Spatial Databases

Dr Claire Ellul

c.ellul@ucl.ac.uk

Assignment Progress

- By now you should have:
  - Created your system specification
  - Created your conceptual and logical diagrams and written the documentation
  - Written the DDL, DML and the non-spatial queries
  - Made good progress on your 500 word assignment
- You will refine the above in the next 2 weeks to add spatial information (this and next week) and 3D (next week) after which you can complete the assignment

Overview

- What is spatial data
  - Georeferencing
- Modelling spatial data in a database
- Storing spatial data in PostgreSQL/PostGIS
  - DDL – adding a spatial column
  - DML – inserting data
- Visualising the Data

Spatial Data

- CEGE0052 is a *spatial* databases module
- In this case *spatial* refers to any data that can be located somewhere on the earth's surface (or above or below the surface)

** See Week 1 slides for more detail **

Spatial Data

- The ability to create “maps” using spatial data can be found in:
  - Geographical Information Systems (GIS)
    - E.g. QGIS, ArcMap
  - Building Information Modelling (BIM)
    - E.g. Revit, Bentley Architecture, ArchiCAD
- In both cases, the maps can be 2D or 3D

Spatial Data

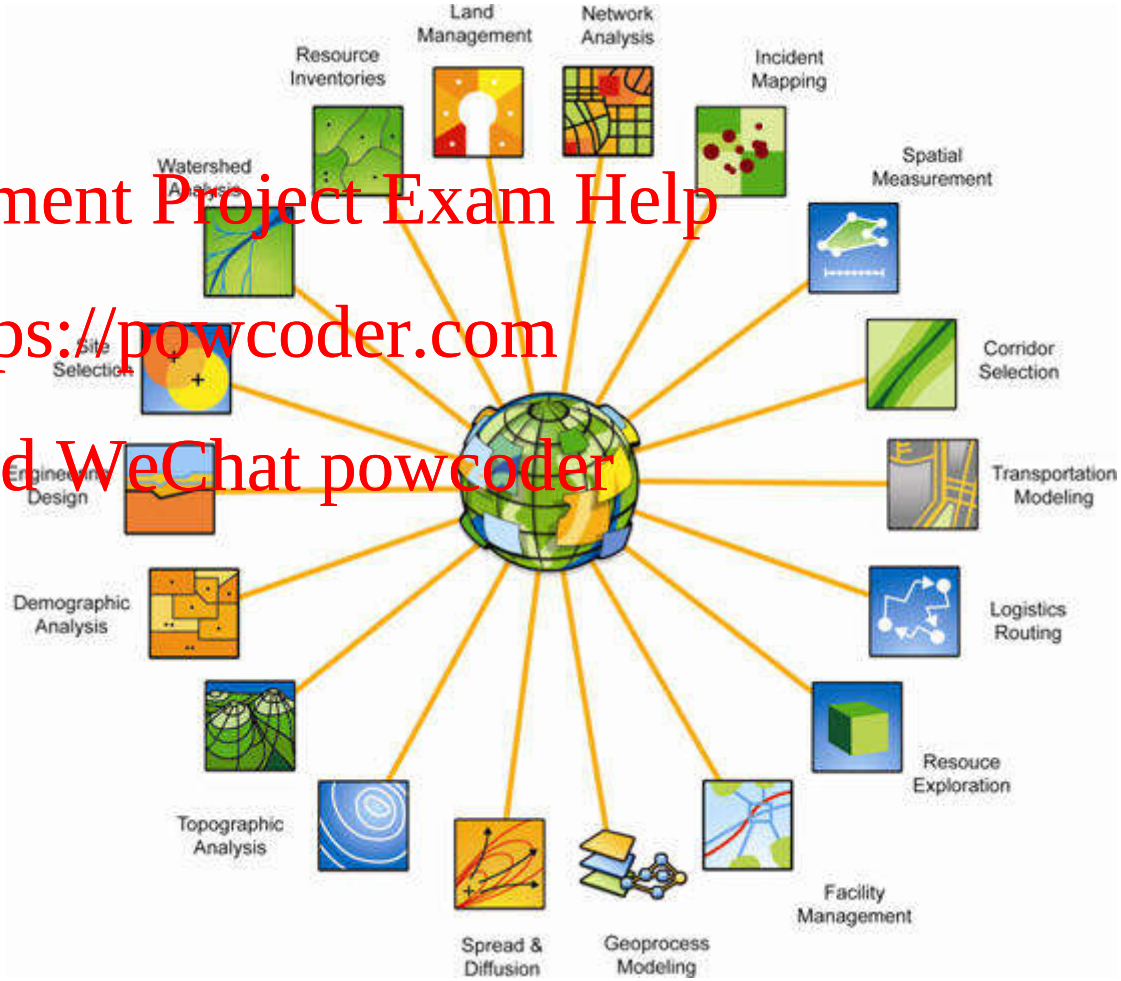
- We will be using GIS for mapping during this module as GIS software currently works best with databases
  - GIS are also extensively used in Asset Management
- … BIM software is slowly becoming better at working with databases but isn't quite there yet ..

Everything Happens Somewhere

http://www.iotphils.com/wp-content/uploads/2014/07/Smart-Cities.png

Spatial Data

- Used to understanding WHERE and WHEN things happen
  - And thus solve problems and provide useful services to people



http://www.directionsmag.com/images/newsletter/2011/01_week1/world_lg.jpg

Spatial Data

- How do we model the world using spatial data?
  - Using four types of geometric representation
    - Point
    - Line
    - Polygon
    - Polyhedron (3D)
  - (Also other types of representations e.g. for continuous surfaces – not part of this module)

Spatial Data – Points and Lines

- Spatial Types:
  - Points (also called nodes) are used for single point objects such as a well or a street light or traffic lights or – depending on the scale – a city or even a country.
    - Properties of a point include its location and its centroid (geometric centre)
  - Lines (also called polylines, arcs, edges) are used for interlinked objects such as rivers, water pipes or roads or for objects that appear linear from the air e.g. walls, fences, hedges.
    - Properties of a line include location, length, centroid and end-points.

Spatial Data – Polygons and Polyhedra

- Spatial Types:
  - Polygons are used for objects having an actual area – e.g. buildings, parks, gardens. Polygons are also used for administrative boundaries (counties, city boundaries, school catchment areas, country boundaries).
    - Polygons have associated area, perimeter and centroid measures and are two-dimensional.
  - Polyhedra (or volumes) are three dimensional objects and provide the most realistic representation of real-world objects (e.g. buildings, geological rock strata).
    - They have associated measures of surface area and volume (the measure of the total enclosed space). Polyhedra are three-dimensional.

Grouping Spatial Data

- Spatial data is grouped into themes (often known as layers)
  - A layer in a GIS is a way of collecting all the information relating to a particular object type type into one group.
    - E.g. – Rivers, Countries, Buildings, roads, rubbish bins, noise measurements
  - A layer can have any name you like
  - Layers are ‘stacked’ in the map to show all the data in one place
    - Maps are usually 2D but 3D is emerging (see later on in this module)

Grouping Spatial Data

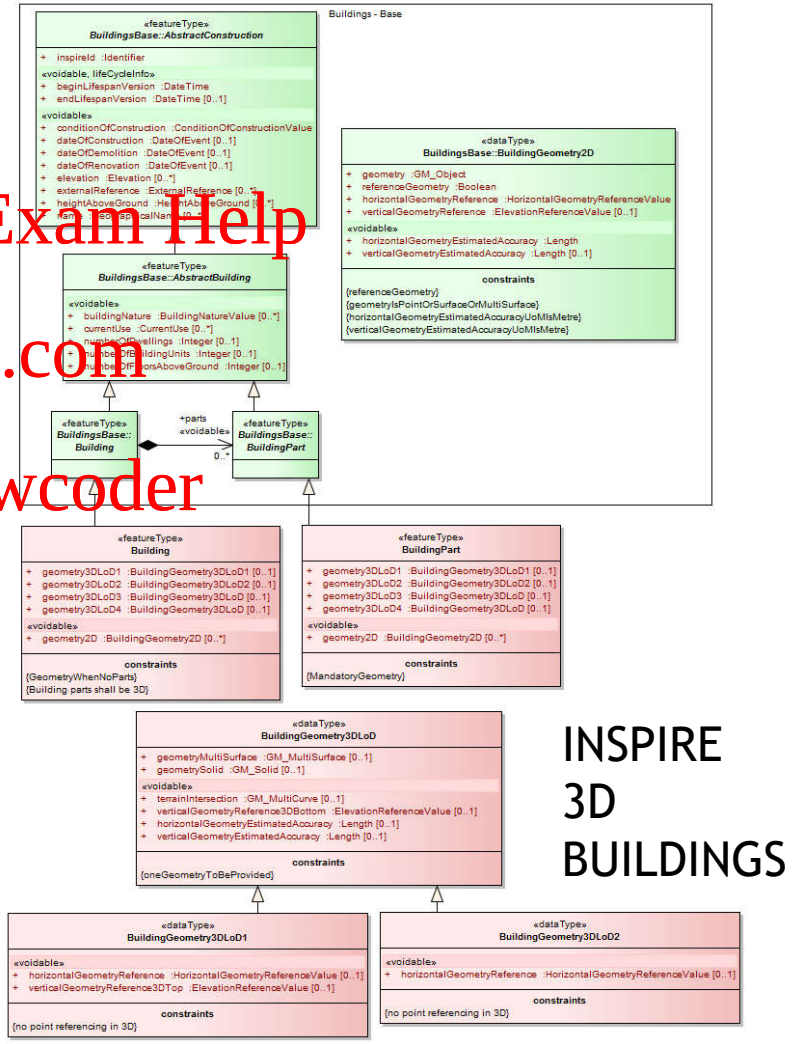
- For the purposes of this module:
  - a layer is a table in the database that has a spatial column
    - So the entities in your ERD become layers of spatial data if they are entities that can be mapped
  - see later on in this lecture for more information about making entities mappable using spatial columns

Grouping Spatial Data

- In theory, in GIS you can name your entities anything you like and structure them how you like
- However, if you want to share data, you probably want to use a standard
  - Standards tell you exactly what to model
  - (For your assignment you should NOT use any standards – it needs to be your own work)

