

Exercice 20 et sa correction

QGIS 3.34 Septembre 2024



QGIS Perfectionnement

Table des matières

I - Exercice : Exercice récapitulatif du module QGIS Formulaires	3
Solutions des exercices	6

Exercice : Exercice récapitulatif du module QGIS Formulaires



Exercice récapitulatif du module QGIS Formulaires

Ce projet présente des informations sur l'agglomération de Nantes :

Ouvrir le projet QGIS **Formulaire_recapitulatif.qgs**.

- la couche des communes de l'agglomération
- la couche des quartiers de la commune de Nantes
- la couche des zones d'activités économiques de l'agglomération
- la couche des sections cadastrales de l'agglomération
- la couche des implantations potentielles de nouveaux équipements
- la table de référence des vocations de ces équipements et leurs catégories

Question

[solution n°1 p. 6]

Réaliser le formulaire suivant, comportant les informations descriptives des communes (couche "**Communes_Agglo_Ntes**")

Fiche Descriptive	
Nom_commun	BASSE-GOULAIN
INSEE_COMM	44009
Chef lieu de Canton	
▼ Géographie et population	
Region	PAYS DE LA LOIRE
Superficie	1374
Departement	LOIRE-ATLANTIQUE
Population	8168
Site Internet	http://www.basse-goulaine.fr/

Fiche descriptive communale

Ajouter un onglet « **Zones Économiques** » et donner la liste des zones économiques situées sur une commune lors de l'interrogation de la fiche communale :

Dans le formulaire de la couche « **Communes_Agglo_Ntes** », ajouter un onglet « **Photo** » intégrant le champ « **Photo** » de la couche.

Alimenter les communes de **LES SORINIERES** et **Reze** avec une photo.

Sur la couche « **Équipement potentiel** », réaliser un formulaire selon la copie d'écran ci-dessous :

Lors de la saisie d'un nouvel équipement :

- Les champs « **Commune** », « **Quartier** » et « **Section cadastrale** » sont alimentés automatiquement par interrogation des objets des couches « **Communes** », « **Quartier_Nantes** », et « **Sections_cadastrales** » (seule la commune de Nantes a, dans cet exercice, des quartiers)
- Le champ **Thème** doit proposer une liste déroulante des différentes valeurs du champ **OBJET** de la table **Typologie_eqpt_public**
- Le champ **Type** ne doit proposer que les valeurs du champ **CATEGORIE** de la table **Typologie_eqpt_public** correspondant au thème choisi.
- Le champ « **Parking** » doit répondre aux contraintes suivantes :
 - Tant qu'il n'y a pas de choix de Thème, le champ « **Parking** » est nul
 - Si l'on choisit le thème « **CULTURE** », on doit proposer moins de 50 places
 - Si l'on choisit le thème « **DEPLACEMENT** », on doit proposer plus de 100 places
 - Pour tout autre choix, le nombre de places est supérieur à 10

Enregistrer le projet sous le nom "**NomStagiaire_EX20.qgs**" dans le répertoire "**data_foad_qgis**"

Envoyez ce fichier par mail à la boîte aux lettres de l'équipe de formation qui vous a été indiquée dans votre protocole individuel de formation.

Solutions des exercices



[exercice p. 3] Solution n°1

Sur la couche **Communes_Agglo_Ntes**, faire un click droit **Propriétés** et choisir l'onglet **Formulaire d'attributs**.

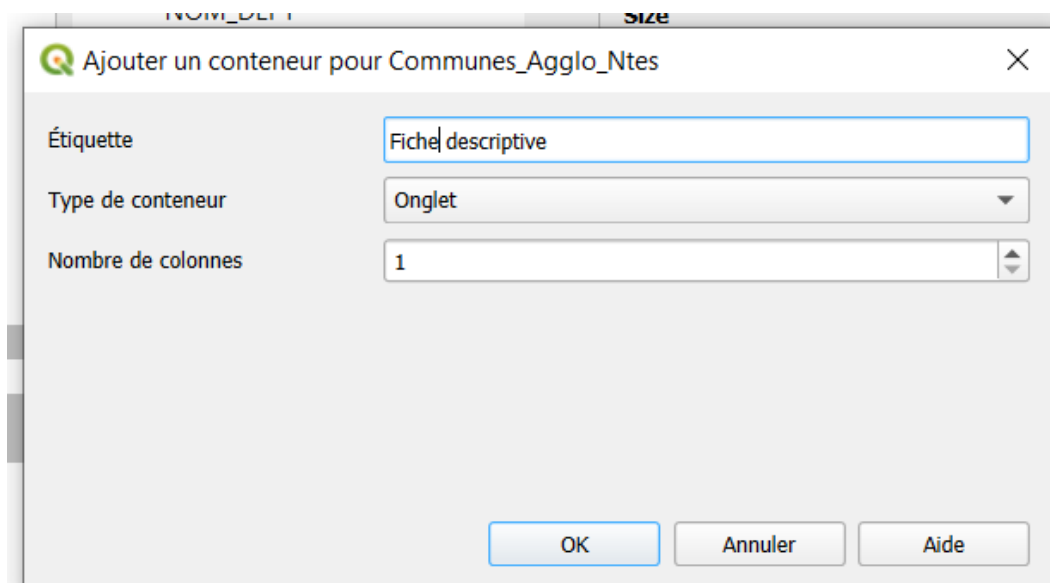
Pour les champs de la table renommés dans le formulaire, saisir les alias suivants :

- Champ Nom_commun Alias : Commune
- Champ INSEE_COMM Alias : Code INSEE
- Champ STATUT Alias : Chef-lieu de canton
- Champ NOM_REGION Alias : Région
- Champ NOM_DEPT Alias : Département
- Champ SUPERFICIE Alias : Superficie
- Champ POPULATION Alias : Population (en milliers)
- Champ Web Alias : Site Internet

