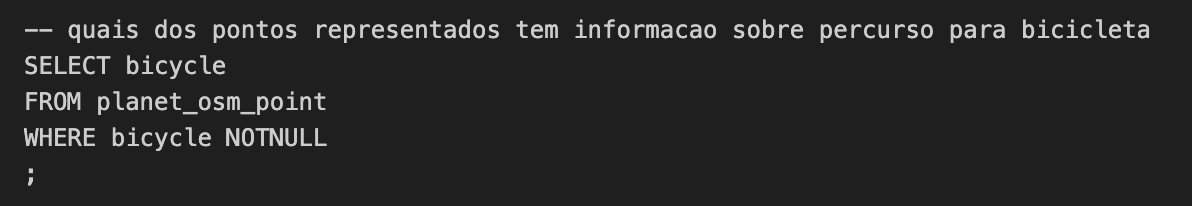
b) Quantos percursos para bicicleta existem em Lisboa?

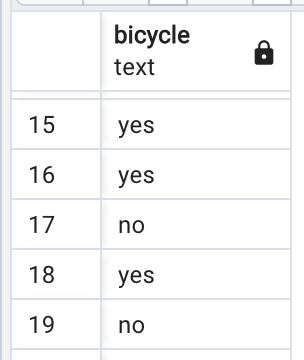


c) Qual a cota Z dos pontos representados em Lisboa?



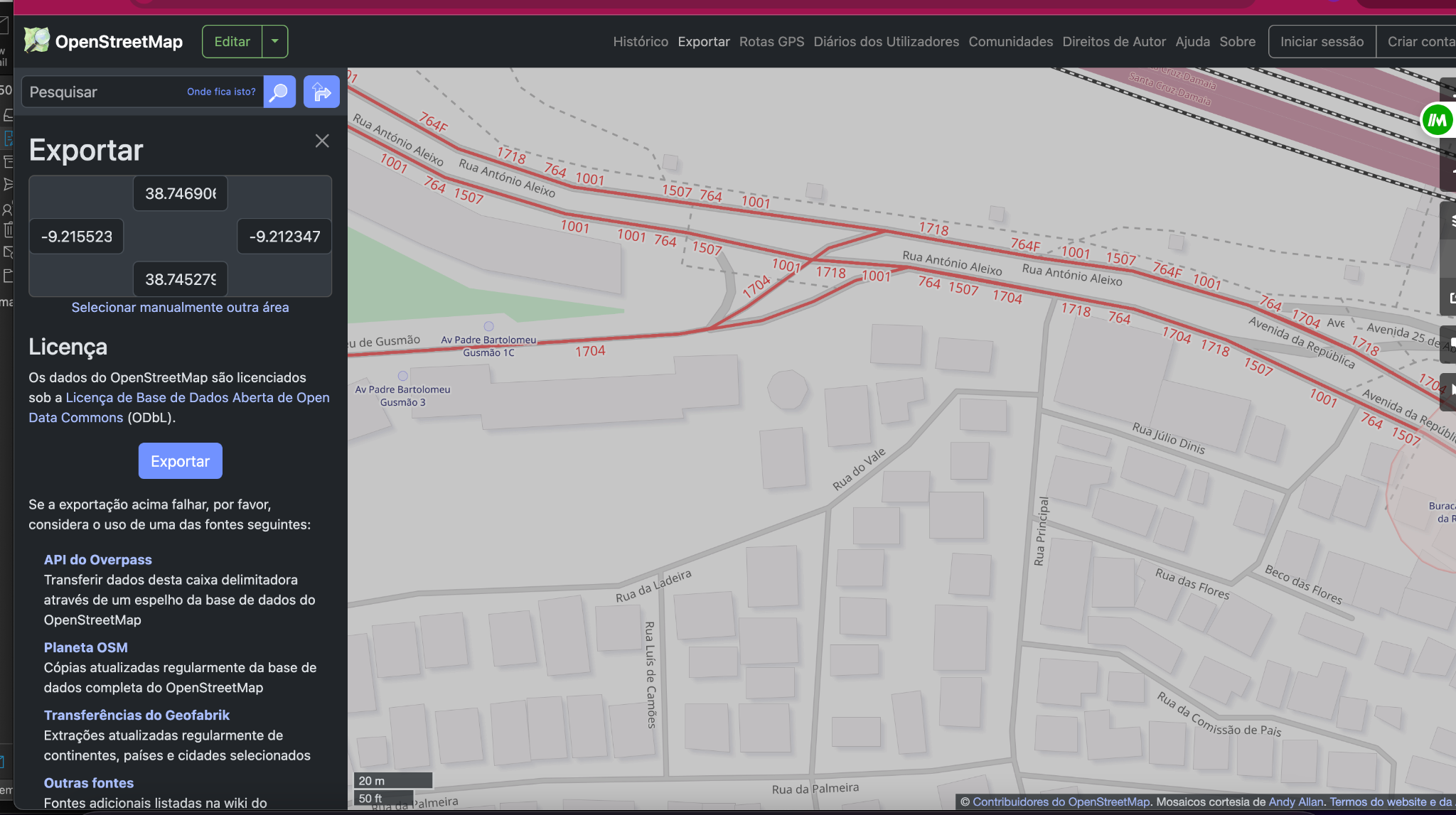
d) Quais dos pontos representados têm informação sobre percurso para bicicleta?