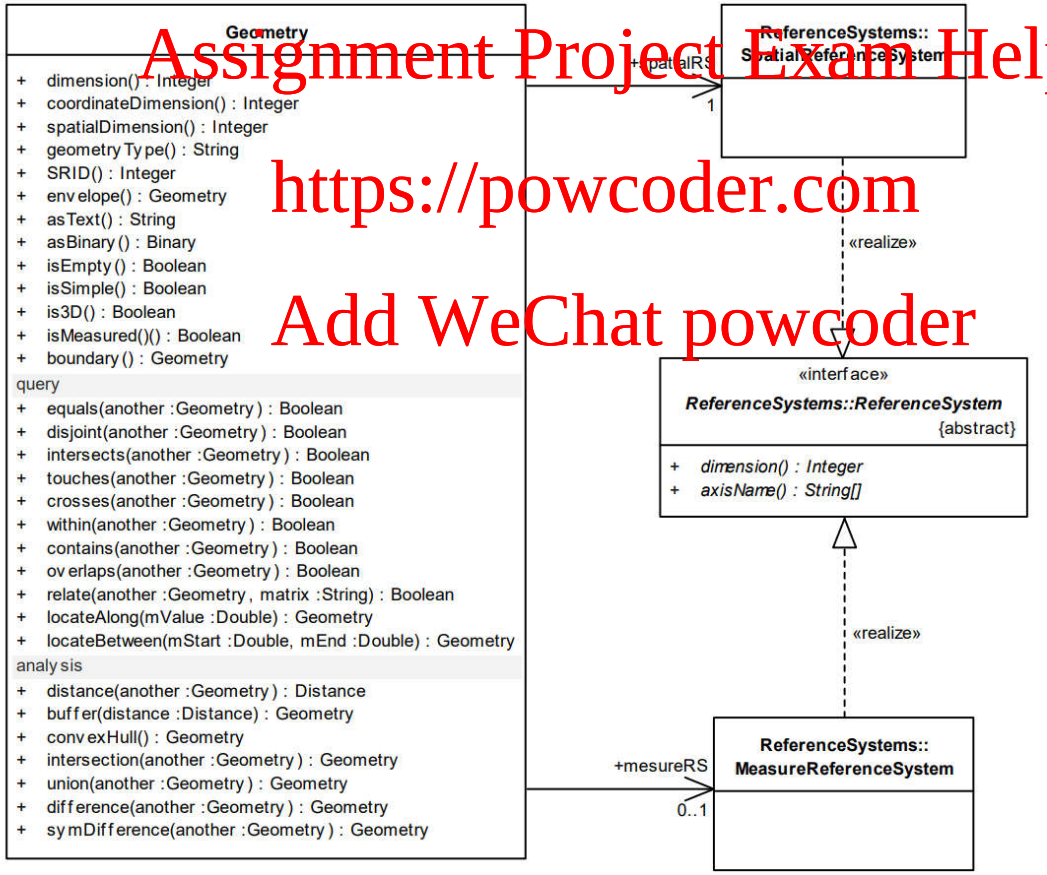
Grouping Spatial Data

- Standards
  - An EU directive called INSPIRE has created 100s of standards for spatial data sharing
  - We may also see CityGML, for 3D data sharing, as part of the advanced topics work



https://inspire.ec.europa.eu/data-model/approved/r4618-ir/html/index.htm?goto=2:2:1:3:76938

OGC Standard – Simple Features



http://www.opengeospatial.org/standards/sfa

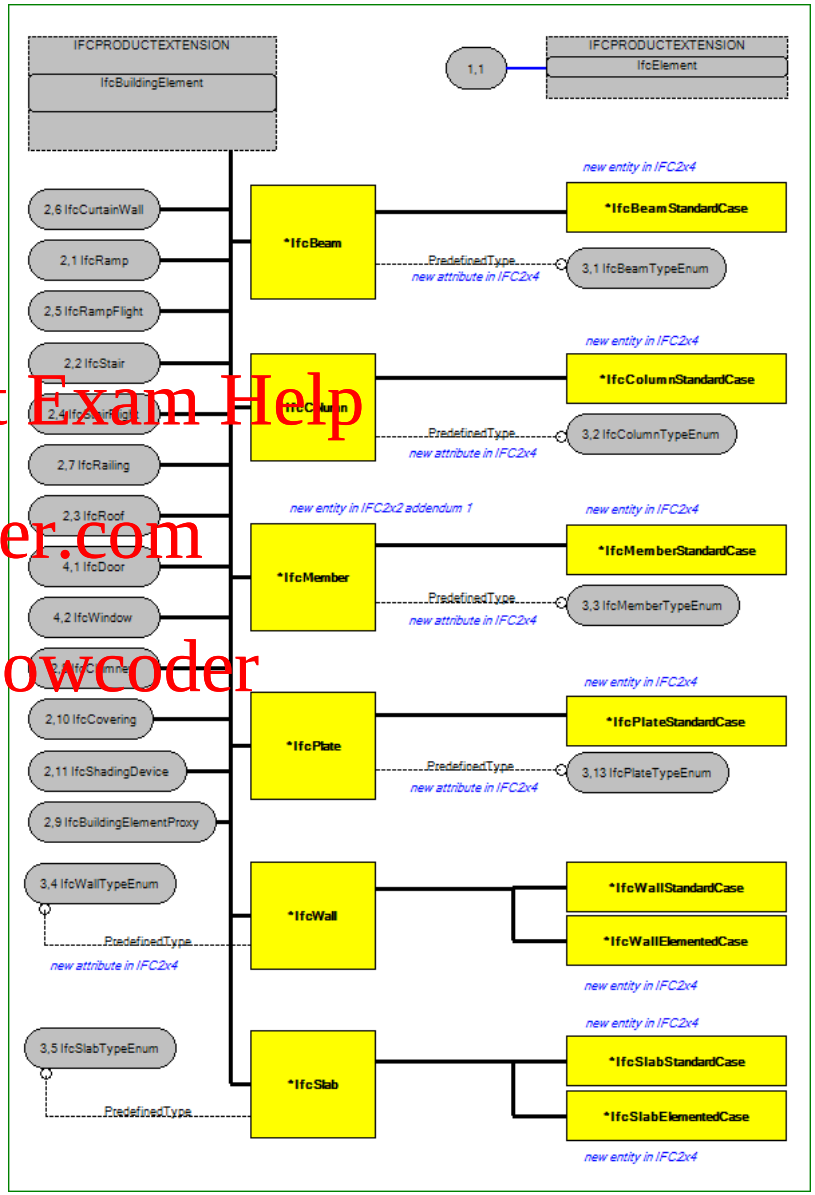
Grouping Spatial Data - BIM

- In BIM objects are always represented in 3D
  - Or at least that is the aim of Level 2 and Level 3 BIM
- In BIM information is grouped by construction object type
  - E.g. concrete slab, window, door, wall, duct

- (For information only, not required for your assignment)

Grouping Spatial Data - BIM

- In BIM the entity names are defined through a standard called Industry Foundation Classes



- (For information only, not required for your assignment)

Spatial Data

- Most important thing for this module:
  - You can store spatial data in the database just like any other type of data
  - When you map the data, you don't only get the points/lines/polygons/polyhedral
    - YOU ALSO GET THE OTHER INFORMATION (attributes/columns) FOR THAT DATA

Overview

- What is spatial data
  - **Georeferencing**
- Modelling spatial data in a database
- Storing spatial data in PostgreSQL/PostGIS
  - DDL – adding a spatial column
  - DML – inserting data
- Visualising the Data

Spatial Data

- Spatial data includes ***anything*** that can be modelled using some form of *location* information!
  - i.e. where something is, referenced to a shared framework (could be a coordinate system, a map of London Boroughs, countries of the world, UK counties and many more)
- This referencing is called *geo-referencing*

Geo-Referencing

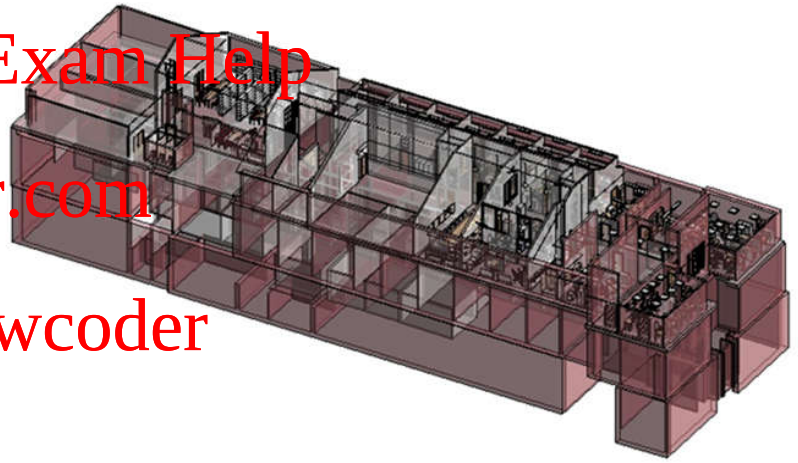
- Can be direct:
  - E.g. a map that shows a building or another object, x/y coordinates, GPS coordinates
- Or indirect
  - For example, a Post Code or a Street Address is an indirect geo-reference that can be used to link non-spatial data to a position on the map. A PDF file containing the specification of a water pipe can be linked to the location of that pipe.
- See later on in the module for more details about georeferencing

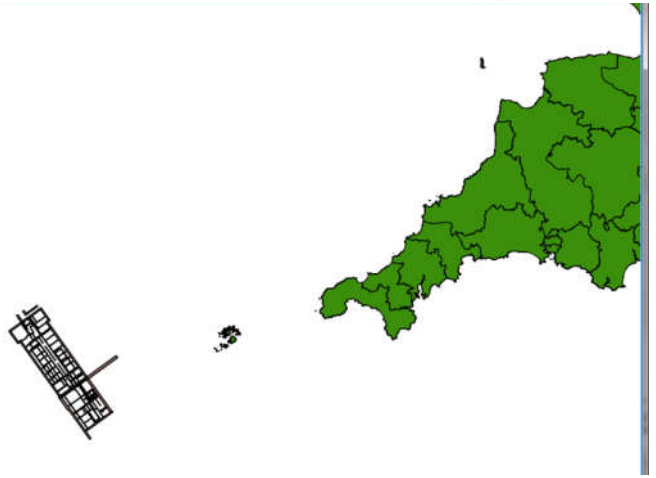
Direct - Coordinate Systems

- Direct referencing works by mapping objects using their real coordinates (e.g. the coordinates that a GPS captures)
- Depending on where you are in the world, and what system you are using these coordinates may be referenced to different ‘origin’ points ..