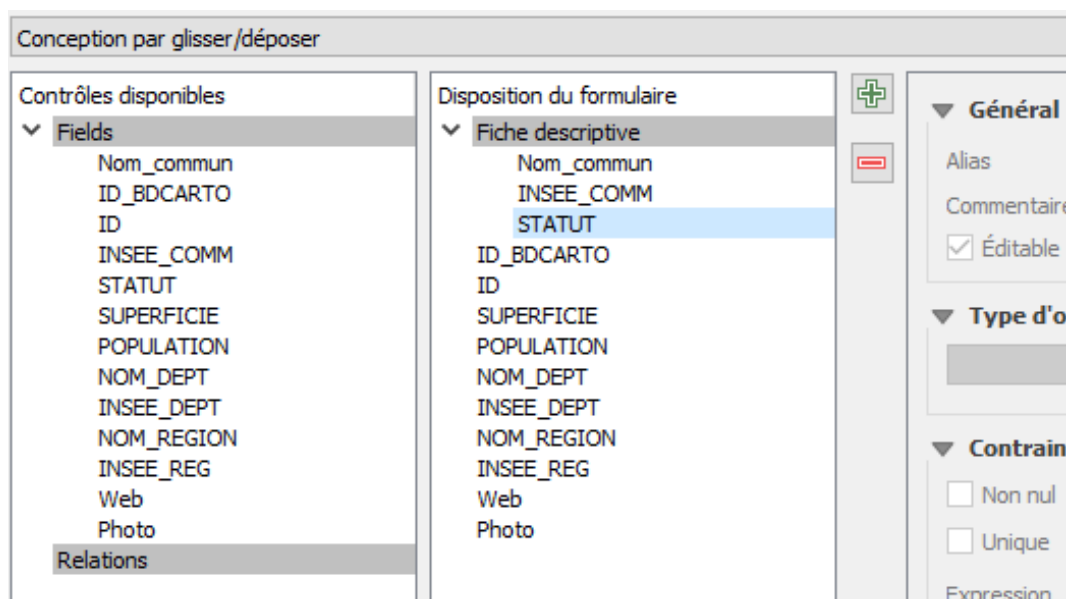
Pour construire le formulaire personnalisé, choisir dans la liste déroulante « **Conception par glisser/déplacer** ».



Créer l'onglet « **Fiche descriptive** » en cliquant sur le bouton

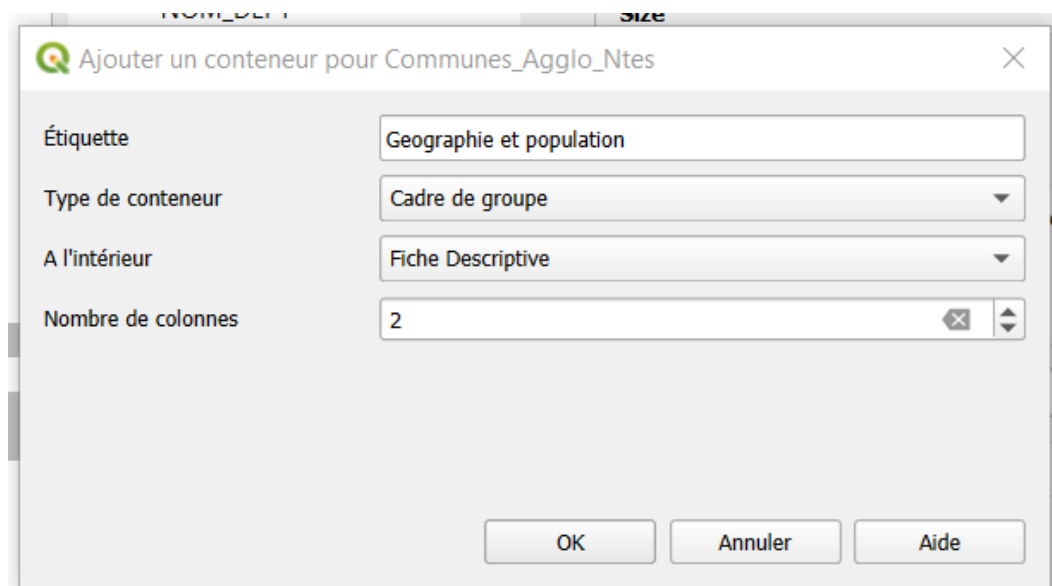


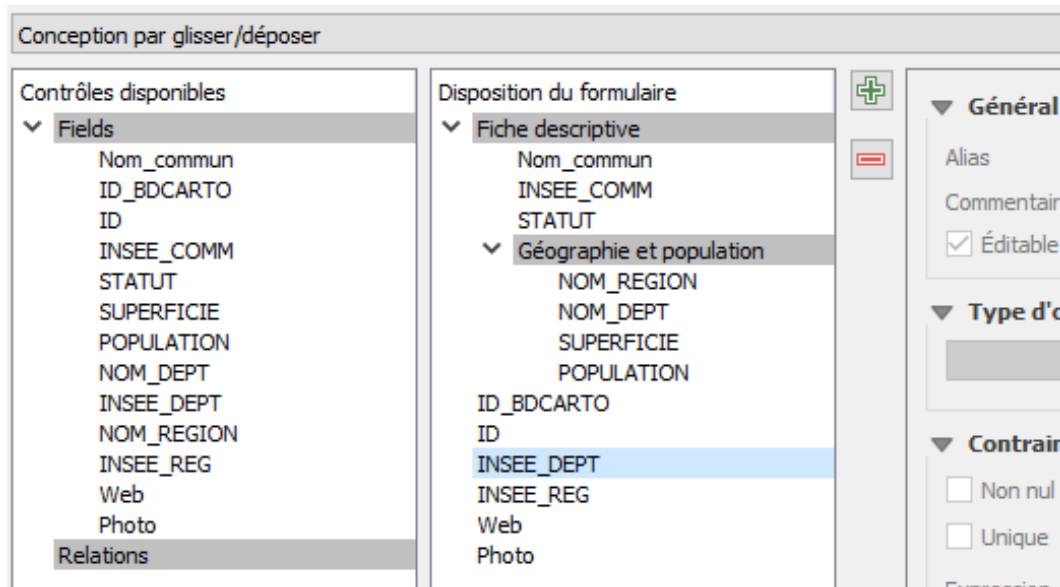
Glisser-déposer les champs « **Nom_commun** », « **INSEE_COMM** » et « **STATUT** » dans l'onglet « **Fiche descriptive** » :




Créer un groupe « **Géographie et population** » inclus dans l'onglet « **Fiche descriptive** ».