**6. ...outros locais obtidos diretamente do OpenStreetMap**

a) Caso pretenda explorar outros locais pode aceder diretamente ao OpenStreetMap (http://www.openstreetmap.org/) fazer o “Search” que pretender (e.g., “São Miguel, Açores”) e carregar em <Export>; em “Manually select a different area” pode redefinir o espaço a exportar.



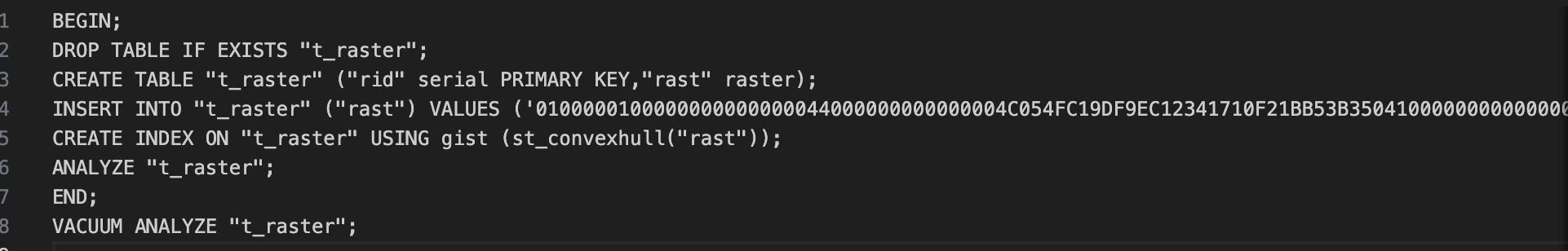
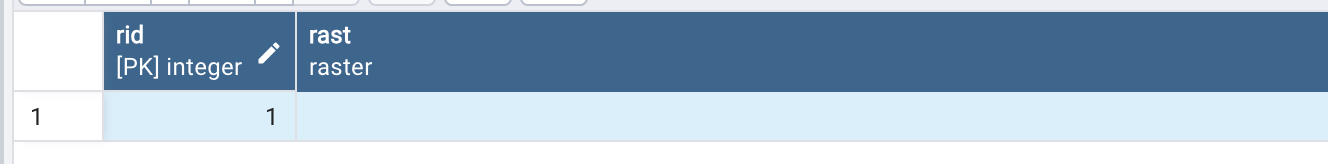
**7. *Formato “Raster” (“Bitmap”)* – Povoar com a “Ilha das Flores”**

a) Analise as instruções em “\_03\_comando-raster2pgsql.txt”. Note que a descrição das opções de  
“raster2pgsql” está disponível em: “http://postgis.net/docs/using\_raster\_dataman.html”.

O comando raster2pgsql é uma ferramenta de linha de comando que converte dados raster em um formato SQL, permitindo a sua inserção em uma tabela PostGIS. As principais opções utilizadas no script são:

* **-s 4236:** Define o Sistema de Referência Espacial (SRID) do raster como 4236, que corresponde ao sistema geodésico mundial (WGS84).
* **-I:** Cria um índice espacial (GiST) na coluna raster, otimizando as consultas espaciais.
* **-d:** Dropa (apaga) a tabela antes de criar uma nova. Use com cuidado, pois apagará todos os dados existentes na tabela.
* **-M:** Realiza um VACUUM ANALYZE na tabela, atualizando as estatísticas e otimizando o desempenho das consultas.
* **-a:** Adiciona (append) novas linhas à tabela existente, sem a criar novamente.