Direct – Local Coordinate Systems

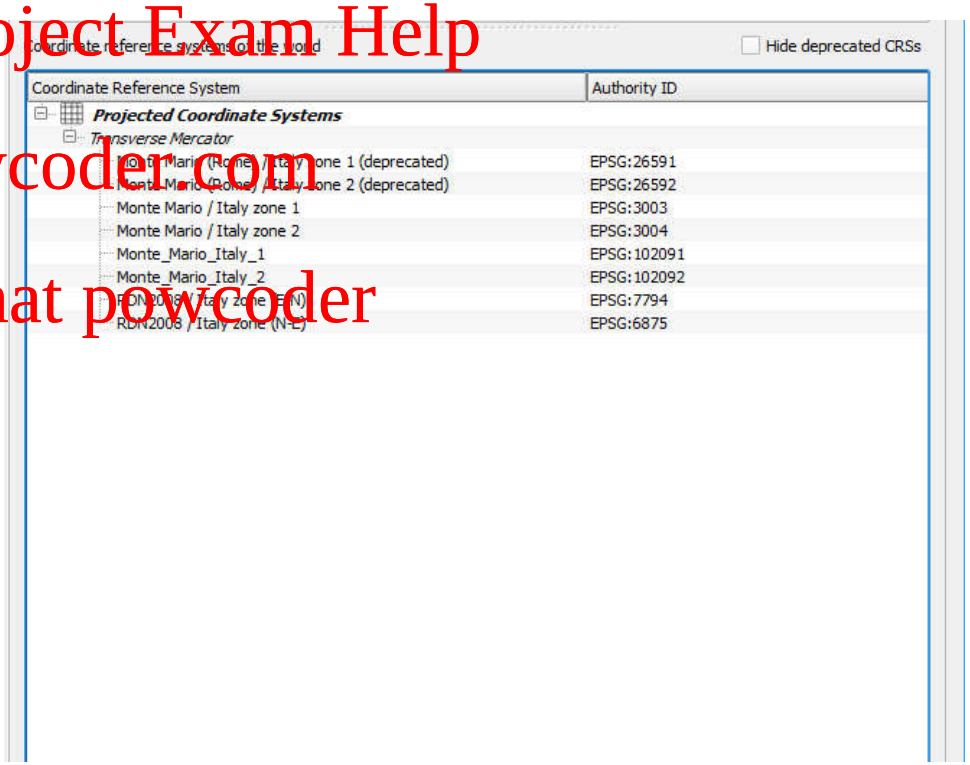
- Used in CAD/BIM
- Have a local reference point as the 0,0 point
  - Usually the edge of a construction site
- All distances and angles measured from this local reference point
- Also Cartesian (flat surface)





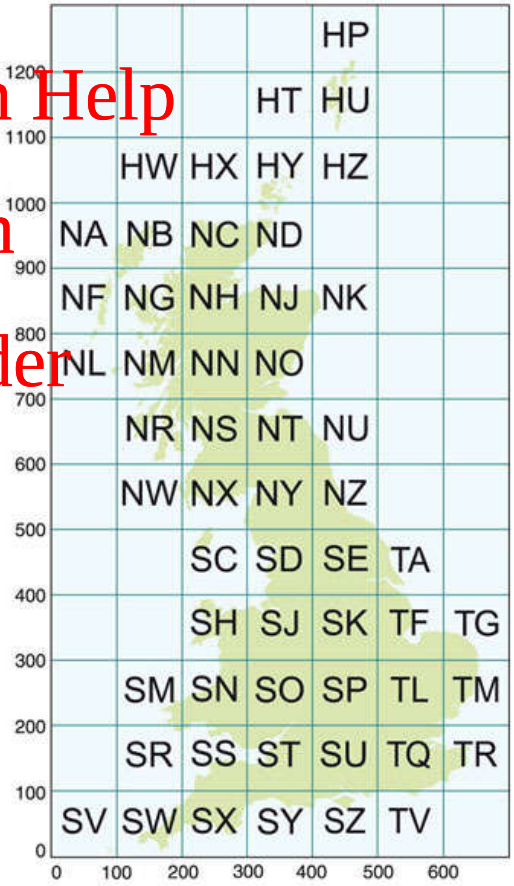
Direct – National Coordinate Systems

- National coordinate systems are usually created by an organisation such as a National Mapping Agency
- Some countries have more than one



Direct – National Coordinate Systems

- In Great Britain, our mapping system uses "British National Grid" which has its 0,0 off the south west coast



- Cartesian system

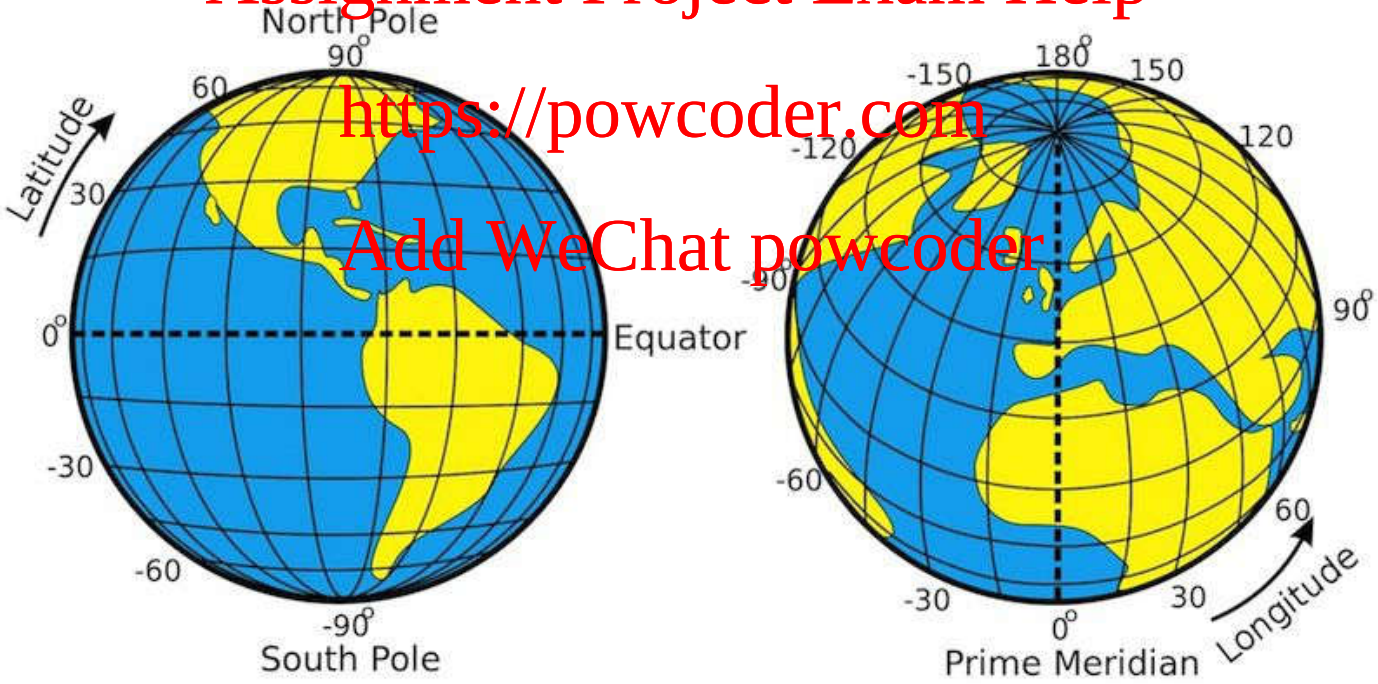
https://timetrail.warwickshire.gov.uk/assets/exhibition/Image/Toolkit/gb-map.jpg

Direct – Global Coordinate Systems

- As satellite systems such as GPS don't only map Great Britain, they use a reference system that covers the world
  - Coordinates are latitude/longitude
    - Longitude ranges from 0 at the Prime Meridian passing through Greenwich, England, to +180 toward the east and 0 to -180 toward the west.
    - Latitude ranges from 0 at the equator to +90 at the North Pole and 0 to -90 at the South Pole. For example, Denver's position shown in the figure is -104.9 degrees longitude (west) and +39.8 degrees latitude (north).

http://www.innovativegis.com/basis/pfprimer/topic7/topic7.html

Direct – Global Coordinate Systems



https://thumbs-prod.si-cdn.com/0rQSHAWkucV1O0dmyJVN8Ml0sS4=/800x600/filters:no_upscale()/https://public-media.smithsonianmag.com/filer/5c/ea/5cea567c-050b-432a-834f-fc94dcb1b49e/coordinates.jpg

Coordinate Reference Systems – Standard Codes

- Local coordinate reference systems are not set by any authority but are just defined by whoever is working on a project
- However, national and international systems are public and are assigned a code by the European Petroleum Standards Group
  - This is called an EPSG code
- In the UK
  - EPSG 27700 – British National Grid
  - EPSG 4326 – the WGS84 system used by GPS

Coordinate Reference Systems – Standard Codes

Coordinate Reference Systems – Linking Local and National Data

- If you have a locally referenced dataset, you can transform the data to a national reference system
- At a very basic level, if you know the real world x/y of one point (e.g. a corner of the building) in national units, then you can use this to shift all the coordinates
  - Tools such as Revit (for BIM) allow you to do this
  - Might also need to change the units from mm to m
- More sophisticated methods also exist (the geospatial students might learn some of these over the coming year)

Overview

- What is spatial data
  - Georeferencing
- **Modelling spatial data in a database**
- Storing spatial data in PostgreSQL/PostGIS
  - DDL – adding a spatial column
  - DML – inserting data
- Visualising the Data

Databases – Storing Spatial Data

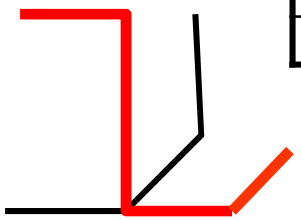
- Data Types
  - So far, we have seen the following data types in our databases:
    - VARCHAR
    - NUMBER (m, n)
    - DATE
  - Most modern Databases also have a special data type for storing vector spatial data
  - In Oracle, this data type is called SDO_GEOMETRY
  - SDO stands for Spatial Data Object
  - In PostGIS this is a GeometryColumn

Spatial Data – Basic Representation

- Points represented as two coordinates – x,y
  -

ID	X	Y	Attribute
1	100	**100**	**Well1**
2	400	**400**	**Well2**

- Lines are represented as a list of coordinate pairs

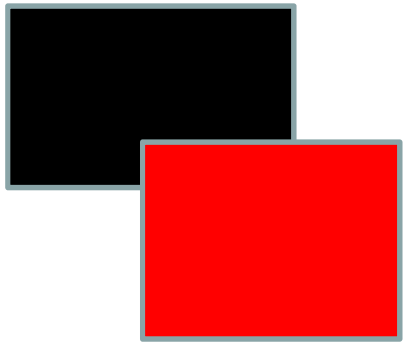


ID	X1	Y1	X2	Y2	X3	Y3	x4	Y4	X5	Y5	Attribute
1	125	150	325	150	450	375	450	500			**Road 1**
2	135	500	335	500	335	150	500	150	600	250	**Road 2**

How do you model lines with more than 5 nodes?

