

QGIS, ses formulaires et expressions pour faire un outil de saisie au poil

Comment exploiter QGIS pour créer une application de saisie et d'exploitation de données professionnelle ?



Michaël DOUCHIN - 3LIZ

Les points abordés

- **Import** d'une donnée "tableur" dans un Geopackage et affichage dans QGIS
- Création et correction de données via des **expressions**
- Mettre les données en **relation**
- Créer un **formulaire de saisie** avancé
- Visualiser les données problématiques ou importantes dans la **table attributaire**

On privilégiera dans cette présentation l'utilisation d'**expressions** dans QGIS pour la majorité des traitements:
-> pas d'utilisation de requêtes SQL en base de données ou d'algorithmes complexes.

L'exemple choisi

- On fait des **observations faunistiques** sur le terrain: **Moqueur Gorge Blanche** à la Martinique
- La saisie est historiquement faite **dans un tableur** (MS Excel ou LibreOffice Calc)
- Une observation est
 - un **point** avec une longitude et une latitude,
 - faite à une **date**,
 - par des **observateurs**
 - On doit respecter une certaine **nomenclature**



Les données source

Un fichier au format LibreOffice Calc `observations_source.ods`

espece_support	date_observation	commentaires	observateurs	hauteur_nid	nid_nb_oeuf	nid_nb	nid_r	nid_p	longitude	latitude
Maytenus laevigata	12/08/2020	Observation ponc	DUPONT Jean (ACME), DULOUP	3,82	0	0	0	1	718446,57138	1629776,8986
Pisonia fragans	15/09/2020	Caméra installée	DUPONT Jean (ACME), DULOUP	2,32	0	0	0	0	719111,92879	1629578,8868
Pisonia fragans	23/09/2020	Présence de coq	DUPONT Jean (ACME), DUBOIS	2,30	0	0	0	0	719287,53531	1629202,8678
Myrcia citrifolia	23/09/2020		DUPONT Jean (ACME), DUBOIS	3,26	0	0	0	0	716186,33091	1628888,6646
Myrcia citrifolia	23/09/2020		DUPONT Jean (ACME), DUBOIS	4,80	0	0	0	0	718663,03803	1629327,0948
Pisonia fragans	23/09/2020		DUPONT Jean (ACME), DUBOIS	2,97	0	0	0	0	718154,58598	1629292,9505
Eugenia ligustrina	23/09/2020	Caméra installée	DUPONT Jean (ACME), DUBOIS	2,36	1	0	0	0	717713,30872	1630189,9734
Pisonia fragans	23/09/2020		DUPONT Jean (ACME), DUBOIS	3,90	0	0	0	0	716691,11436	1628843,1457
Maytenus laevigata	23/09/2020		DUPONT Jean (ACME), DUBOIS	3,82	0	0	0	0	718487,46142	1628844,6034

Importer la donnée source

- On ajoute la donnée dans QGIS, par glisser-déplacer du fichier LibreOffice dans QGIS
- On ouvre les propriétés de la couche `observations_source` et visualise les champs dans l'onglet **Champs**

Id ▾	Nom	Alias	Type	Type identifié	Longueur	Précision
abc 0	espece_support		QString	String	0	0
abc 1	date_observation		QString	String	0	0
abc 2	commentaires		QString	String	0	0
abc 3	observateurs		QString	String	0	0
1.2 4	hauteur_nid		double	Real	0	0
abc 5	nid_nb_oeuf		QString	String	0	0
abc 6	nid_nb_poussin		QString	String	0	0
abc 7	nid_nb_immature		QString	String	0	0
abc 8	nid_nb_adulte		QString	String	0	0
1.2 9	longitude		double	Real	0	0
1.2 10	latitude		double	Real	0	0