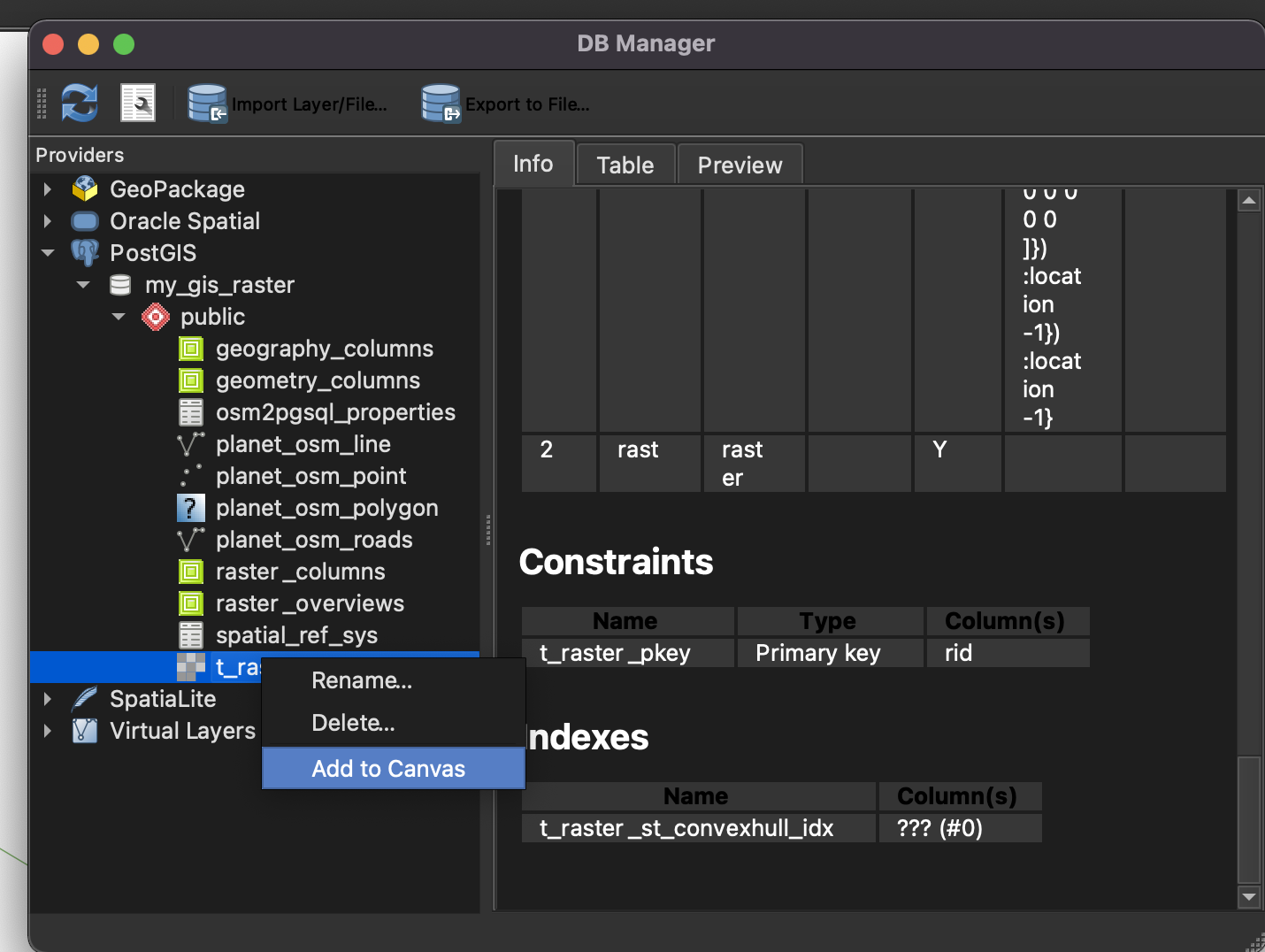
b) Analise o script “.\scripts\\_script\_CONNECT\_INIT\_BD.txt” e note que o conteúdo deste “script” vai ser concatenado ao resultado da execução de “raster2pgsql” para estabelecer a conexão com a base de dados antes de inserir os dados e assim automatizar este processo.