Spatial Data – Basic Representation

- Polygons are also represented as a series of x, y points – but the first and last point must be the same to close the loop

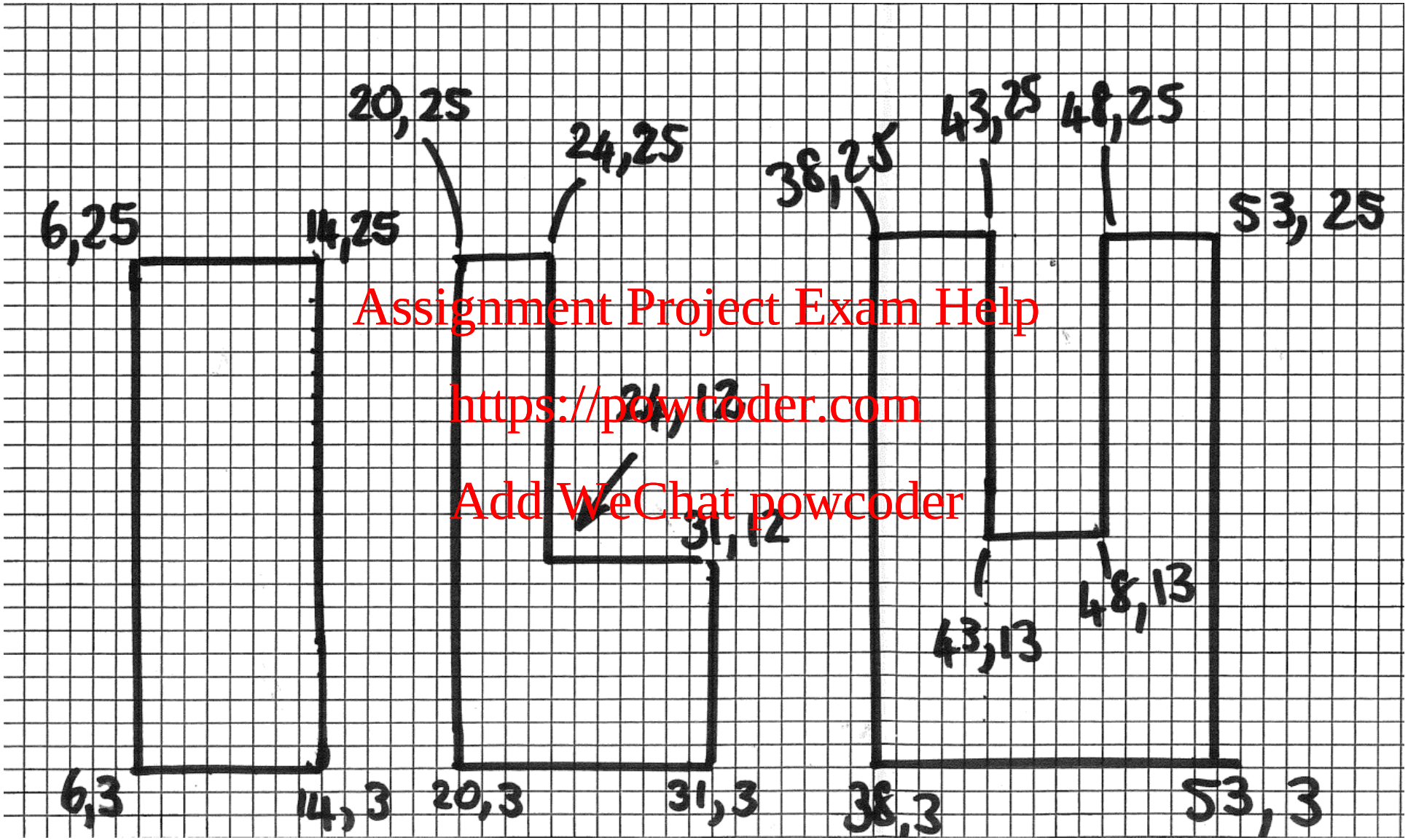


ID	X1	Y1	X2	Y2	X3	Y3	x4	Y4	X5	Y5	Attribute
1	125	150	300	150	300	450	125	450	125	150	House 1
2	225	50	450	50	450	200	225	200	225	50	House 2

How do you model polygons with more than 4 nodes (coordinate pairs)?

Exercise

- Draw the table that you would need to store these polygons in a database



Modelling more complex objects

- Option 1
  - Keep adding columns
    - But you could have 1000s of nodes!
    - Also could have lots of empty space
- Option 2
  - Use an ‘object relational’ approach – i.e. create a primitive type to store all the required information – this is called a ‘geometry’ data type

Modelling more complex objects

- Object-Relational Approach
  - Geometry is just another column in the table

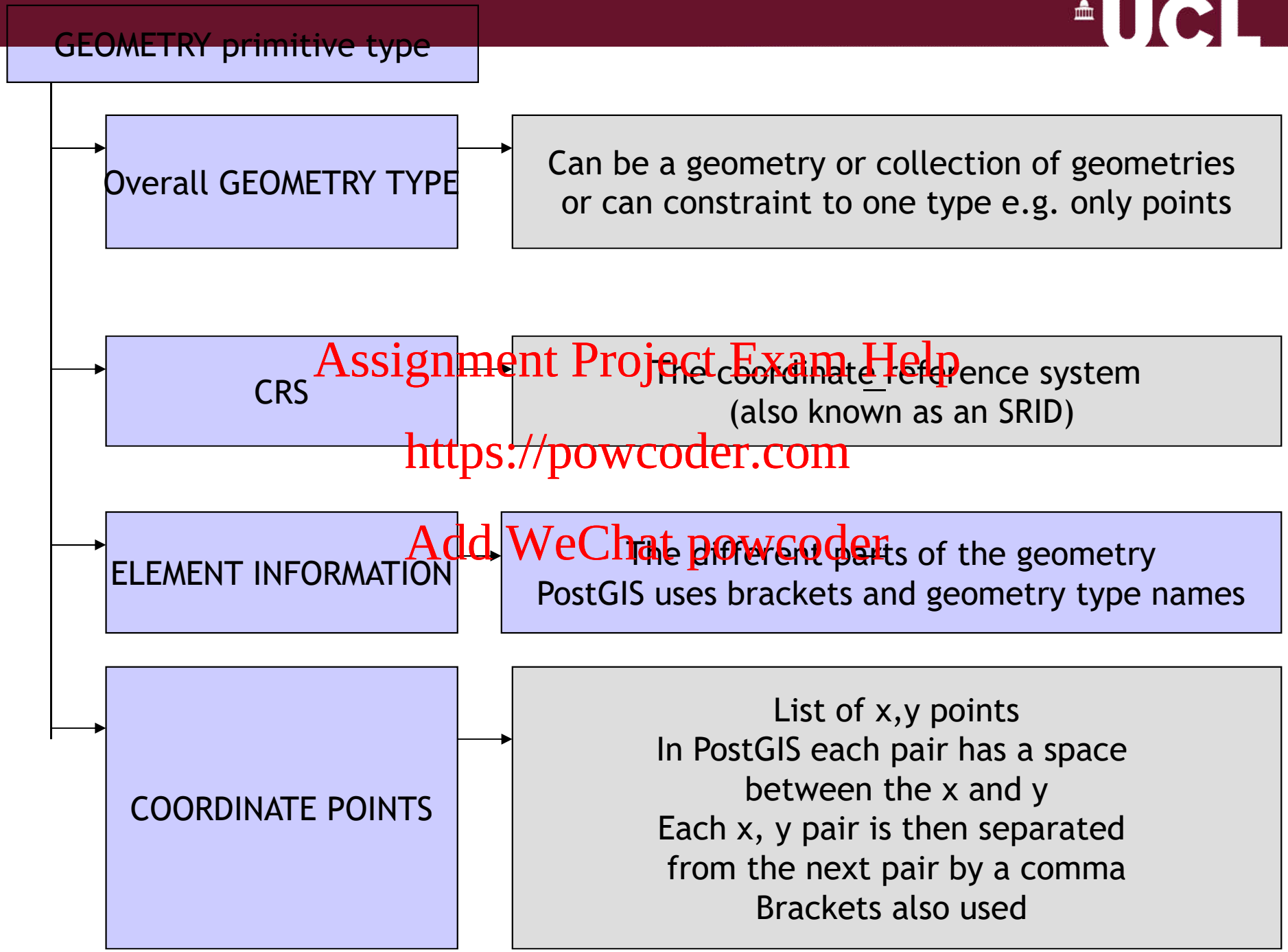
Attribute	Attribute	Attribute	Geometry
			<< all the geometry information stored in one column as a single object >>

- (Some additional information usually stored separately)

Databases – Storing Spatial Data

- Knowing the x,y values is not enough!
- What do the following numbers represent? Draw as many possible representations of these numbers (single points, lines, polygons, combinations of these etc).

