8 Import Excel worksheet and add geometry

8.1 Make a table of PostgreSQL for a POINT data layer
8.2 Add columns using QGIS
8.3 Export CSV file from Excel worksheet
8.4 Import CSV file to PostgreSQL
8.5 Add geometry using georeferencing information in the same table

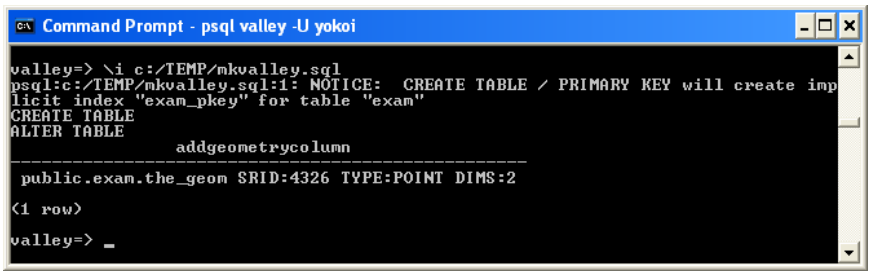
8.1 Make a table of PostgreSQL for a POINT data layer

Add new table “exam” in the database “valley”. This procedure has been described in “2_creating_vecor_layer.ppt”.
Copy the sql batch file “D:/batch_sql/mkpoint.sql” to “C:/TEMP/mkvalley.sql”.
Edit “C:/TEMP/mkvalley.sql” using WordPad as shown below.

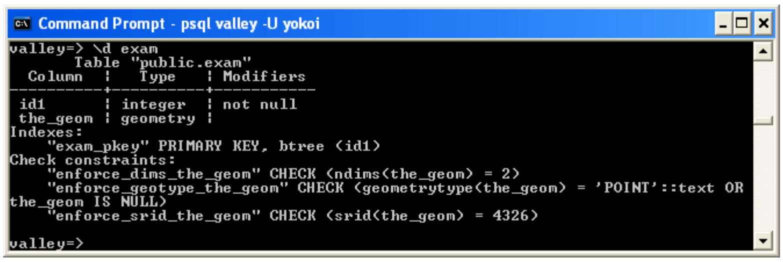
```
CREATE TABLE exam (id1 integer NOT NULL, CONSTRAINT
  exam_pkey PRIMARY KEY (id1)) WITHOUT OIDS;
ALTER TABLE exam OWNER TO yokoi;
select AddGeometryColumn(‘exam', 'the_geom', 4326, 'POINT', 2);
```

where the changed parts are shown blue, SRID “4326” means wgs84.
Execute the sql batch file “C:/TEMP/mkvalley.sql” using “¥i” command.

```
¥i C:/TEMP/mkvalley.sql
```

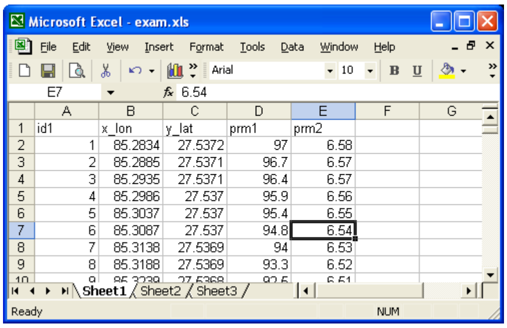


```
valley=> \i c:/TEMP/mkvalley.sql
psql:c:/TEMP/mkvalley.sql:1: NOTICE:  CREATE TABLE / PRIMARY KEY will create imp
licit index "exam_pkey" for table "exam"
CREATE TABLE
ALTER TABLE
                addgeometrycolumn
--------------------------------------------------
 public.exam.the_geom SRID:4326 TYPE:POINT DIMS:2

(1 row)

valley=> _
```



```
valley=> \d exam
        Table "public.exam"
  Column  |   Type   | Modifiers
----------+----------+-----------
 id1      | integer  | not null
 the_geom | geometry |
Indexes:
    "exam_pkey" PRIMARY KEY, btree (id1)
Check constraints:
    "enforce_dims_the_geom" CHECK (ndims(the_geom) = 2)
    "enforce_geotype_the_geom" CHECK (geometrytype(the_geom) = 'POINT'::text OR
the_geom IS NULL)
    "enforce_srid_the_geom" CHECK (srid(the_geom) = 4326)

valley=>
```

Created table "exam" has only two columns: "id1" and "the_geom", whereas the Excel worksheet "exam.xls" that will be imported has other columns as shown bellow.