Ficheiro “out\_raster.txt”:   
 

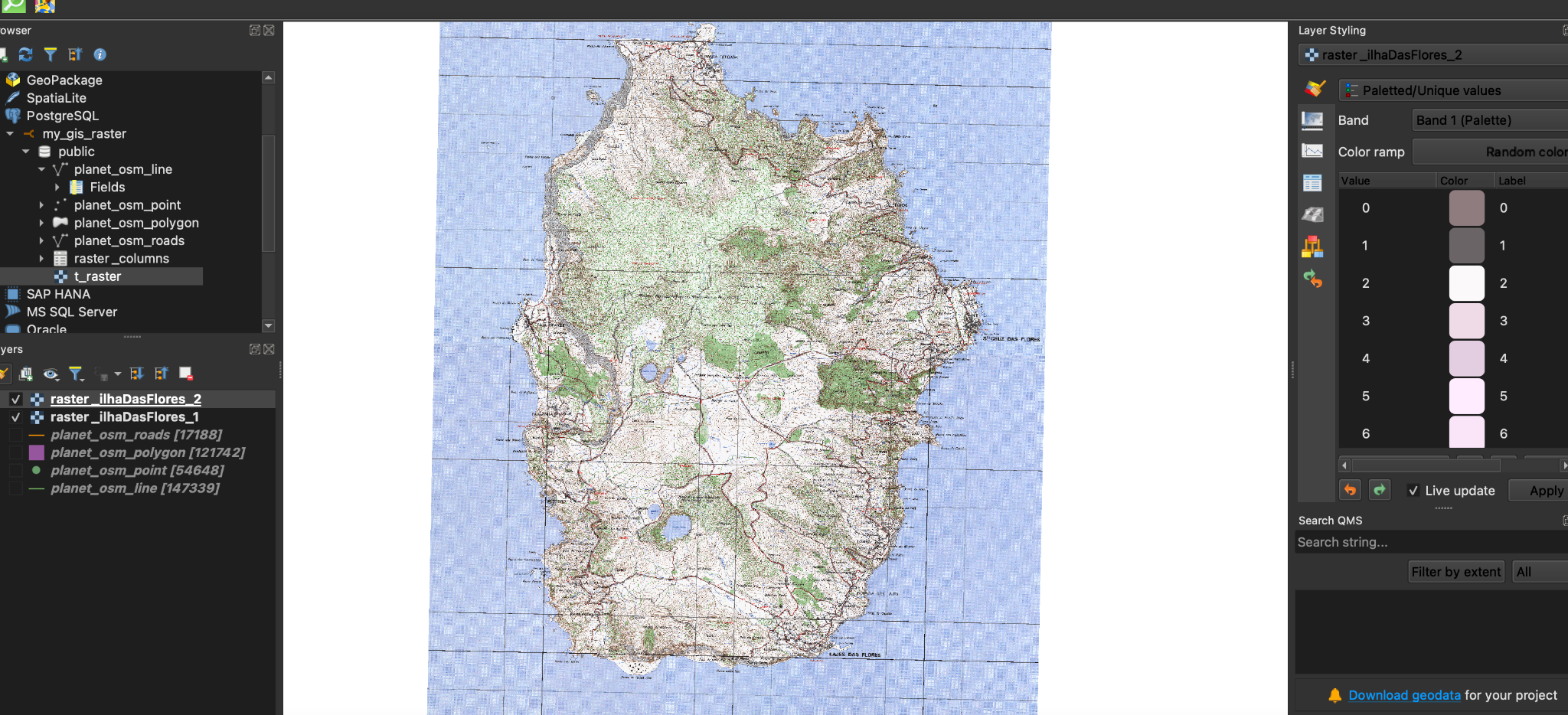
c) Execute as instruções em “\_03\_comando-raster2pgsql.txt” de modo a inserir, na base dados, os dois blocos de dados “raster” que (em conjunto) representam a “Ilha das Flores; Açores”.

**8. ...visualizar a “Ilha das Flores”(via QuantumGIS)**

a) Inicie o QuantumGIS e construa conexão com a base de dados “my\_gis\_raster” (Layer \ Add PostGIS Layer \ Edit \ [my\_gis\_raster | localhost | 5432 | my\_gis\_raster | postgres | *password*]).

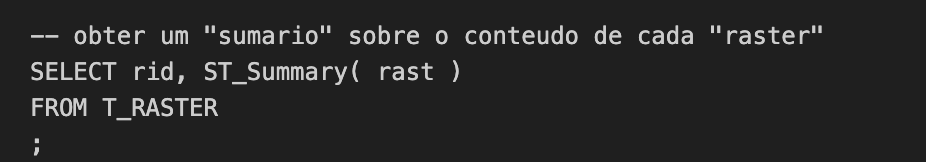
b) Carregue os “layer raster” escolhendo, na interface do QuantumGIS: “\Database \ DB Manager \ DB Manager \ [PostGIS | my\_gis\_raster | public | t\_raster] e com o cursor em “t\_raster” escolha o botão direito e <Add to canvas>  