2,4, 4,3, 10,3, 13,5, 13,9, 11,13, 5,13, 2,11, 2,4, 7,5, 7,10, 10,10, 10,5, 7,5



Overview

- What is spatial data
  - Georeferencing
- Modelling spatial data in a database
- Storing spatial data in PostgreSQL/PostGIS
  - **DDL - adding a spatial column**
  - DML – inserting data

Adding a Geometry (Spatial) Column

- Create some test tables first


```
create table assetsclass.london_poi(id serial);

create table assetsclass.london_highway(id serial);

create table assetsclass.london_counties(id serial);
```

Adding a Geometry (Spatial) Column

- Generic column can take any geometry type


```
select AddGeometryColumn(<<schema>>,<<tablename>>,<<column name>>,SRID, <<type of geometry>>,<<number of dimensions>>);

alter table assetsclass.building drop column if exists location;

select AddGeometryColumn('assetsclass','buildings','location',0, 'geometry',3);
```

Adding a Geometry (Spatial) Column

- Column for a specific geometry type
  - This is a form of constraint


```
alter table assetsclass.rooms drop column location;

select AddGeometryColumn('assetsclass','rooms','location',0, 'polygon',2);
--          (2 dimensions, polygons)

alter table assetsclass.buildings drop column if exists location;

select AddGeometryColumn('assetsclass','buildings','location',0,
'polyhedralsurface',3);
            (3 dimensions, polyhedral surfaces)
```

Adding a Geometry (Spatial) Column

- Coordinate reference systems and constraints - local reference system


```
alter table assetsclass.buildings drop column if exists location;

select AddGeometryColumn('assetsclass','buildings','location',0,
'polyhedralsurface',3);
```

Adding a Geometry (Spatial) Column

- Coordinate reference systems and constraints – British National Grid

```
alter table assetsclass.london_counties drop column if exists
location;


select AddGeometryColumn
('assetsclass','london_counties','location',27700, 'polygon',2); --
British National Grid
```

Adding a Geometry (Spatial) Column

- Coordinate reference systems and constraints – WGS84 (world wide)


```
alter table assetsclass.london_highway drop column if exists location;
```



```
select AddGeometryColumn
(‘assetsclass','‘london_highway','location',4326, ‘linestring',2);
```

```
alter table assetsclass.london_highway drop column if exists location;
Select addGeometryColumn
(‘assetsclass','‘london_poi','location',4326, ‘point',2);
```

Overview

- What is spatial data
  - Georeferencing
- Modelling spatial data in a database
- Storing spatial data in PostgreSQL/PostGIS
  - DDL – adding a spatial column
  - **DML – inserting data**
- Visualising the Data

Storing Spatial Data - PostGIS

- Well-Known Text
  - WKT is a human-readable format for representing geometry, and is therefore often used when populating databases using SQL.
  - Used by PostGIS for spatial data creation
  - Readable to the human eye
  - Not very compact

Storing Spatial Data - Storing Spatial Data - PostGIS

- Well-Known Text
  - POINT(0 0)
  - LINESTRING(0 0,1 1,1 2)
  - POLYGON((0 0,4 0,4 4,0 4,0 0),(1 1, 2 1, 2 2, 1 2,1 1))

Storing Spatial Data - Storing Spatial Data - PostGIS

- Well-Known Text
  - MULTIPOINT(0 0,1 2)
  - MULTILINESTRING((0 0,1 1,1 2),(2 3,3 2,5 4))
  - MULTIPOLYGON(((0 0,4 0,4 4,0 4,0 0),(1 1,2 1,2 2,1 2,1 1)), ((-1 -1,-1 -2,-2 -2,-2 -1,-1 -1)))
  - GEOMETRYCOLLECTION(POINT(2 3),LINESTRING(2 3,3 4))

Storing Spatial Data - Storing Spatial Data - PostGIS

- Well-Known Binary

  - "The Well-known Binary Representation for Geometry (WKB Geometry) provides a portable representation of a geometric object as a contiguous stream of bytes.

  - It permits geometric object to be exchanged between an SQL/CLI client and an SQL-implementation in binary form" (OGC Simple Features Specification, 2006).

Storing Spatial Data - Storing Spatial Data - PostGIS

- Well-Known Binary
  - Unlike WKT, WKB is not readable to the human eye, and is a more compact format for storing geometry objects.
    - It is therefore used in particular for data exchange and transferring data between one platform and another. It makes use of Binary Large Objects inside the database to store the geometry as a stream of bytes.

Storing Spatial Data

- WKT and WKB
  - The WKT and WKB formats handle many of the items on the list of information required to be stored in a database to fully represent spatial data.
  - In particular, information describing how to use the coordinates (do they represent a point or set of points, a line, multiple disjoint lines, a simple polygon, a polygon with holes or multiple disjoint polygons, with or without holes) is present, as is information describing which coordinates in the list correspond to these individual object parts.

Storing Spatial Data

- WKT or WKB on their own are not enough
- You also need additional metadata to describe your spatial objects
- In PostGIS, the remaining information is stored separately from the main data, a metadata table (view) called *geometry_column*. This contains the following information about each spatial dataset:
  - schema_name - which user owns the dataset
  - table_name - the name of the spatially enabled table

Storing Spatial Data - Storing Spatial Data - PostGIS

- In PostGIS, the remaining information is stored separately from the main data, a metadata table called *geometry_columns*. This contains the following information about each spatial dataset:
  - column_name - the column containing the spatial information
  - srid - the spatial reference system (this cross-references a table called SPATIAL_REF_SYS, which contains information about over 3000 coordinate systems)
  - type- used to restrict the column to a single type of spatial object (point, line or polygon etc) if required
  - dimension - whether the data is 2 or 3D.

Data Output | Explain | Messages | Notifications | Query History

	srid [PK] integer	auth_name character varying (256)	auth_srid integer	srtext character varying (2048)	proj4text character varying (2048)
1	2000	EPSG	2000	PROJCS["Anguilla 1957 / British West Indies Grid",GEOGCS["Anguilla...	+proj=tmerc +lat_0=0 +lon_0=-62 +k=0.9995000000000001 +x_0=400000 +y_0=0 +ellps=clrk80 +units=m +no_defs
2	2001	EPSG	2001	PROJCS["Antigua 1943 / British West Indies Grid",GEOGCS["Antigua ...	+proj=tmerc +lat_0=0 +lon_0=-62 +k=0.9995000000000001 +x_0=400000 +y_0=0 +ellps=clrk80 +towgs84=-255,-15,71,0,0,0,0
3	2002	EPSG	2002	PROJCS["Dominica 1945 / British West Indies Grid",GEOGCS["Domin...	+proj=tmerc +lat_0=0 +lon_0=-62 +k=0.9995000000000001 +x_0=400000 +y_0=0 +ellps=clrk80 +towgs84=725,685,536,0,0,0,0
4	2003	EPSG	2003	PROJCS["Grenada 1953 / British West Indies Grid",GEOGCS["Grenad...	+proj=tmerc +lat_0=0 +lon_0=-62 +k=0.9995000000000001 +x_0=400000 +y_0=0 +ellps=clrk80 +towgs84=72,213.7,93,0,0,0,0
5	2004	EPSG	2004	PROJCS["Montserrat 1958 / British West Indies Grid",GEOGCS["Mon...	+proj=tmerc +lat_0=0 +lon_0=-62 +k=0.9995000000000001 +x_0=400000 +y_0=0 +ellps=clrk80 +towgs84=174,359,365,0,0,0,0
6	2005	EPSG	2005	PROJCS["St. Kitts 1955 / British West Indies Grid",GEOGCS["St. Kitts ...	+proj=tmerc +lat_0=0 +lon_0=-62 +k=0.9995000000000001 +x_0=400000 +y_0=0 +ellps=clrk80 +towgs84=9,183,236,0,0,0,0 +u
7	2006	EPSG	2006	PROJCS["St. Lucia 1955 / British West Indies Grid",GEOGCS["St. Luci...	+proj=tmerc +lat_0=0 +lon_0=-62 +k=0.9995000000000001 +x_0=400000 +y_0=0 +ellps=clrk80 +towgs84=-149,128,296,0,0,0,0
8	2007	EPSG	2007	PROJCS["St. Vincent 45 / British West Indies Grid",GEOGCS["St. Vinc...	+proj=tmerc +lat_0=0 +lon_0=-62 +k=0.9995000000000001 +x_0=400000 +y_0=0 +ellps=clrk80 +towgs84=195.671,332.517,27
9	2008	EPSG	2008	PROJCS["NAD27(CGQ77) / SCoPQ zone 2 (deprecated)",GEOGCS["N...	+proj=tmerc +lat_0=0 +lon_0=-55.5 +k=0.9999 +x_0=304800 +y_0=0 +ellps=clrk66 +units=m +no_defs
10	2009	EPSG	2009	PROJCS["NAD27(CGQ77) / SCoPQ zone 3",GEOGCS["NAD27(CGQ77)...	+proj=tmerc +lat_0=0 +lon_0=-58.5 +k=0.9999 +x_0=304800 +y_0=0 +ellps=clrk66 +units=m +no_defs
11	2010	EPSG	2010	PROJCS["NAD27(CGQ77) / SCoPQ zone 4",GEOGCS["NAD27(CGQ77)...	+proj=tmerc +lat_0=0 +lon_0=-61.5 +k=0.9999 +x_0=304800 +y_0=0 +ellps=clrk66 +units=m +no_defs
12	2011	EPSG	2011	PROJCS["NAD27(CGQ77) / SCoPQ zone 5",GEOGCS["NAD27(CGQ77)...	+proj=tmerc +lat_0=0 +lon_0=-64.5 +k=0.9999 +x_0=304800 +y_0=0 +ellps=clrk66 +units=m +no_defs
13	2012	EPSG	2012	PROJCS["NAD27(CGQ77) / SCoPQ zone 6",GEOGCS["NAD27(CGQ77)...	+proj=tmerc +lat_0=0 +lon_0=-67.5 +k=0.9999 +x_0=304800 +y_0=0 +ellps=clrk66 +units=m +no_defs
14	2013	EPSG	2013	PROJCS["NAD27(CGQ77) / SCoPQ zone 7",GEOGCS["NAD27(CGQ77)...	+proj=tmerc +lat_0=0 +lon_0=-70.5 +k=0.9999 +x_0=304800 +y_0=0 +ellps=clrk66 +units=m +no_defs

Data Output | Explain | Messages | Notifications | Query History

	f_table_catalog character varying (256)	f_table_schema name	f_table_name name	f_geometry_column name	coord_dimension integer	srid integer	type character varying (30)
1	user1db	public	united_kingdom...	geom	2	27700	MULTIPOLYGON
2	user1db	public	united_kingdom...	geom	2	4326	MULTILINESTRING
3	user1db	public	united_kingdom...	geom	2	4326	MULTIPOINT
4	user1db	public	formdata	geom	2	0	GEOMETRY

Inserting Spatial Data

- Geometry is treated the same as the other columns
  - Use ST_GEOMFROMTEXT and WKT
  - Note the quotation marks


```
insert into assetsclass.rooms (floor, room_number, building_id, location,
last_repainted, room_use)
values
(1, '1.01',(select building_id from assetsclass.buildings where building_name =
'Chadwick'),
st_geomfromtext('POLYGON((3 2, 8 2, 8 12, 3 12, 3 2))', 0),'12-Jan-
1950','classroom');
```

Inserting Spatial Data

- …. (note the quotation marks)

st_geomfromtext('POLYGON((3 2, 8 2, 8 12, 3 12, 3 2))', 0)

Inserting Spatial Data – National CRS


```
insert into assetsclass.london_counties (location)
values
(st_geomfromtext('POLYGON((328103 186492, 328108
186492, 328108 186502, 328103 186502, 328103 186492))',
27700));
```