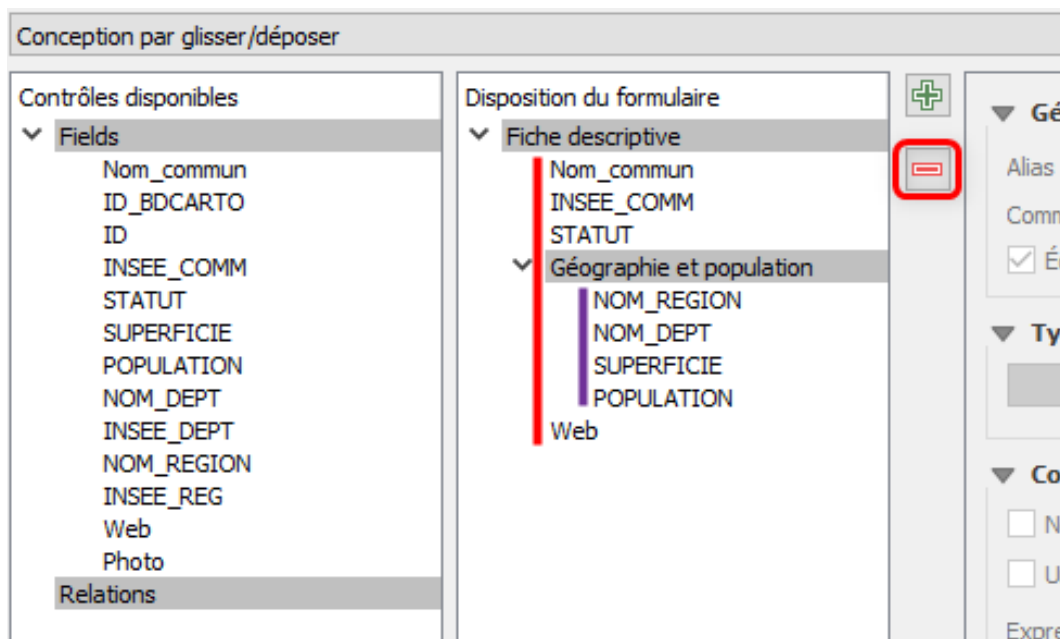
Le formater en **2 colonnes**





Glisser-déposer les champs « **NOM_REGION** », « **NOM_DPT** », « **SUPERFICIE** » et « **POPULATION** » dans le groupe **Géographie et population**

Le champ « **Web** » est à glisser dans l'onglet **Fiche descriptive** mais hors du groupe **Géographie et population** : contrôler l'incrémentation des champs et des groupes. Il ne reste qu'à supprimer les champs restant qui étaient glissés par défaut avec le bouton .



La mise en forme du formulaire étant réalisée, on va paramétrer les outils d'édition des champs :

- Champ « **Statut** » est une **case à cocher activée** sur la valeur **Chef-lieu de canton**

▼ **Général**

Alias

Commentaire

☒ Éditable ☐ Réutiliser la dernière valeur saisie ☐ Étiquette au sommet

▼ **Type d'outil**

Case à cocher

Représentation

L'état de représentation coché et décoché correspond aux valeurs brutes stockées dans la couche lorsque la case est cochée ou décochée.

Etat coché

Etat non coché

- Champ « **Web** » est une URL qui s'ouvre dans le navigateur par défaut. Il faut utiliser l'outil d'édition **Pièce jointe** et indiquer que l'on souhaite un visualiseur de page Web pour ouvrir la ressource.

▼ **Général**

Alias

Commentaire

☒ Éditable ☐ Réutiliser la dernière valeur saisie ☐ Étiquette au sommet

▼ **Type d'outil**

Pièce jointe

Type de stockage

Chemin

Chemin par défaut ...

Stocke le chemin comme

Mode d'enregistrement

☒ **Afficher le chemin de la ressource**

☐ **Afficher un bouton de sélection de fichier**

Filtre

☒ **Utiliser un hyperlien pour le chemin du document (lecteur seule)**

☒ **Afficher le chemin complet**

Visualiseur de document interne

Type

Largeur Spécifiez la taille de l'aperçu. Si vous laissez cette valeur à Auto, une taille optimale sera estimée.

Hauteur

2) Ajouter un onglet « **Zones Économiques** » et donner la liste des zones économiques situées sur une commune lors de l'interrogation de la fiche communale

L'objectif recherché est d'insérer le formulaire des zones économiques dans celui des objets de la couche des communes.


On va commencer par réaliser le formulaire de la couche des ZE (par glisser/déposer), puis on établira la relation entre les couches Communes et ZA pour insérer dans le formulaire Communes un sous-formulaire ZA

Pour réaliser le formulaire de la couche **Zones Activités**, reprendre les éléments supra.

- Champ **LIBELLE** Alias : **Zone**
- Champ **VOCATION_S** Alias : **Vocation**
- Champ **Surface** Alias : **Surface (ha)**

Les objets de la couche **Zones Activités** sont liés par le code **INSEE** aux **communes**.

Pour créer la relation, faire **Projet – propriétés du projet , onglet Relations**

Cliquer sur Ajouter une relation  Ajouter une relation

Ajouter une relation [X]

Id: [Généré automatiquement]

Nom: ZA_Commune

Force de la relation: Association

Couche et correspondance des champs

	Référencée (parent)	Référencante (enfant)
Couche	Communes_Agglo_Ntes	Zones Activités
Champ 1	abc Code INSEE	123 INSEE_COM

OK Annuler Aide

Nous avons désormais établi une relation de un-à-plusieurs entre chaque objet de la couche **Communes** et les **différentes ZA**.