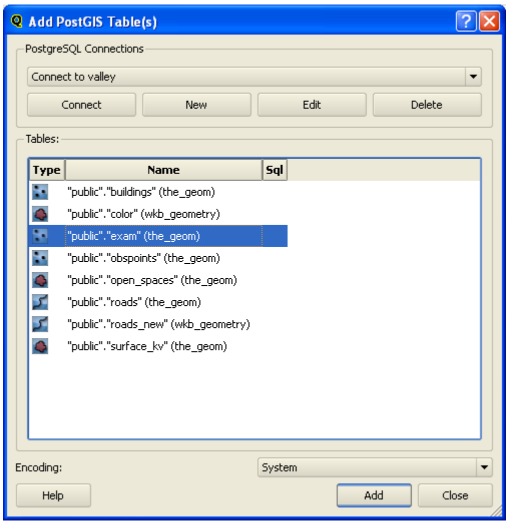


E7 6.54

	A	B	C	D	E	F	G
1	id1	x_lon	y_lat	prm1	prm2		
2	1	85.2834	27.5372	97	6.58		
3	2	85.2885	27.5371	96.7	6.57		
4	3	85.2935	27.5371	96.4	6.57		
5	4	85.2986	27.537	95.9	6.56		
6	5	85.3037	27.537	95.4	6.55		
7	6	85.3087	27.537	94.8	6.54		
8	7	85.3138	27.5369	94	6.53		
9	8	85.3188	27.5369	93.3	6.52		
10	9	85.3239	27.5368	92.5	6.51		

8.2 Add columns using QGIS

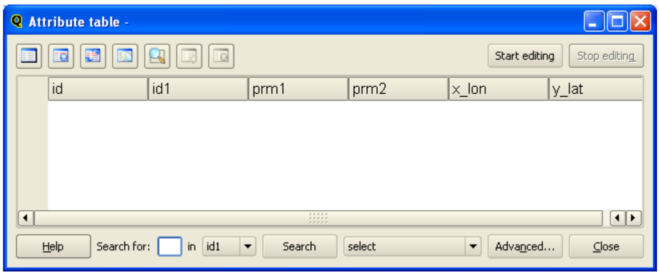
Open QGIS and connect to the table "exam" of the database "valley" using "Add PostGIS layer" button.



Connect to valley

"public"."buildings" (the_geom)
"public"."color" (wkb_geometry)
"public"."exam" (the_geom)
"public"."obspoints" (the_geom)
"public"."open_spaces" (the_geom)
"public"."roads" (the_geom)
"public"."roads_new" (wkb_geometry)
"public"."surface_kv" (the_geom)

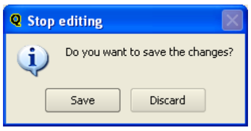
Encoding: System



id	id1	prm1	prm2	x_lon	y_lat

Open “Attribute Table” using “Open Table” button. Click “Start Editing” button.

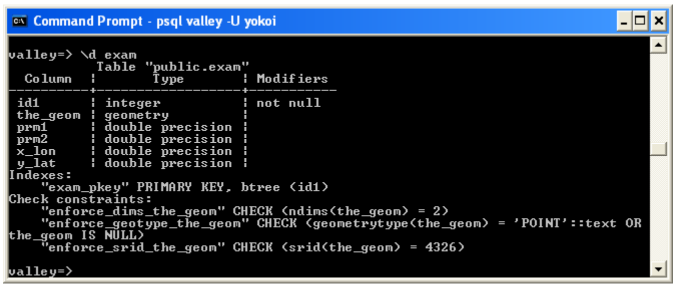
Add columns “prm1”, “prm2”, ”x_lon”, ”y_lat” with type “double precision” using “New Column” button. Then, click “Stop Editing” button.



Do you want to save the changes?

Click “Save”.

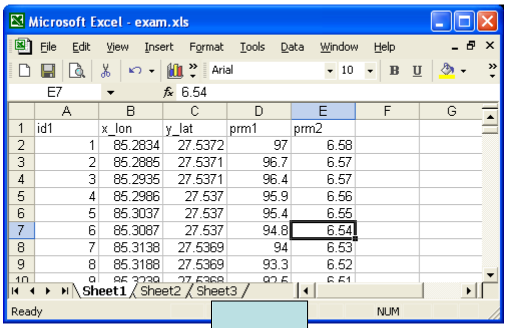
Check the table “exam” in “Command Prompt” of PostgreSQL.
¥d exam



```
valley=> \d exam
            Table "public.exam"
  Column   |       Type       | Modifiers
-----------+------------------+-----------
 id1       | integer          | not null
 the_geom  | geometry         |
 prm1      | double precision |
 prm2      | double precision |
 x_lon     | double precision |
 y_lat     | double precision |
Indexes:
    "exam_pkey" PRIMARY KEY, btree (id1)
Check constraints:
    "enforce_dims_the_geom" CHECK (ndims(the_geom) = 2)
    "enforce_geotype_the_geom" CHECK (geometrytype(the_geom) = 'POINT'::text OR
the_geom IS NULL)
    "enforce_srid_the_geom" CHECK (srid(the_geom) = 4326)

valley=>
```

Pay attention to the order of columns. The file being imported must have the same column structure.

