

SIG : Géoinformatique

Exercice 5 : Opérations sur les couches

Votre êtes employé.e par la commune de Lausanne pour faire une étude sur l'urbanisation de la commune. Vous devez calculer la longueur cumulée des routes de la commune ainsi que la surface au sol occupée par tous les bâtiments. La commune aimerait que vous fournissiez toutes ses informations directement dans une couche vectorielle en format geopackage. Vous devez donc créer un code dans la console python qui vous permette de calculer ces différents éléments et qui vous permette aussi de les ajouter à la couche comme nouveaux attributs.

Vous avez à disposition les données suivantes :

- Commune_lausanne.gpkg : Les limites de la commune de Lausanne avec les attributs suivants :

	fid	Commune	Canton	numero_id
1	5	Lausanne	Vaud	1

- Buildings_lausanne.gpkg : La couche vectorielle contenant la surface au sol occupée par tous les bâtiments de la commune de Lausanne avec les attributs suivants :

id	nr	nr2	datums_aenderung	datums_aenderung2	erstellungs_jahr	erstellungs_monat	gruend_aenderung	hersteller	herstell_jahr	herstell_monat	objektname	revision_jahr	revision_monat	revision_nr
1	14289	14289 (7546272-02)	29/03/2014	28/04/2016	2008	6	Verlassen	reitstall	2011	6	Gebrauch	2020	1	2021-04
2	14290	14290 (9467749-02)	28/04/2014	28/04/2016	2008	6	Verlassen	reitstall	2011	6	Other Gebrauch	2020	1	2021-04
3	14291	14291 (WA53300-02)	28/03/2014	28/04/2016	2008	6	Verlassen	reitstall	2011	6	Gebrauch	2020	1	2021-04
4	14292	14292 (6067781-02)	08/10/2016	28/04/2016	2008	6	Verlassen	reitstall	2012	6	Gebrauch	2020	1	2021-04

Comme bien souvent en Suisse, les attributs sont en allemand. Heureusement pour ceux qui n'ont pas appris l'allemand, nous allons utiliser ici uniquement la géométrie des différentes features.

- Roads_lausanne.gpkg : La couche vectorielles contenant toutes les routes de la commune de Lausanne avec les attributs suivants :

id	id	name	daten_verarbeitung	daten_verarbeitung	erstmalig_jahr	erstmalig_reihen	gesamt_anzahl	gesamt_reihen	hersteller	hersteller_id	hersteller_reihen	objektid	mission_jahr	mission_reihen	mission_anzahl
1	10488	E04781_103730C-011	38/11/2001	27/04/2008	2805	4	Verbesser	verbesser	2000	4	4m Straße	2020	E-2021_Acc		
2	10489	E04781_108819C-011-DEL	26/11/2001	27/04/2008	2805	4	Verbesser	verbesser	2000	4	3m Straße	2020	E-2021_DEL		
3	10250	E04726_108756C-042	28/11/2001	21/04/2006	2805	4	Verbesser	verbesser	2000	4	3m Straße	2020	E-2021_Acc		
4	10501	E04740_1036501-091	28/11/2001	27/04/2008	2805	4	Verbesser	verbesser	2000	4	8m Straße	2020	E-2021_Acc		

Comme pour les bâtiments, biens que les attributs soient en allemand, nous nous intéressons ici uniquement aux géométries.

A l'aide de ces couches vectorielles, ajoutez les attributs suivants à la couche communes_lausanne.gpkg :

- La longueur cumulée de toutes les routes de la commune avec comme nom d'attribut « longueur_cumulee » (float).

- La surface totale occupée par les bâtiments de la commune en hectares avec comme nom d'attribut « surface_cumulee » (float).
- Le pourcentage de la surface totale que représentent les bâtiments de la commune avec comme nom d'attribut « pct_surface » (entier).

De plus, l'attribut « numero_id » provient d'une erreur. Faites en sorte de retirer cet attribut de la couche vectorielle.

Pour mener à bien ce petit mandat, vous aurez probablement besoin de faire appel aux fonctionnalités suivantes de PyQGIS :

Créer un nouvel attribut :

```
layer.dataProvider().addAttributes([QgsField(attribute_name,QVariant.type)])
```

- **attribut_name** : une chaîne de caractères comme par exemple "pct_surface"
- **QVariant.type** : le type de variable conservée dans l'attribut comme par exemple : QVariant.Double, QVariant.Int, etc. Il faut donc changer *type* par *Int* ou *Double* par exemple.

Une fois les différents attributs créés, il est nécessaire de mettre à jour la couche avec la méthode `updateFields()` tel que :

```
layer.updateFields()
```

Ajouter / modifier la valeur d'un attribut :

Pour ajouter une valeur ou modifier la valeur d'un attribut pour un feature spécifique, il est nécessaire d'activer le mode **édition** de la couche sur laquelle on fait ces modifications comme suivant :

```
with edit(layer) :
```

```
#appliquer les différentes modifications ici (sur une feature avec la couche en mode édition)
feature[«attribut_name »] = ....
layer.updateFeature(feature)
```

Il faut mettre à jour la feature dès qu'on a fini de la modifier avec la méthode `updateFeature(feature)` tel que :

```
layer.updateFeature(feature)
```

Supprimer un attribut :

Pour supprimer un attribut, il faut récupérer de l'attribut par le biais de son nom avec la méthode suivante :

```
indice = layer.fields().indexFromName(«attribut_name »)
```

Et ensuite supprimer l'attribut avec la méthode `deleteAttribute(indice)` tel que :

```
layer.deleteAttribute(indice)
```

Attention, si vous ajouter des nouveaux attributs et que vous les modifiez, faites bien attention de mettre à jour les attributs et les features avant de supprimer un attribut pour ne pas avoir de confusion au niveau des indices.