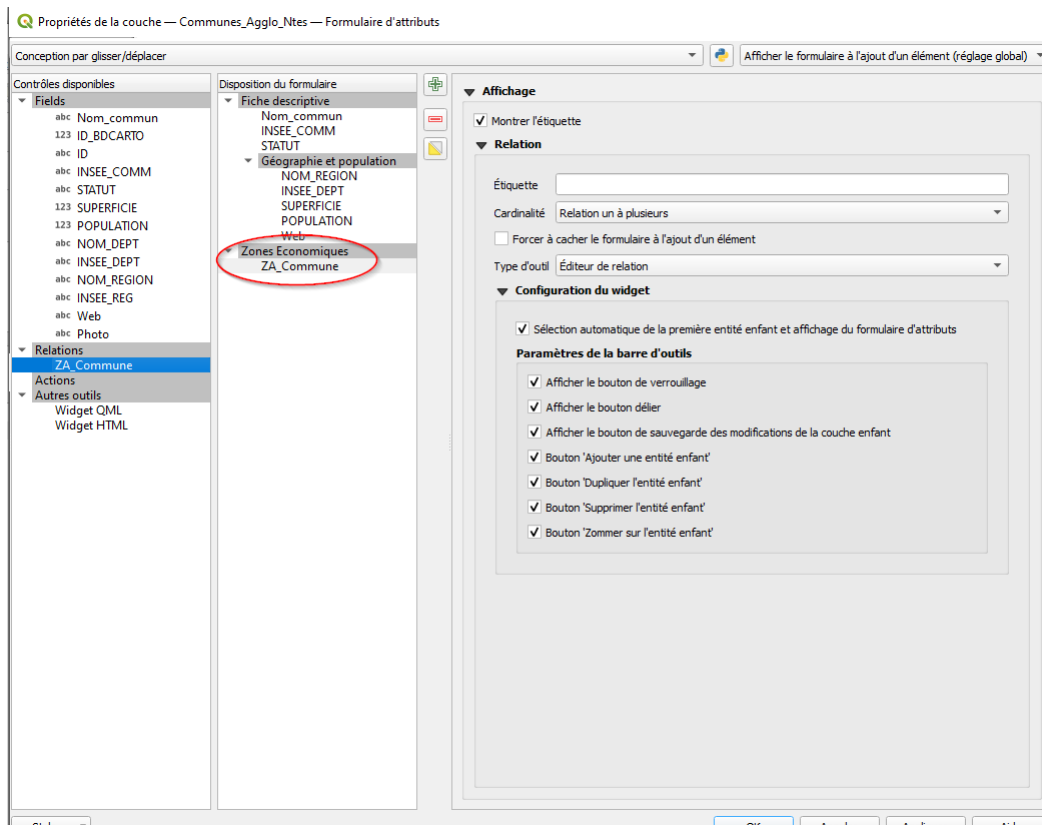
Nous pouvons intégrer cette relation dans le formulaire de la couche **Communes**.

On ouvre les propriétés de la couche **Communes**, onglet **Formulaire d'attributs**.

Dans le générateur de formulaire, on ajoute un nouvel onglet **Zones Économiques**.

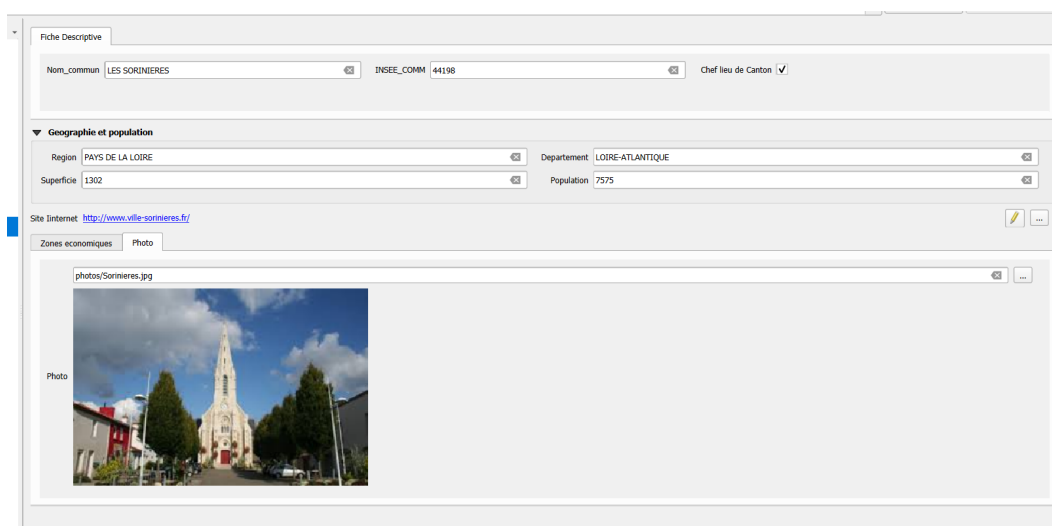
En dépliant le menu **Relations**, on trouve la relation définie dans le projet qu'il suffit de placer par glisser/déposer dans l'onglet créé.

On vient ainsi d'ajouter un sous-formulaire.

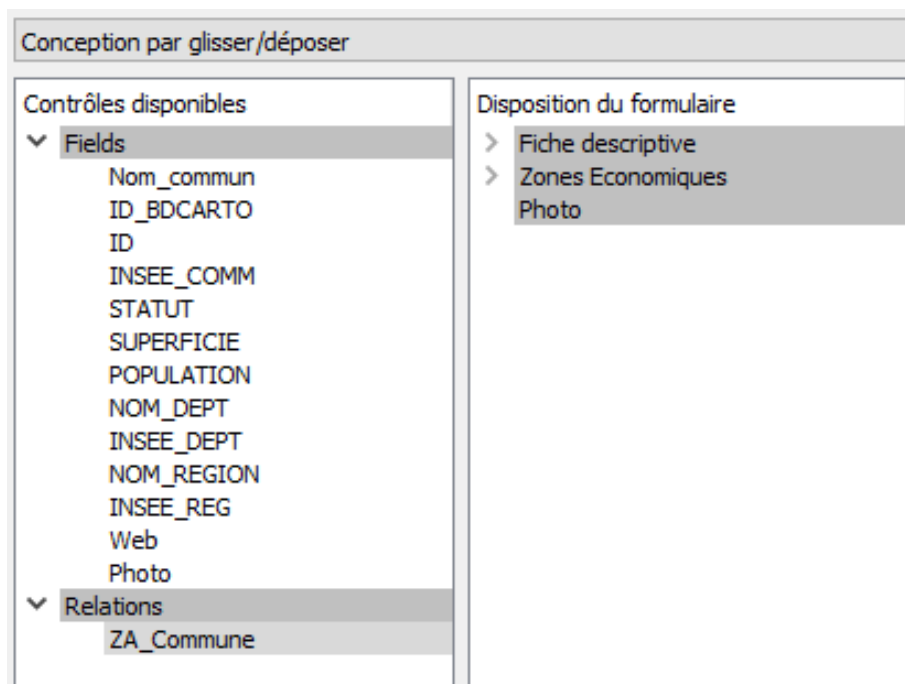


3) Dans le formulaire de la couche « **Communes_Aglo_Ntes** », ajouter un onglet « **Photo** » intégrant le champ « **Photo** » de la couche.