8.3 Export CSV file from Excel worksheet

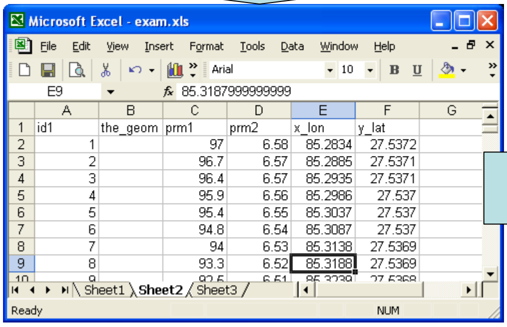

Microsoft Excel - exam.xls
File Edit View Insert Format Tools Data Window Help
Arial 10
Sheet1 Sheet2 Sheet3
Ready
NUM

E7 6.54

	A	B	C	D	E	F	G
1	id1	x_lon	y_lat	prm1	prm2		
2	1	85.2834	27.5372	97	6.58		
3	2	85.2885	27.5371	96.7	6.57		
4	3	85.2935	27.5371	96.4	6.57		
5	4	85.2986	27.537	95.9	6.56		
6	5	85.3037	27.537	95.4	6.55		
7	6	85.3087	27.537	94.8	6.54		
8	7	85.3138	27.5369	94	6.53		
9	8	85.3188	27.5369	93.3	6.52		
10	9	85.3239	27.5368	92.5	6.51		

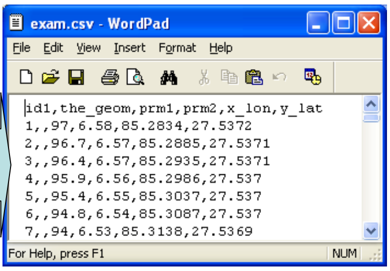
Change the order of columns and add a new column for “the_geom” on Excel worksheets.

Then, save it as a csv file.


Microsoft Excel - exam.xls
File Edit View Insert Format Tools Data Window Help
Arial 10
Sheet1 Sheet2 Sheet3
Ready
NUM

E9 85.3187999999999

	A	B	C	D	E	F	G
1	id1	the_geom	prm1	prm2	x_lon	y_lat	
2	1		97	6.58	85.2834	27.5372	
3	2		96.7	6.57	85.2885	27.5371	
4	3		96.4	6.57	85.2935	27.5371	
5	4		95.9	6.56	85.2986	27.537	
6	5		95.4	6.55	85.3037	27.537	
7	6		94.8	6.54	85.3087	27.537	
8	7		94	6.53	85.3138	27.5369	
9	8		93.3	6.52	85.3188	27.5369	
10	9		92.5	6.51	85.3239	27.5368	

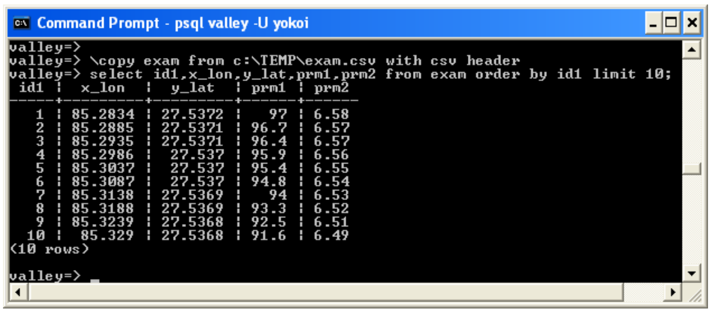


```
id1,the_geom,prm1,prm2,x_lon,y_lat
1,,97,6.58,85.2834,27.5372
2,,96.7,6.57,85.2885,27.5371
3,,96.4,6.57,85.2935,27.5371
4,,95.9,6.56,85.2986,27.537
5,,95.4,6.55,85.3037,27.537
6,,94.8,6.54,85.3087,27.537
7,,94,6.53,85.3138,27.5369
```

8.4 Import CSV file to PostgreSQL

Import “exam.csv” to PostgreSQL.