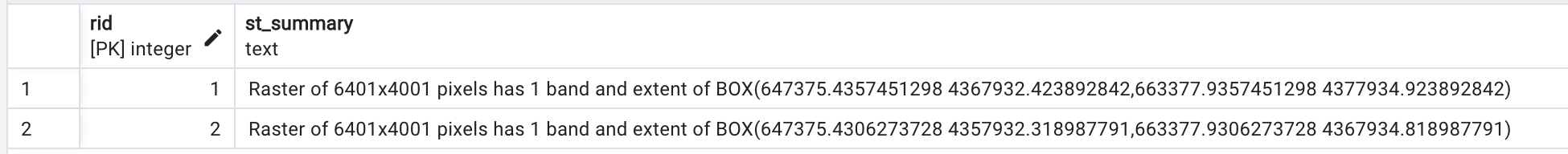
c) O “layer” deve aparecer em “grayscale” mas selecionando o “layer” e escolhendo “\Layer \ Properties.... \ Hue [x] Colorize” define uma conversão mais agradável!



**9. ...interrogar o “raster” (“bitmap”) usando o PostGIS**

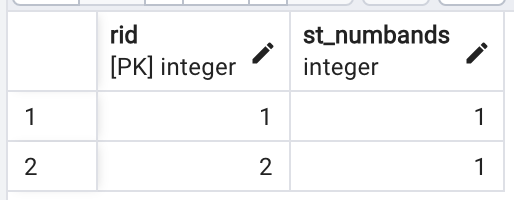
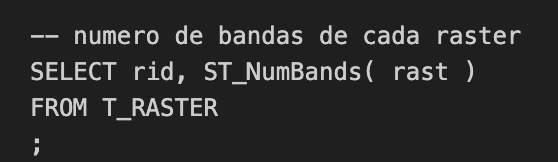
Inicie o pgAdmin e considere as diretivas SQL no ficheiro “\_04\_interrogacoes-sobre-raster.txt”. Note que as diretivas de manipulação de “ratser” estão em: http://postgis.net/docs/RT\_reference.html.

a) Obtenha um “sumário” sobre a estrutura dos “raster” já carregados (em alínea anterior).  
 

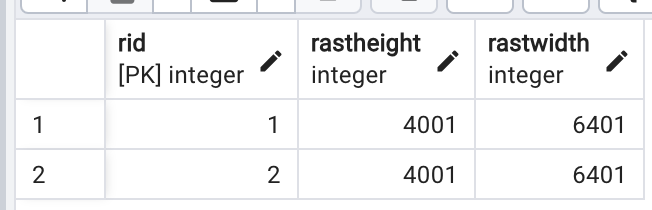
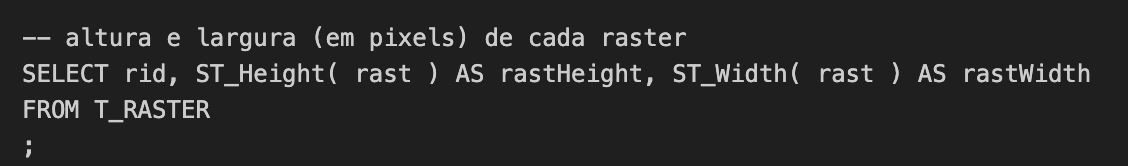


Utiliza a função ST\_Summary para obter um resumo conciso de cada raster, incluindo o número de bandas, dimensões, extensão geográfica e tipo de dados.

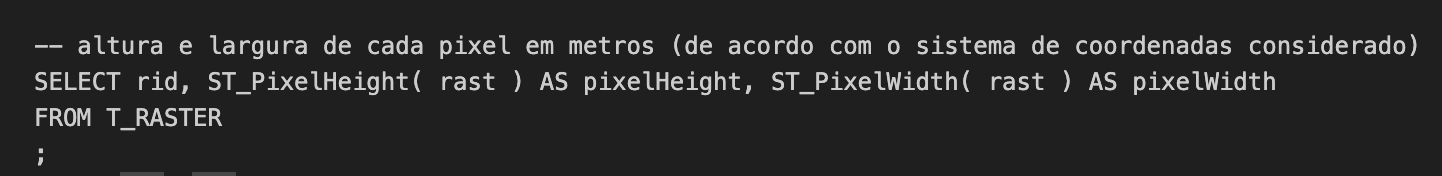
b) Qual o número de “bandas” (“canais”) de cada “raster” (“bitmap”)?