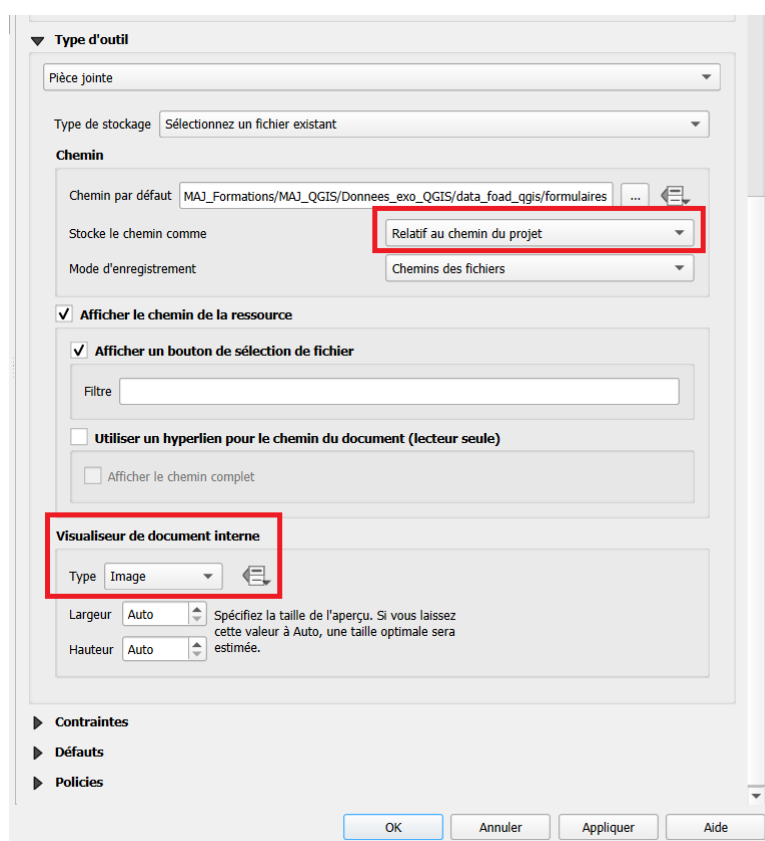
Alimenter les communes de Vertou et Reze avec une photo.



Dans la couche **Communes**, ajouter un onglet **Photo**



Définir le champ **Photo** de la couche **Communes** avec l'outil d'édition **Pièce jointe**



Sur la couche **Communes**, passer en mode Édition



Interroger une commune et cliquer sur l'onglet **Photo**.

En mode édition, l'outil **Photo** offre un explorateur de fichiers permettant d'associer un fichier à l'objet en cours d'édition.

Aller dans le répertoire **Photos** pour désigner une image et repasser en mode consultation de la couche **Communes**.

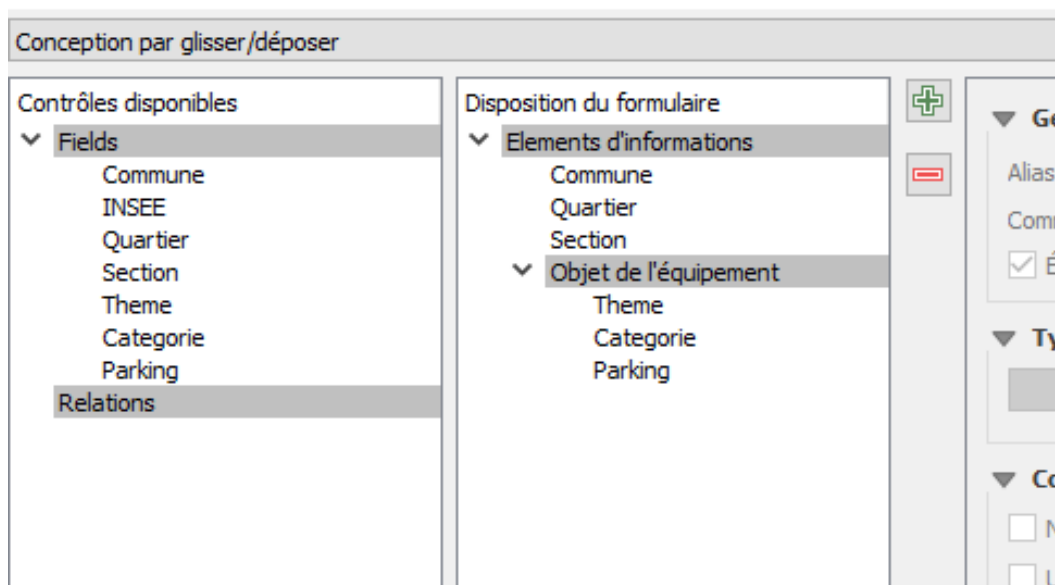
La photo associée à l'objet est visible dans l'onglet

4) Sur la couche « Équipement potentiel », faire un formulaire selon la copie d'écran ci-dessous.

Lors de la saisie d'un nouvel équipement :

- Les champs « **Commune** », « **Quartier** » et « **Section cadastrale** » sont alimentés automatiquement par interrogation des objets des couches « **Communes** », « **Sections cadastrales** » et « **Quartier Nantes** » (seule la commune de Nantes a, dans cet exercice, des quartiers)
- Le champ **Thème** doit proposer une liste déroulante des différentes valeurs du champ OBJET de la table **Typologie_eqpt_public**
- Le champ **Type** ne doit proposer que les valeurs du champ **CATEGORIE** de la table **Typologie_eqpt_public** correspondant au thème choisi.
- Le champ « **Parking** » doit répondre aux contraintes suivantes :
 - Tant qu'il n'y a pas de choix de Thème, le champ « Parking » est nul
 - Si l'on choisit le thème « CULTURE », on doit proposer moins de 50 places
 - Si l'on choisit le thème « DEPLACEMENT », on doit proposer plus de 100 places
 - Pour tout autre choix, le nombre de places est supérieur à 10

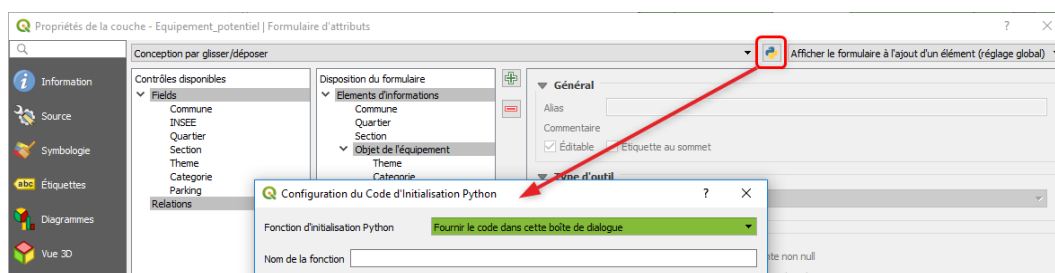
Créer le **formulaire** sur la couche **Equipement_potentiel**