Inserting Spatial Data – Global CRS


```
insert into assetsclass.london_poi (location)
values
(st_geomfromtext('POINT(-5.4233444 50.1876552)',4326));
```



```
insert into assetsclass.london_highway (location)
Values
(st_geomfromtext('LINESTRING((-4.1997314 50.4060347,-4.1998784
50.4061017,-4.1999345 50.406374,-4.2000396 50.4066819,-4.2002497
50.4069987,-4.2004388 50.4071015,-4.2006227 50.4070739))',4326);
```

Inserting Spatial Data

- Violating a constraint ..


```
insert into assetsclass.london_poi (location)
values
(st_geomfromtext('MULTIPOINT(-5.4233444
50.1876552)',4326));
```

-- Geometry type (MultiPoint) does not match column type (Point)

Inserting Spatial Data – Using an UPDATE statement


```
update assetsclass.buildings set location =
(st_geomfromtext('POLYHEDRALSURFACE(((480501.5
131048.4 0,480501.5 175767.7 0,543813.3 175767.7
0,543813.3 131048.4 0,480501.5 131048.4 0)))', 0)) where
building_id =(select building_id from assetsclass.buildings
where building_name = 'Chadwick');
```

Overview

- What is spatial data
  - Georeferencing
- Modelling spatial data in a database
- Storing spatial data in PostgreSQL/PostGIS
  - DDL – adding a spatial column
  - DML – inserting data
- **Visualising the Data**

Visualising the Data

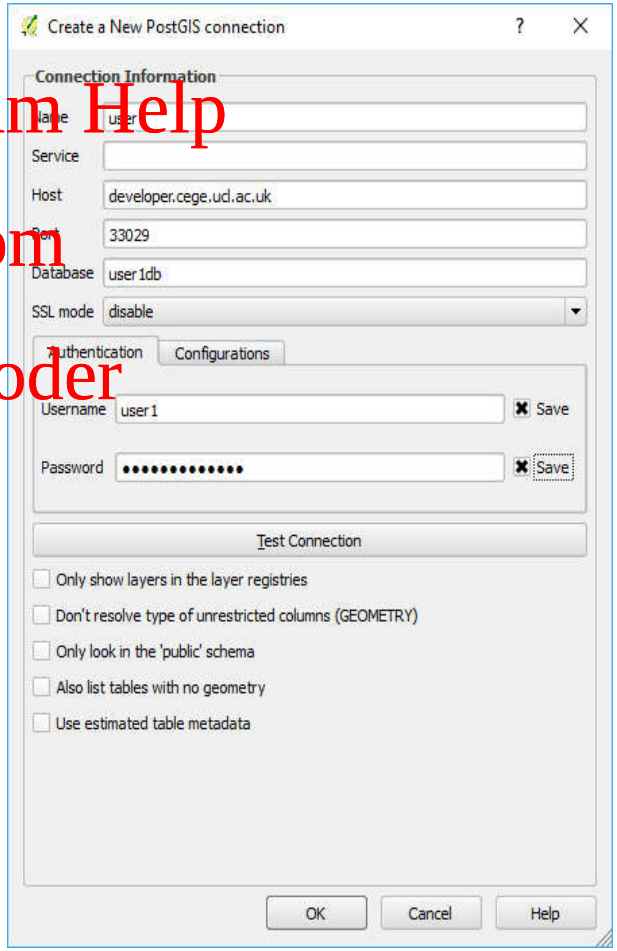
- As WKT (well known text) is a standard, then most GIS software packages can easily connect onto a database that uses this standard and visualise the data
  - As a reminder – GIS = geographical information system – the software that stores, edits, analyses and visualises spatial data
  - Map creation – and the spatial SQL we will see next week – is just a very small part of what a GIS can do

Visualising the Data

- For this module, we will be using two GIS software packages to visualise the data
  - QGIS (NB: version 2.18)
  - FME

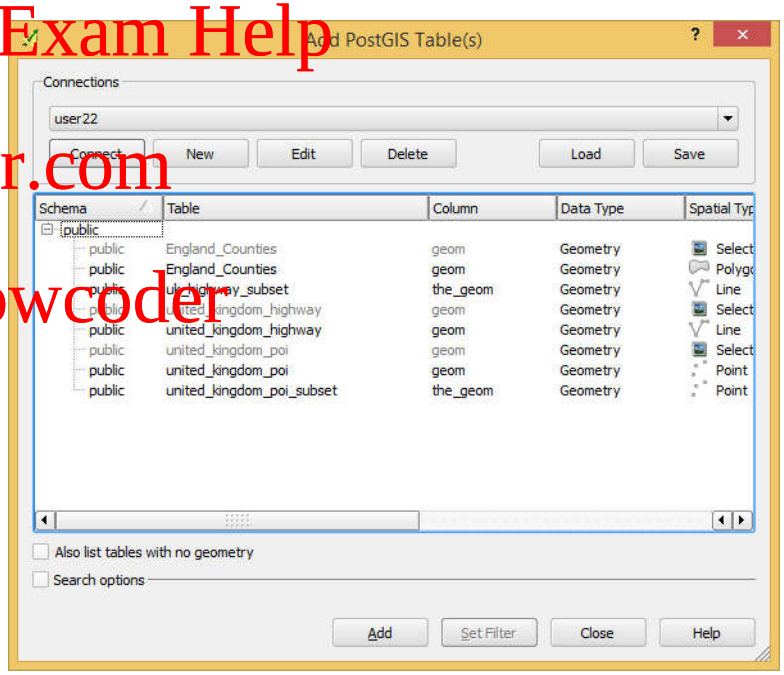
Step 1 – connect to the database

- Connect to the database (use your databases username and password for this)

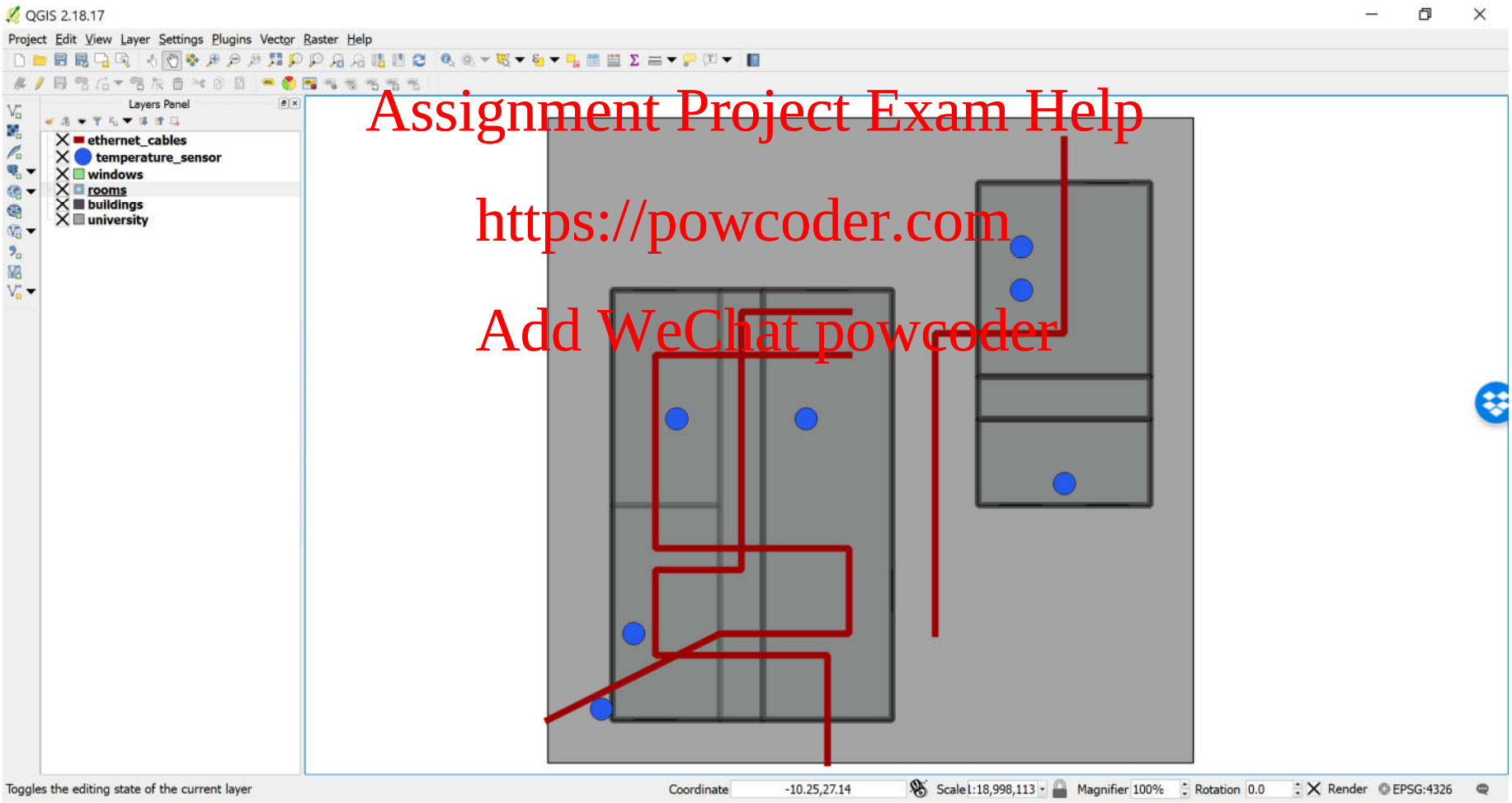


Step 2 – select the layer(s) you want to view

- From data that already exists in your database
- from the spatial tables you have created (once you have done this)