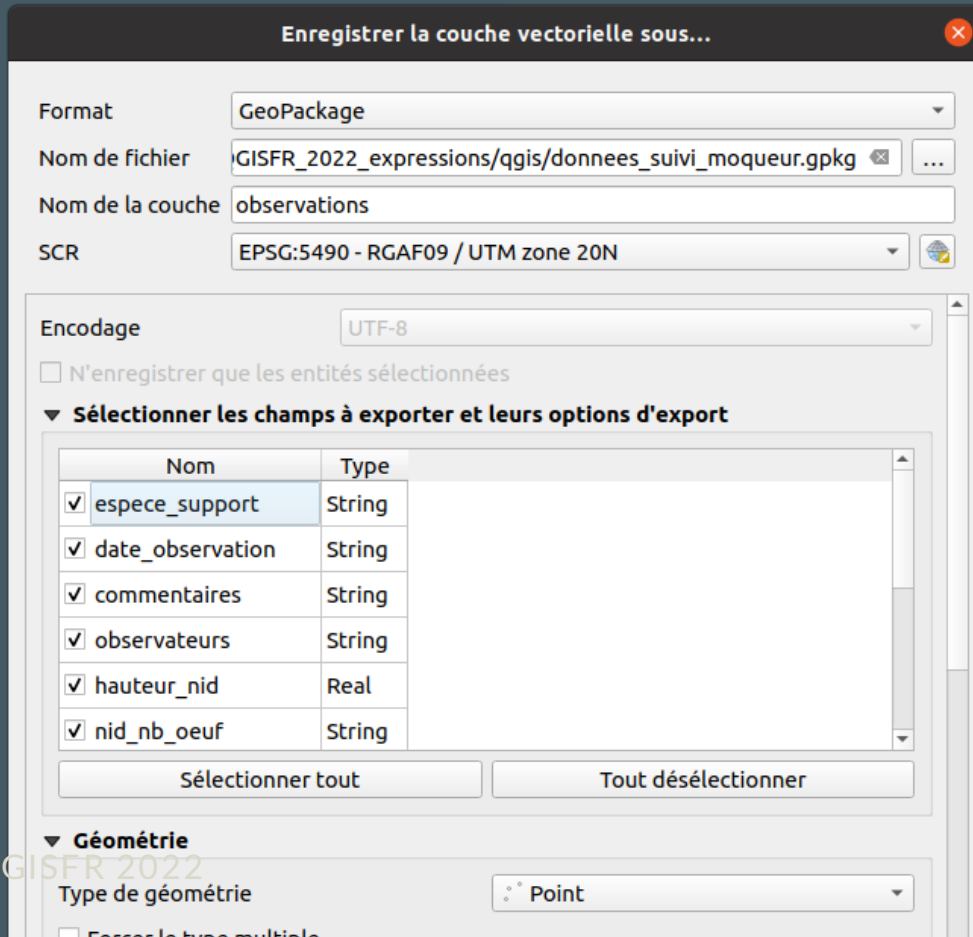
Étudier la donnée source

On visualise la table attributaire qui reprend bien ce qu'on avait dans LibreOffice:

	espece_support	date_observation	commentaire	observateurs	hauteur_nid	nid_nb_oeuf	nid
1	Myrcia citri...	15/05/2013	Program...	NULL	2,5	NULL	1
2	Myrcia citri...	23/05/2013	Program...	NULL	2,16	1	1
3	Pisonia fra...	23/05/2013	Program...	NULL	3,8	NULL	NU
4	NULL	23/05/2013	Program...	NULL	NULL	NULL	NU
5	Myrcia citri...	30/05/2013	Program...	NULL	2,96	2	NU
6	NULL	30/05/2013	Program...	DESCHAMPS Laure (BIRD)	2,98	NULL	NU
7	Damburne...	06/06/2013	Program...	DESCHAMPS Laure (BIRD)	3	2	0
8	Damburne...	13/06/2013	Program...	ATICIEN Jérôme (ACME), DESCHAMPS Laure (BI...	3,16	NULL	NU
9	Myrcia citri...	13/06/2013	NULL	DESCHAMPS Laure (BIRD)	2,41	2	NA
10	NULL	13/06/2013	Program...	DESCHAMPS Laure (BIRD)	4,46	NULL	NU
11	NULL	18/06/2013	Program...	ATICIEN Jérôme (ACME), DESCHAMPS Laure (BI...	2,66	1	NU
12	Myrcia citri...	18/06/2013	Program...	ATICIEN Jérôme (ACME), DESCHAMPS Laure (BI...	3	NSP	NU
13	Damburne...	20/06/2013	Program...	DESCHAMPS Laure (BIRD)	2,16	NULL	NU
14	Myrcia citri...	20/06/2013	Program...	DESCHAMPS Laure (BIRD)	2,69	NULL	NU