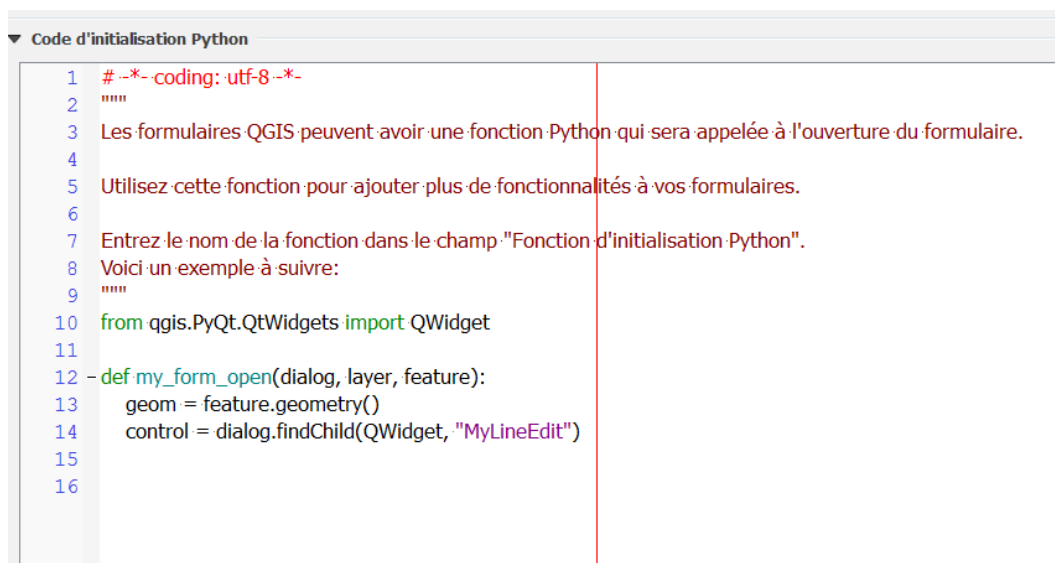
Pour alimenter les champs « **Commune** », « **Quartier** », « **Section** » automatiquement, on va utiliser le script Python fourni.

Ce script est basé sur la fonction **Contains** et retourne les attributs souhaités des objets des différentes couches constitutives du projet.

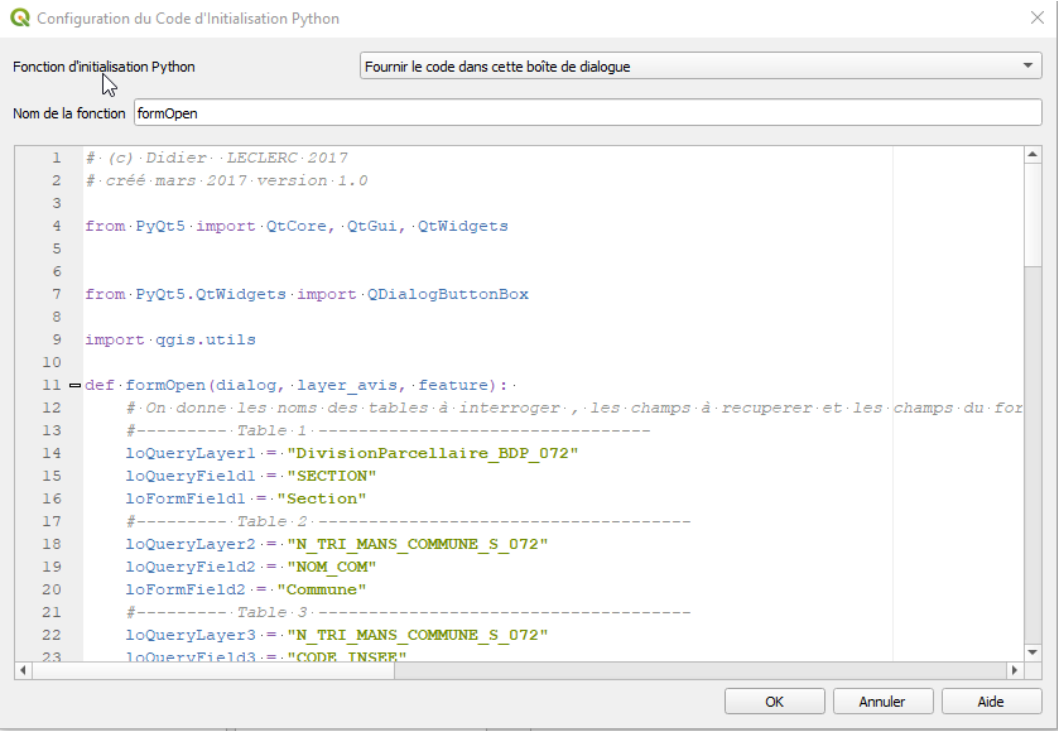
Dans l'onglet Formulaire d'attributs de la couche Equipement_potentiel, indiquer que l'on souhaite fournir le code dans cette boîte de dialogue



Nettoyer la boîte de dialogue du code d'initialisation Python que QGIS fournit par défaut



Ouvrir le script Python fourni dans un fichier texte et en copier le contenu dans la boîte de dialogue