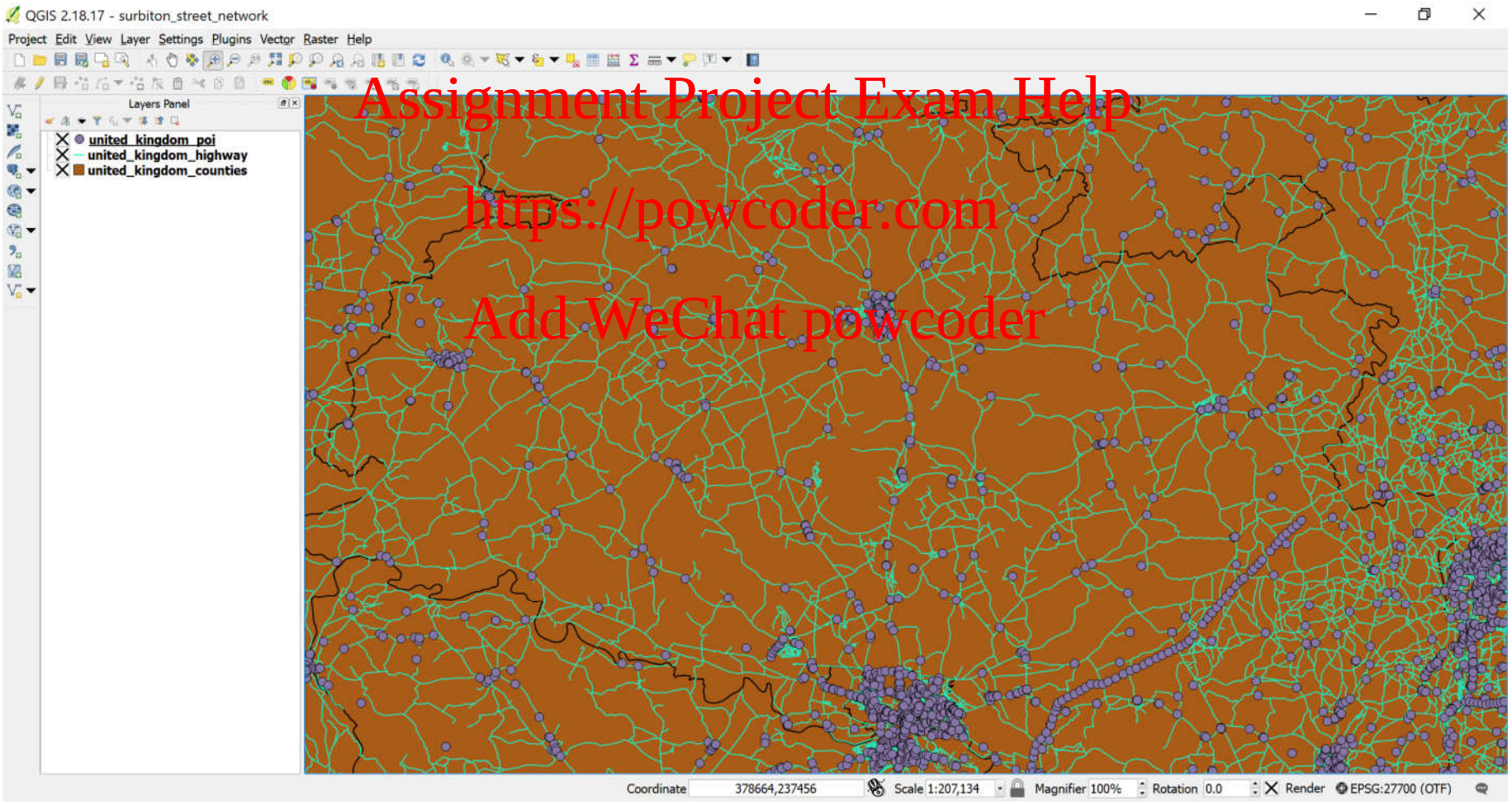
Step 3 – View the Data



Step 4 – map styling

- In a GIS you can – in theory – make your layers use any colours you like
  - Cartographers – map makers – do have general guidelines e.g. motorways are orange
  - Large mapping projects – e.g. HS2 or Crossrail – will have their own styles and templates
  - Some disciplines – e.g. space syntax – also have their own conventions

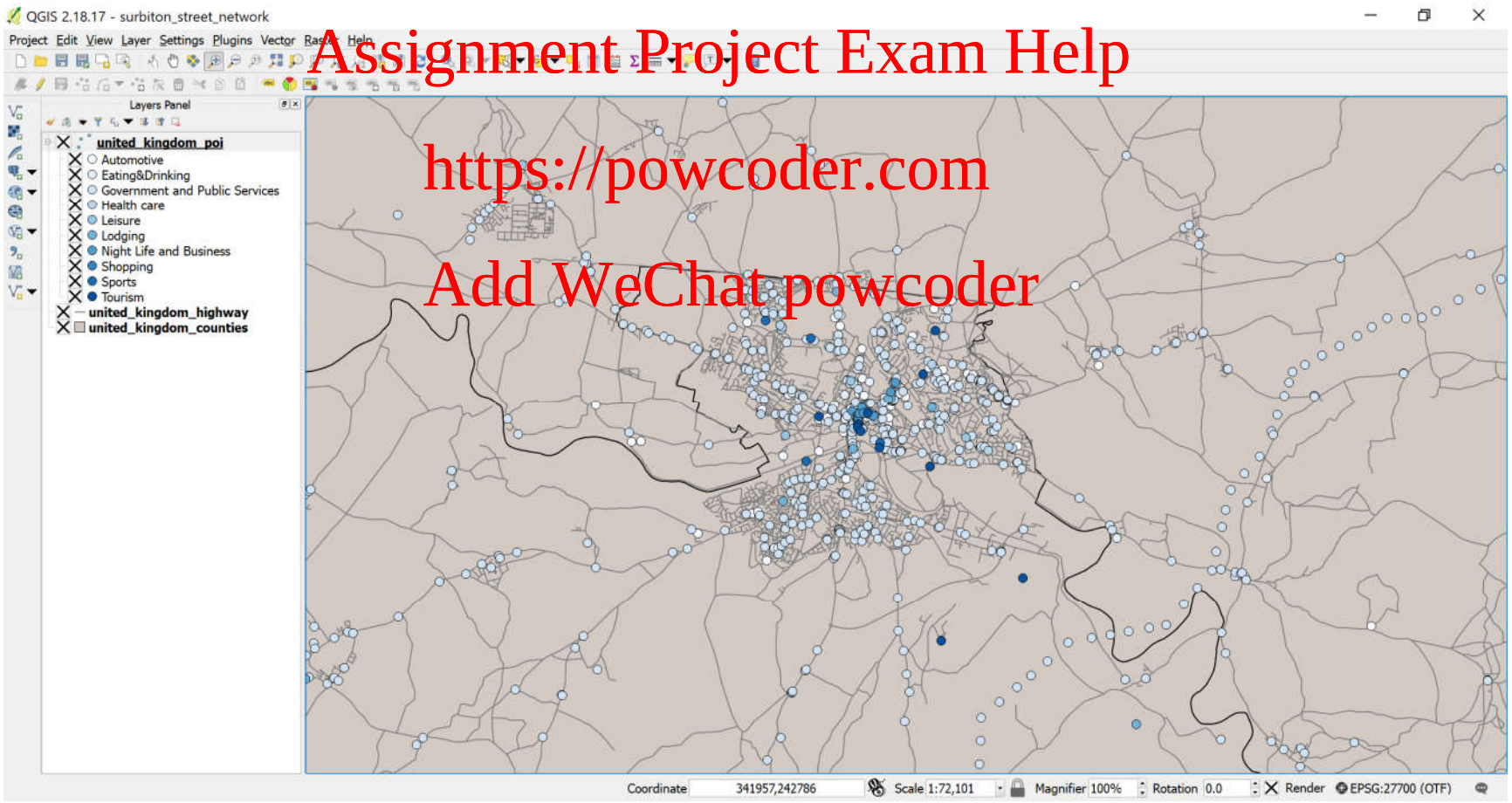
Step 4 – choose colours (random)



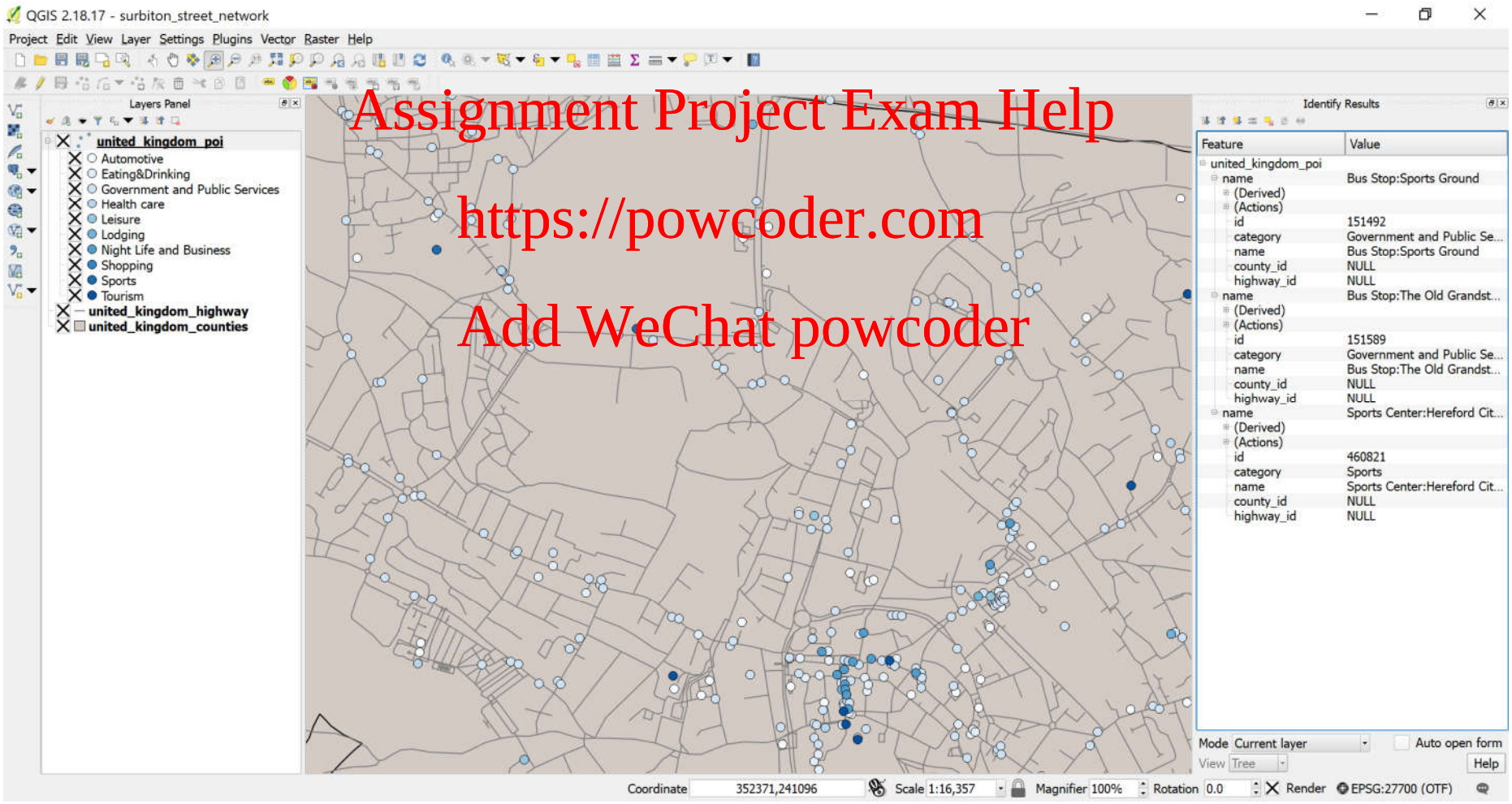
Map Styling

- http://proceedings.esri.com/library/userconf/fed16/papers/fed_86.pdf
- ColorBrewer provides some help on colour choices http://colorbrewer2.org/#type=sequential&scheme=BuGn&n=3
- Production maps also need a legend, scale bar and north arrow
  - (Geospatial students – you will learn about this in more detail)
- For your assignment – screenshots are sufficient, professional maps not required