Sauvegarder cette donnée en GeoPackage

On sauvegarde dans un **GeoPackage** `donnees_suivi_moqueur.gpkg` dans le **répertoire du projet**, table `observations`, de type `Point` même si la table initiale n'est pas spatiale



Enregistrer la couche vectorielle sous...

Format: GeoPackage

Nom de fichier: GISFR_2022_expressions/qgis/donnees_suivi_moqueur.gpkg

Nom de la couche: observations

SCR: EPSG:5490 - RGAF09 / UTM zone 20N

Encodage: UTF-8

☐ N'enregistrer que les entités sélectionnées

▼ Sélectionner les champs à exporter et leurs options d'export

Nom	Type
<input checked="" type="checkbox"/> espece_support	String
<input checked="" type="checkbox"/> date_observation	String
<input checked="" type="checkbox"/> commentaires	String
<input checked="" type="checkbox"/> observateurs	String
<input checked="" type="checkbox"/> hauteur_nid	Real
<input checked="" type="checkbox"/> nid_nb_oeuf	String

Sélectionner tout Tout désélectionner

▼ Géométrie

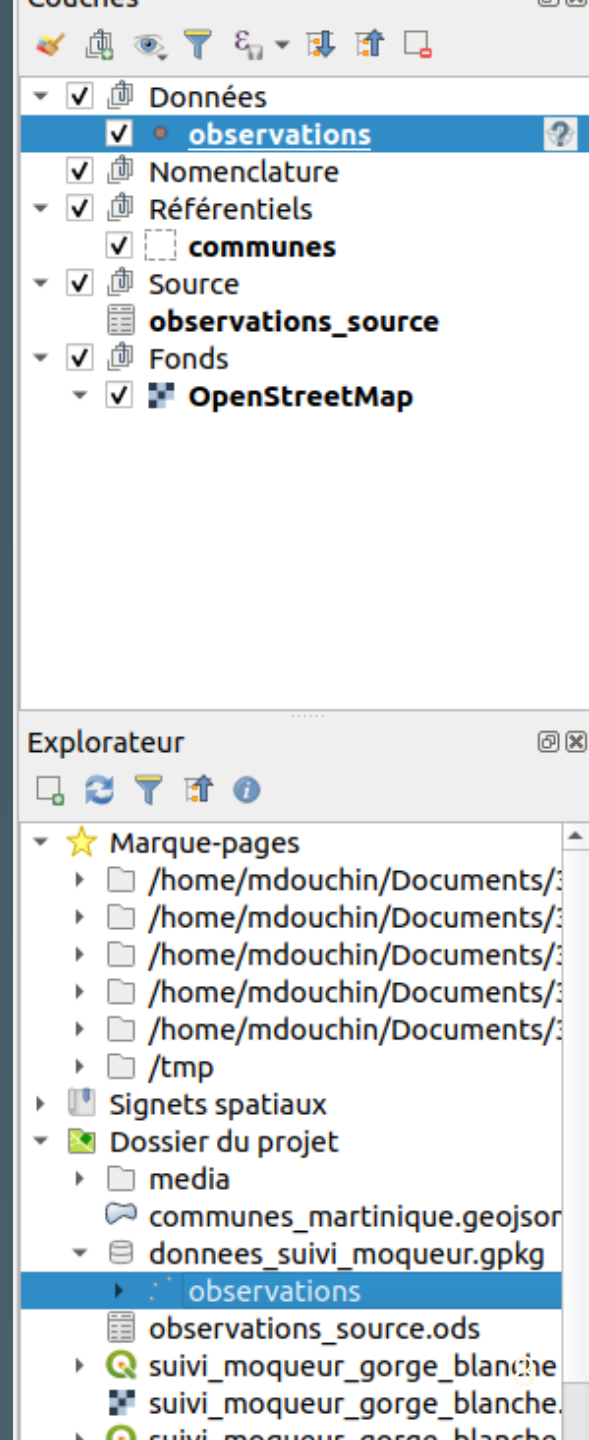
Type de géométrie: Point

☐ Forcer le type multiple

Ouverture de la couche du GeoPackage dans QGIS