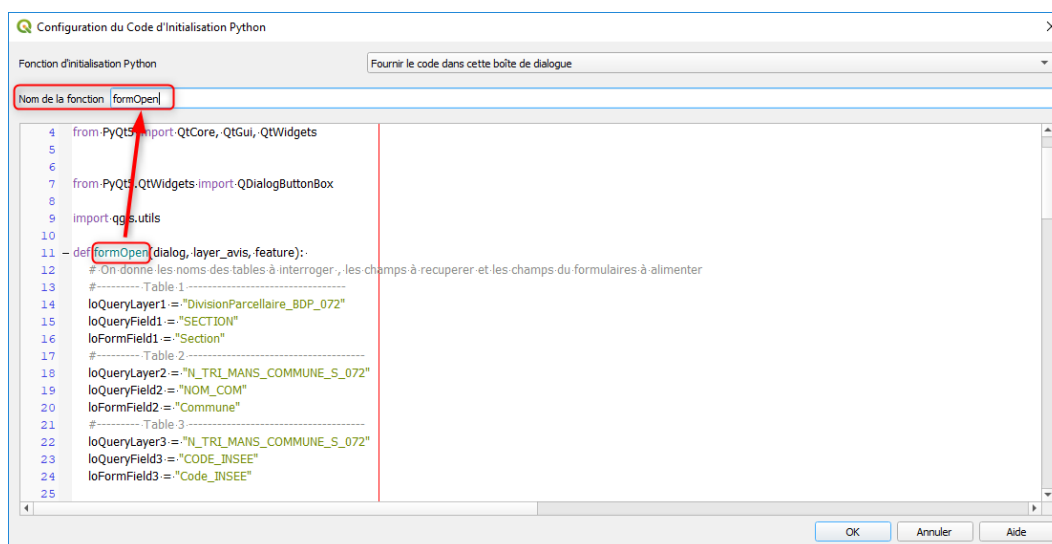
Le script va interroger les couches désignées et retourner, pour les objets de ces couches contenant le point saisi, les valeurs des attributs dans les champs du formulaire

Champ du formulaire à alimenter	Nom de la couche fournisseuse	Nom du champ de la couche fournisseuse
Commune	Communes_Agglo_Ntes	Nom_commun
Quartier	Quartiers_Nantes	NOM
Section	Sections_cadastrales	SECTION

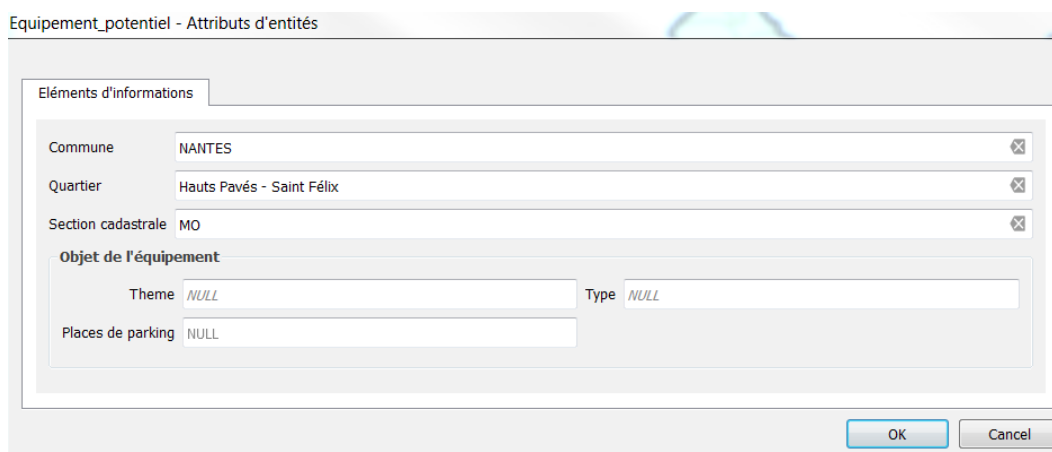
On porte ces paramètres dans le script

```
5 - def formOpen(dialog, layer_avis, feature):
6
7     # On donne les noms des tables à interroger, les champs à récupérer et les champs du fo
8     #-----Table 1-----
9     loQueryLayer1 = "Communes_Agglo_Ntes"
10    loQueryField1 = "Nom_commun"
11    loFormField1 = "Commune"
12    #-----Table 2-----
13    loQueryLayer2 = "Quartiers_Nantes"
14    loQueryField2 = "NOM"
15    loFormField2 = "Quartier"
16    #-----Table 3-----
17    loQueryLayer3 = "Sections_cadastrales"
18    loQueryField3 = "SECTION"
19    loFormField3 = "Section"
20
```


On indique à QGIS le nom de la fonction du script



Lors de la saisie d'un nouvel objet dans la couche **Equipement_potentiel**, les champs « **Commune** », « **Quartier** » et « **Section** » du formulaire sont désormais alimentés :



Le champ **Thème** du formulaire est une liste déroulante proposant les valeurs du champ OBJET de la table **Typologie_eqpt_public**.

Dans les propriétés de la couche **Equipement_potentiel**, onglet **Formulaire d'attributs**, déclarer le champ **Theme** avec l'outil d'édition **Liste de valeurs**.

On choisit **Charger des données depuis la couche** :

The screenshot shows the QGIS 'Form Builder' window for the 'Equipement_potentiel' layer. The 'Type d'outil' (Tool type) section is active, and the 'Charger des données depuis la couche' (Load data from layer) button is highlighted with a red arrow. A modal dialog titled 'Charger les valeurs depuis la couche' (Load values from layer) is open, displaying a table of data from the 'Typologie_eqpt_public' layer. The table has two columns: 'Valeur' (Value) and 'Description' (Description). The data is as follows:

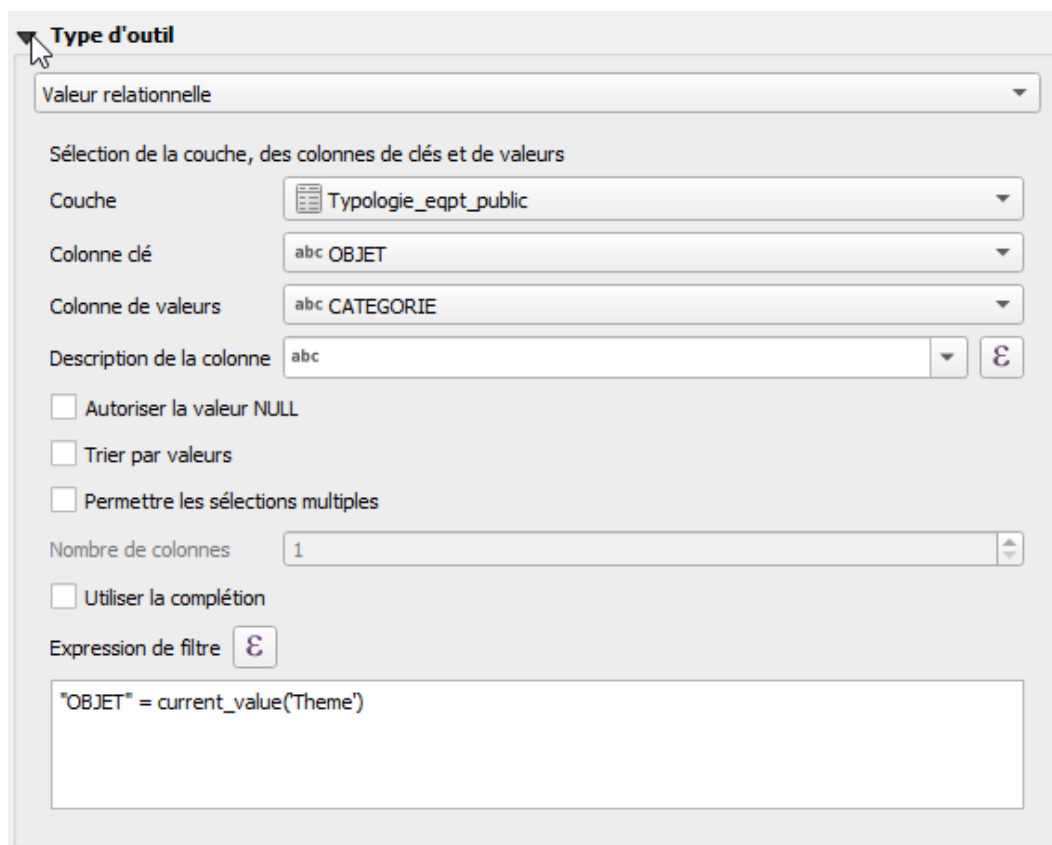
Valeur	Description
1	CULTURE
2	DEPLACEMENT
3	SPORT&LOISIRS
4	