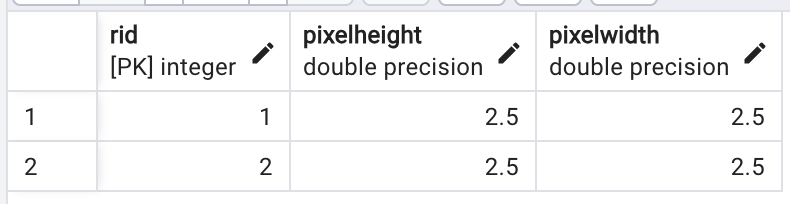


Verifica o número de bandas em cada raster utilizando ST\_NumBands. Bandas representam diferentes canais de informação num raster, como vermelho, verde e azul em imagens coloridas.

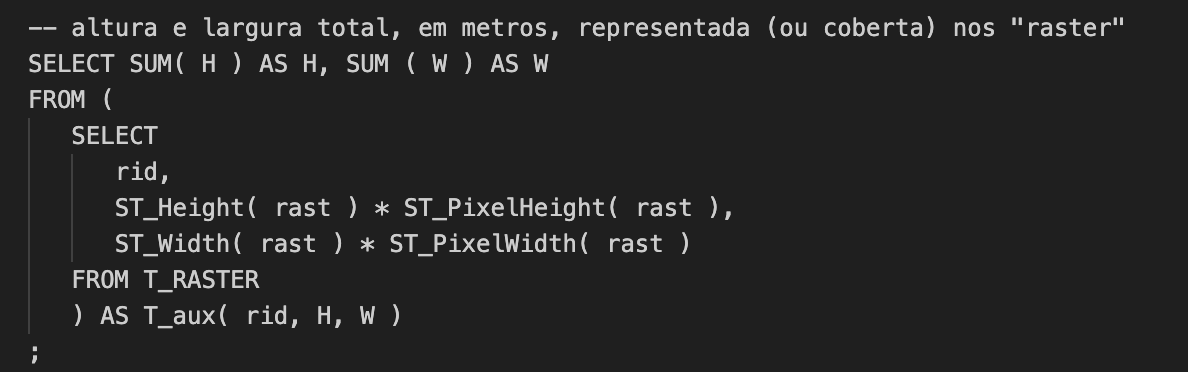
c) Qual a altura e largura, em número de pixéis, de cada “raster”?  
 

Utiliza-se ST\_Height e ST\_Width para obter as dimensões de cada raster em termos de número de pixels.

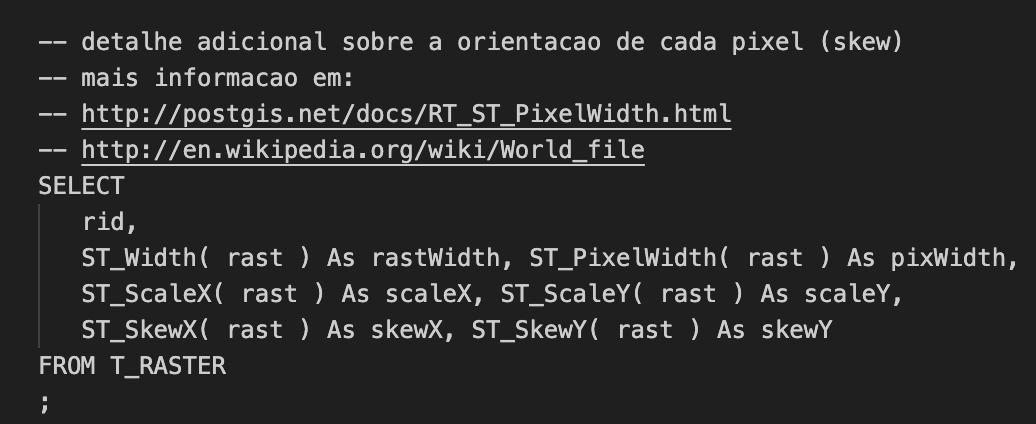
d) Qual a altura e largura, em metros, de cada pixel de cada “raster”?  
 



As funções ST\_PixelHeight e ST\_PixelWidth são utilizadas para calcular o tamanho físico de cada pixel em metros, considerando o sistema de coordenadas geográficas associado ao raster.