On a **importé** le fichier tableur LibreOffice dans un **Geopackage** comme une couche de **Points**.

- Via l'**explorateur** de QGIS, on cherche le **Dossier du projet**,
- On double-clique sur le fichier `donnees_suivi_moqueur.gpkg`
- ajoute la nouvelle table `observations` au projet
- On configure la **projection**, dans notre cas `EPSG:5490`, UTM Zone 20N (Martinique)
- On modifie la **symbolologie** (gros points bleus avec bordure blanche)



Modifier la géométrie via une expression

On va définir les points via **la longitude et latitude** et une

- On passe cette couche en édition
- On utilise la calculatrice de champs et on modifie la géométrie via une expression `make_point("longitude", "latitude")`

donnees_suivi_moqueur observations — Calculatrice de champ

☐ Ne mettre à jour que les 0 entités sélectionnées

☐ Créer un nouveau champ

☐ Créer un champ virtuel

Nom

Type

Longueur du nouveau champ Précision

☒ Mise à jour d'un champ existant

<geometry>

Expression Éditeur de fonction

`make_point(longitude, latitude)`

make_poi

Afficher l'aide

▼ Géométrie

- make_point
- make_point_m

▼ Récent (fieldcalc)

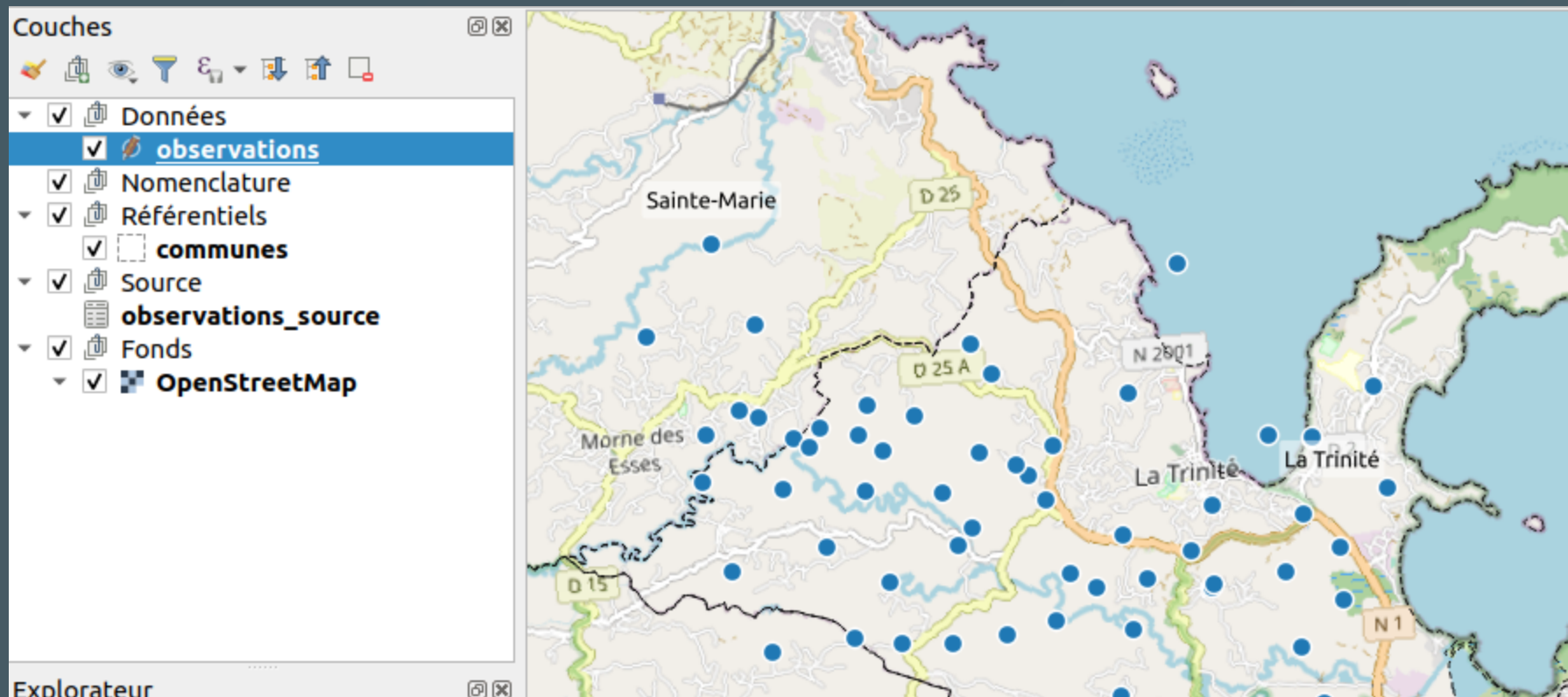
- make_point(720834, ...)