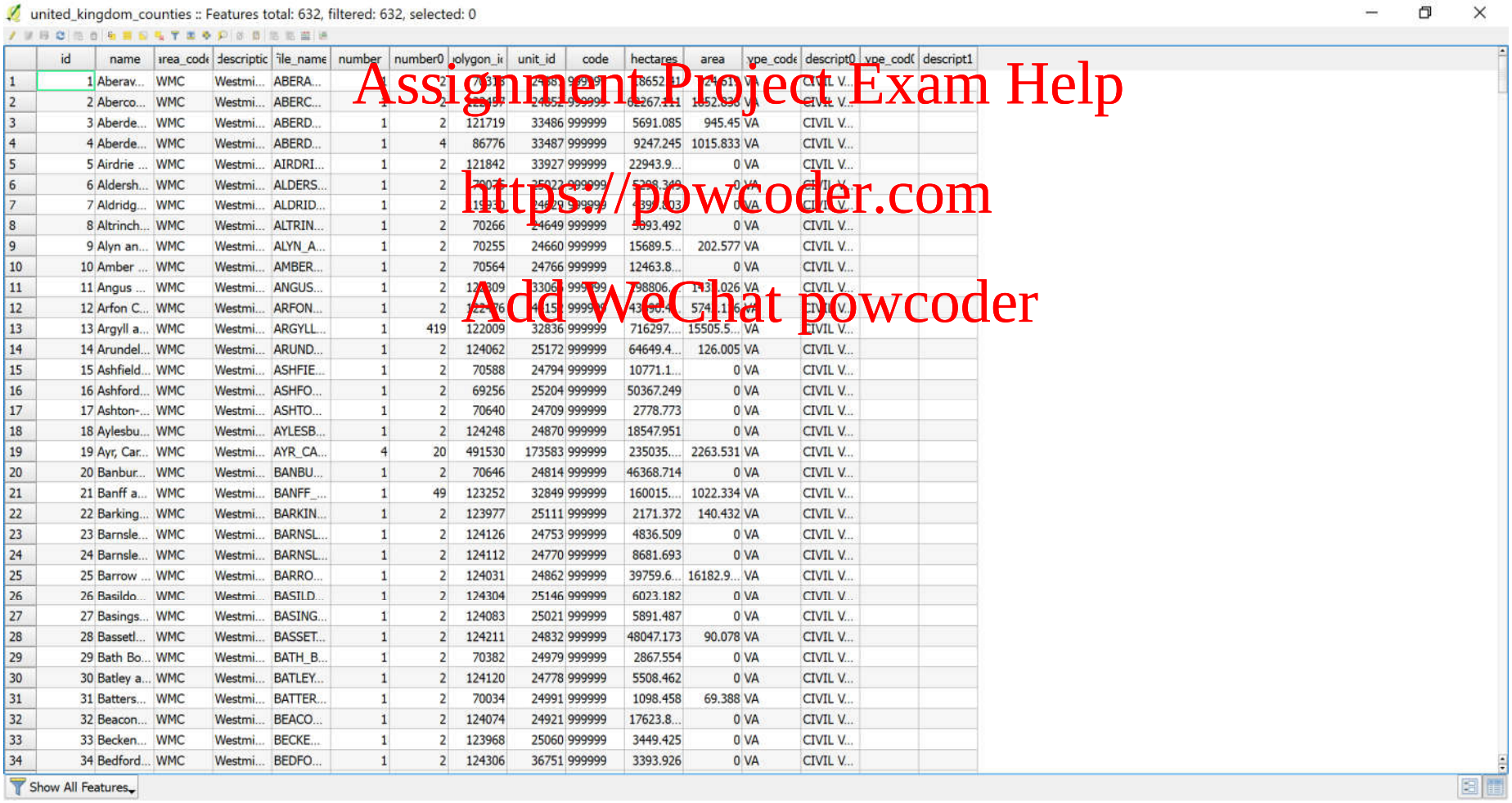
Step 4 – choose colours (by category)



Step 5 – see the other attributes



Step 5 – see the other attributes



	id	name	area_code	descriptic	file_name	number	number0	polygon_id	unit_id	code	hectares	area	type_code	descript0	type_cod	descript1
1	1	Aberav...	WMC	Westmi...	ABERA...	1	2	70316	25083	999999	18652.41	124.61	VA	CIVIL V...		
2	2	Aberco...	WMC	Westmi...	ABERC...	1	2	122497	24852	999999	62267.111	1652.838	VA	CIVIL V...		
3	3	Aberde...	WMC	Westmi...	ABERD...	1	2	121719	33486	999999	5691.085	945.45	VA	CIVIL V...		
4	4	Aberde...	WMC	Westmi...	ABERD...	1	4	86776	33487	999999	9247.245	1015.833	VA	CIVIL V...		
5	5	Airdrie ...	WMC	Westmi...	AIRDRI...	1	2	121842	33927	999999	22943.9...	0	VA	CIVIL V...		
6	6	Aldersh...	WMC	Westmi...	ALDERS...	1	2	70015	25021	999999	5288.349	0	VA	CIVIL V...		
7	7	Aldridg...	WMC	Westmi...	ALDRID...	1	2	119251	24829	999999	4397.103	0	VA	CIVIL V...		
8	8	Altrinch...	WMC	Westmi...	ALTRIN...	1	2	70266	24649	999999	5893.492	0	VA	CIVIL V...		
9	9	Alyn an...	WMC	Westmi...	ALYN_A...	1	2	70255	24660	999999	15689.5...	202.577	VA	CIVIL V...		
10	10	Amber ...	WMC	Westmi...	AMBER...	1	2	70564	24766	999999	12463.8...	0	VA	CIVIL V...		
11	11	Angus ...	WMC	Westmi...	ANGUS...	1	2	123309	33063	999999	798806....	143.026	VA	CIVIL V...		
12	12	Arfon C...	WMC	Westmi...	ARFON...	1	2	122476	24815	999999	43390.4...	574.116	VA	CIVIL V...		
13	13	Argyll a...	WMC	Westmi...	ARGYLL...	1	419	122009	32836	999999	716297....	15505.5...	VA	CIVIL V...		
14	14	Arundel...	WMC	Westmi...	ARUND...	1	2	124062	25172	999999	64649.4...	126.005	VA	CIVIL V...		
15	15	Ashfield...	WMC	Westmi...	ASHFIE...	1	2	70588	24794	999999	10771.1...	0	VA	CIVIL V...		
16	16	Ashford...	WMC	Westmi...	ASHFO...	1	2	69256	25204	999999	50367.249	0	VA	CIVIL V...		
17	17	Ashton-...	WMC	Westmi...	ASHTO...	1	2	70640	24709	999999	2778.773	0	VA	CIVIL V...		
18	18	Aylesbu...	WMC	Westmi...	AYLESB...	1	2	124248	24870	999999	18547.951	0	VA	CIVIL V...		
19	19	Ayr, Car...	WMC	Westmi...	AYR_CA...	4	20	491530	173583	999999	235035....	2263.531	VA	CIVIL V...		
20	20	Banbur...	WMC	Westmi...	BANBU...	1	2	70646	24814	999999	46368.714	0	VA	CIVIL V...		
21	21	Banff a...	WMC	Westmi...	BANFF_...	1	49	123252	32849	999999	160015....	1022.334	VA	CIVIL V...		
22	22	Barking...	WMC	Westmi...	BARKIN...	1	2	123977	25111	999999	2171.372	140.432	VA	CIVIL V...		
23	23	Barnsle...	WMC	Westmi...	BARNSL...	1	2	124126	24753	999999	4836.509	0	VA	CIVIL V...		
24	24	Barnsle...	WMC	Westmi...	BARNSL...	1	2	124112	24770	999999	8681.693	0	VA	CIVIL V...		
25	25	Barrow ...	WMC	Westmi...	BARRO...	1	2	124031	24862	999999	39759.6...	16182.9...	VA	CIVIL V...		
26	26	Basildo...	WMC	Westmi...	BASILD...	1	2	124304	25146	999999	6023.182	0	VA	CIVIL V...		
27	27	Basings...	WMC	Westmi...	BASING...	1	2	124083	25021	999999	5891.487	0	VA	CIVIL V...		
28	28	Bassetl...	WMC	Westmi...	BASSET...	1	2	124211	24832	999999	48047.173	90.078	VA	CIVIL V...		
29	29	Bath Bo...	WMC	Westmi...	BATH_B...	1	2	70382	24979	999999	2867.554	0	VA	CIVIL V...		
30	30	Batley a...	WMC	Westmi...	BATLEY...	1	2	124120	24778	999999	5508.462	0	VA	CIVIL V...		
31	31	Batters...	WMC	Westmi...	BATTER...	1	2	70034	24991	999999	1098.458	69.388	VA	CIVIL V...		
32	32	Beacon...	WMC	Westmi...	BEACO...	1	2	124074	24921	999999	17623.8...	0	VA	CIVIL V...		
33	33	Becken...	WMC	Westmi...	BECKE...	1	2	123968	25060	999999	3449.425	0	VA	CIVIL V...		
34	34	Bedford...	WMC	Westmi...	BEDFO...	1	2	124306	36751	999999	3393.926	0	VA	CIVIL V...		

Connecting via ArcMap

- For the MSc Geospatial Science, MSc Spatio-Temporal Analytics, MSc Civil Engineering with GIS…
  - You can also connect to PostGIS from R (apparently)
  - ArcMap – use ArcCatalog Database Connections

- (not required for your assignment)

https://www.reddit.com/r/dataisbeautiful/comments/9oxlct/a_mosaic_of_world_countries_retaining_their/
http://www.freeusandworldmaps.com/images/WorldProjections_Maps/WorldProjectionCountriesColorA.jpg