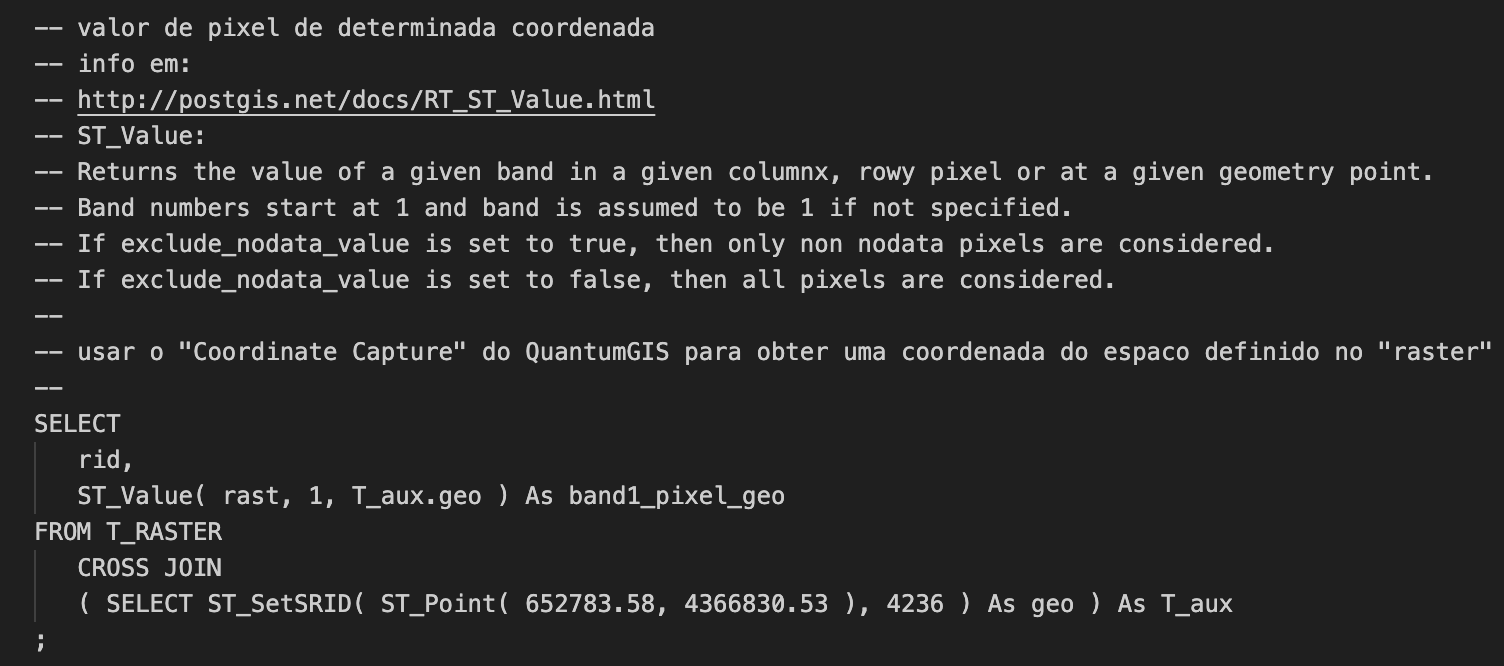
e) Qual a altura e largura total, em metros, representada pelos “raster”?  
 

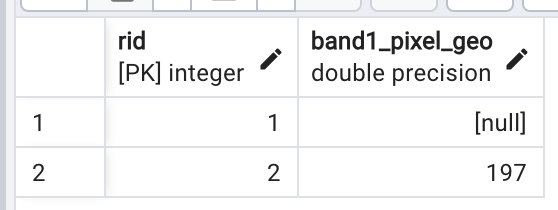


f) Qual o valor do pixel que corresponde ao ponto ST\_Point( 652783.58, 4366830.53 )? Note que aqui a “Ilha das Flores” se representa em 2 “raster” pelo que apenas 1 deles irá conter o ponto.  