Nota : dans la table **Typologie_eqpt_public**, le champ **OBJET** contient plusieurs fois les mêmes valeurs d'attributs.

Si l'on avait fait le choix de l'outil d'édition « Valeur relationnelle », la liste déroulante aurait présenté ces multiples doublons.

Le champ **Categorie** (alias **Type**) du formulaire ne doit proposer que les valeurs de la table **Typologie_eqpt_public** filtrées selon le choix du **Thème**.

On va utiliser l'outil d'édition **Valeur relationnelle**



Type d'outil

Valeur relationnelle

Sélection de la couche, des colonnes de clés et de valeurs

Couche: Typologie_eqpt_public

Colonne clé: abc OBJET

Colonne de valeurs: abc CATEGORIE

Description de la colonne: abc

☐ Autoriser la valeur NULL

☐ Trier par valeurs

☐ Permettre les sélections multiples

Nombre de colonnes: 1

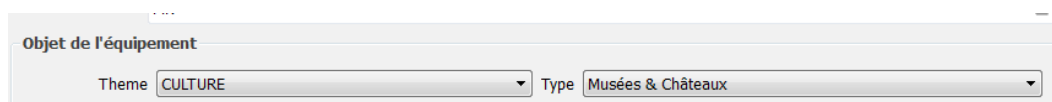
☐ Utiliser la complétion

Expression de filtre: ε

"OBJET" = current_value('Theme')

Lors de la saisie d'un nouvel objet dans la couche **Equipement_potentiel**, le champ **Theme** contient la liste des valeurs de la table **Typologie_eqpt_public** et le contenu du champ **Type** est filtré selon le choix précédent

Nb : sous QGIS 3.4.5, il faut choisir explicitement un thème dans la liste déroulante, ou avoir indiqué qu'une valeur par défaut pour le Thème, 'CULTURE' par exemple.



Objet de l'équipement

Theme: CULTURE

Type: Musées & Châteaux

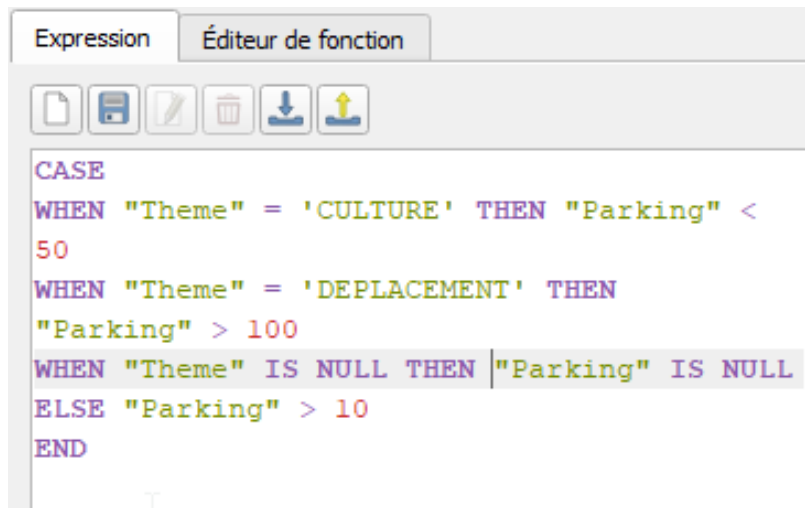
Le champ « **Parking** » doit répondre aux contraintes suivantes :

- Tant qu'il n'y a pas de choix de Thème, le champ « Parking » est nul
- Si l'on choisit le thème « CULTURE », on doit proposer moins de 50 places
- Si l'on choisit le thème « DEPLACEMENT », on doit proposer plus de 100 places
- Pour tout autre choix, le nombre de places est supérieur à 10

Le champ **Parking** de la couche **Equipement_potentiel** doit intégrer une contrainte vérifiant les conditions.

On écrit la contrainte après avoir cliqué sur





The screenshot shows a software interface with two tabs at the top: "Expression" and "Éditeur de fonction". The "Éditeur de fonction" tab is active. Below the tabs is a toolbar with six icons: a document, a save icon, a pencil, a trash can, a download arrow, and an upload arrow. The main area is a text editor containing a SQL CASE statement. The text is as follows:

```
CASE
WHEN "Theme" = 'CULTURE' THEN "Parking" <
50
WHEN "Theme" = 'DEPLACEMENT' THEN
"Parking" > 100
WHEN "Theme" IS NULL THEN "Parking" IS NULL
ELSE "Parking" > 10
END
```

The text is color-coded: keywords like CASE, WHEN, THEN, ELSE, and END are in purple; string literals like 'CULTURE' and 'DEPLACEMENT' are in green; and numeric values like 50, 100, and 10 are in red. The line "WHEN 'CULTURE' THEN 'Parking' < 50" is split across two lines. The line "WHEN 'DEPLACEMENT' THEN 'Parking' > 100" is also split across two lines. The line "WHEN 'CULTURE' THEN 'Parking' < 50" is highlighted with a light blue background. At the bottom of the editor, there is a small cursor icon.