¥copy exam from c:¥TEMP¥exam.csv with csv header



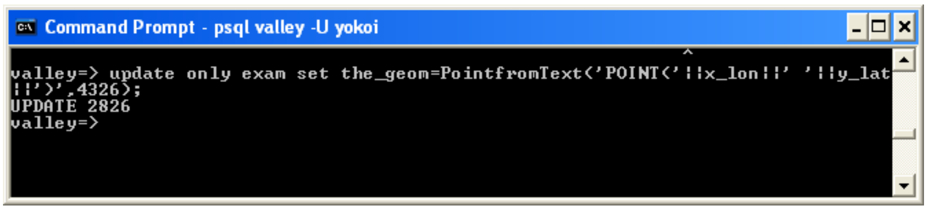
```
valley=>
valley=> \copy exam from c:\TEMP\exam.csv with csv header
valley=> select id1,x_lon,y_lat,prm1,prm2 from exam order by id1 limit 10;
 id1 |  x_lon   |  y_lat   | prm1 | prm2
-----+----------+----------+------+------
   1 | 85.2834  | 27.5372  |   97 | 6.58
   2 | 85.2885  | 27.5371  | 96.7 | 6.57
   3 | 85.2935  | 27.5371  | 96.4 | 6.57
   4 | 85.2986  |  27.537  | 95.9 | 6.56
   5 | 85.3037  |  27.537  | 95.4 | 6.55
   6 | 85.3087  |  27.537  | 94.8 | 6.54
   7 | 85.3138  | 27.5369  |   94 | 6.53
   8 | 85.3188  | 27.5369  | 93.3 | 6.52
   9 | 85.3239  | 27.5368  | 92.5 | 6.51
  10 |  85.329  | 27.5368  | 91.6 | 6.49
(10 rows)

valley=> _
```

Check the table “exam” by browsing the first 10 rows.

select id1,x_lon,y_lat,prm1,prm2, from exam order by id1 limit 10;

8.5 Add geometry using georeferencing information in the same table

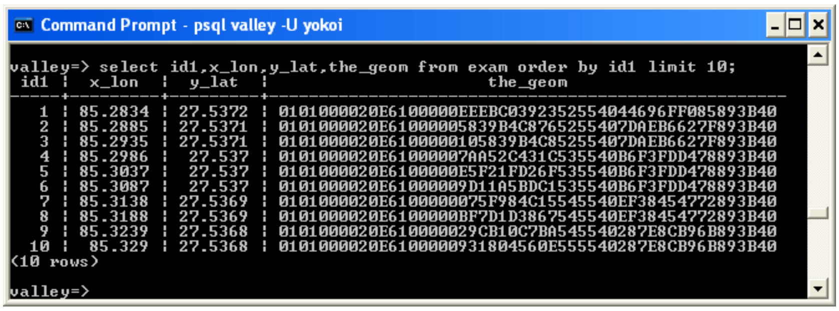
update only exam set the_geom=PointfromText(‘POINT(‘||x_lon||’ ‘||y_lat||’)’,4326);



```
valley=> update only exam set the_geom=PointfromText(’POINT(’||x_lon||’ ’||y_lat
||’)’,4326);
UPDATE 2826
valley=>
```

Check “the_geom” column by browsing the first 10 row.

select id1,x_lon,y_lat,the_geom from exam order by id1 limit 10;



```
valley=> select id1,x_lon,y_lat,the_geom from exam order by id1 limit 10;
 id1 |  x_lon  |  y_lat  |                      the_geom
-----+---------+---------+----------------------------------------------------
   1 | 85.2834 | 27.5372 | 0101000020E6100000EEEBC0392352554044696FF085893B40
   2 | 85.2885 | 27.5371 | 0101000020E61000005839B4C8765255407DAEB6627F893B40
   3 | 85.2935 | 27.5371 | 0101000020E6100000105839B4C85255407DAEB6627F893B40
   4 | 85.2986 |  27.537 | 0101000020E61000007AA52C431C535540B6F3FDD478893B40
   5 | 85.3037 |  27.537 | 0101000020E6100000E5F21FD26F535540B6F3FDD478893B40
   6 | 85.3087 |  27.537 | 0101000020E61000009D11A5BDC1535540B6F3FDD478893B40
   7 | 85.3138 | 27.5369 | 0101000020E610000075F984C155545540EF38454772893B40
   8 | 85.3188 | 27.5369 | 0101000020E6100000BF7D1D3867545540EF38454772893B40
   9 | 85.3239 | 27.5368 | 0101000020E610000029CB10C7BA545540287E8CB96B893B40
  10 |  85.329 | 27.5368 | 0101000020E6100000931804560E555540287E8CB96B893B40
(10 rows)

valley=>
```