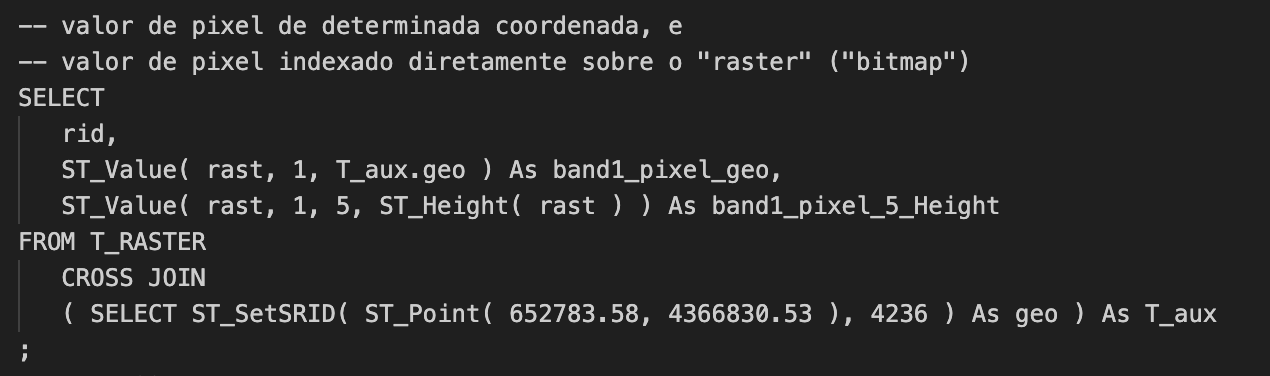
As funções ST\_ScaleX, ST\_ScaleY, ST\_SkewX e ST\_SkewY fornecem informações sobre a orientação dos pixels no espaço. Isso é importante para entender se os pixels estão alinhados com os eixos cartesianos e se há alguma rotação ou inclinação.

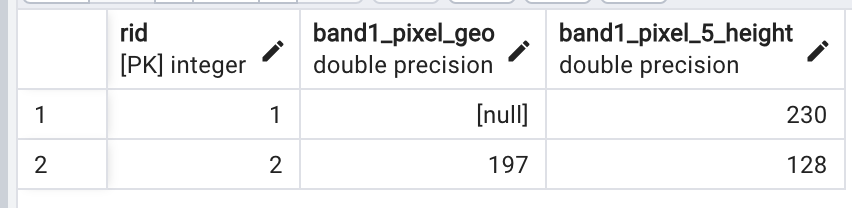
g) Altere a interrogação anterior para obter o valor de algum outro ponto do seu interesse; note que pode sempre recorrer a “Coordinate Capture” (alínea anterior) para obter coordenadas.   
 



A função ST\_Value é utilizada para obter o valor de um pixel em uma coordenada geográfica específica. Isso permite analisar os valores numéricos associados a diferentes pontos no raster.

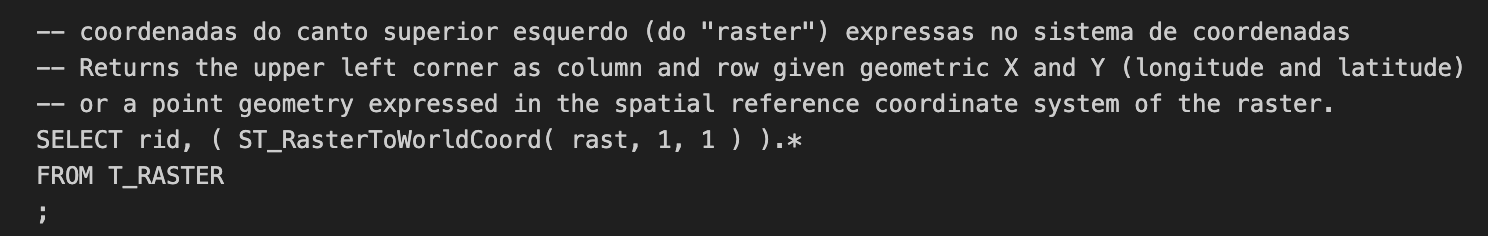
h) Para além da informação anterior projete também o valor do pixel na coluna 5 e última linha (corresponde ao valor de “height”) de cada “raster”; i.e., indexação direta do “raster” (“bitmap”). Note estas questões projetam sempre a “banda” 1 pois estes “raster” apenas têm 1 banda.

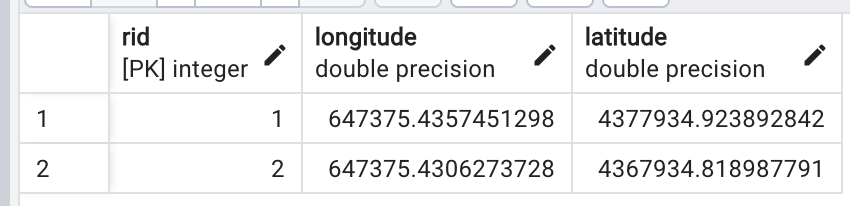




Esta consulta obtém o valor de um pixel pelo seu índice (linha e coluna) dentro da matriz de pixels.

i) Quais as coordenadas do canto superior esquerdo (do “raster”) expressas no sistema de coordenadas?





A função ST\_RasterToWorldCoord converte a posição do pixel superior esquerdo (1,1) para suas coordenadas geográficas correspondentes, fornecendo um ponto de referência para o raster no espaço.