fonction make_point

Crée une géométrie ponctuelle à partir des valeurs x et y (et optionnellement des valeurs z et m).

On visualise alors les points sur la carte

Les géométries sont bien créées et correspondent aux coordonnées. On **sauvegarde** les données via la disquette de la **barre de numérisation** !



On nettoie les valeurs des champs avec des entiers 1/3

Les valeurs **non entières** `NAN`, `NSP`, `Inconnu`, `RAS` ne doivent pas apparaître dans les champs qui attendent des **entiers**

hauteur_nid	nid_nb_oeuf	nid_nb_poussin
2,41	2	NA
4,46	<i>NULL</i>	<i>NULL</i>
2,66	1	<i>NULL</i>
3	NSP	<i>NULL</i>

On peut appliquer une expression avec la **calculatrice de champ** sur tous les champs qui doivent contenir des entiers: `nid_nb_oeuf`, `nid_nb_poussin`, `nid_nb_immature`, `nid_nb_adulte`

```
-- Une condition avec une expression régulière pour ne conserver que les entiers
CASE
    WHEN regexp_match(trim("nid_nb_oeuf"), '^\\d+$')
        THEN to_int(trim("nid_nb_oeuf"))
    ELSE NULL
END
```

On nettoie les valeurs des champs avec des entiers 2/3

Pour `nid_nb_oeuf`, `nid_nb_poussin`, `nid_nb_immatrice`, `nid_nb_adulte`

hauteur_nid	nid_nb_oeuf	nid_nb_poussin
2,41	2	NA
4,46	NULL	NULL
2,66	1	NULL
3	NULL	NULL

observations — Calculatrice de champ

☐ Ne mettre à jour que les 0 entités sélectionnées

☐ Créer un nouveau champ ☒ Mise à jour d'un champ existant

☐ Créer un champ virtuel

Nom

Type

Longueur du nouveau champ Précision

abc nid_nb_oeuf

Expression Éditeur de fonction

```
CASE
  WHEN regexp_match(trim("nid_nb_oeuf"),
    '^\\d+$')
  THEN to_int(trim("nid_nb_oeuf"))
  ELSE NULL
END
```

Afficher l'aide

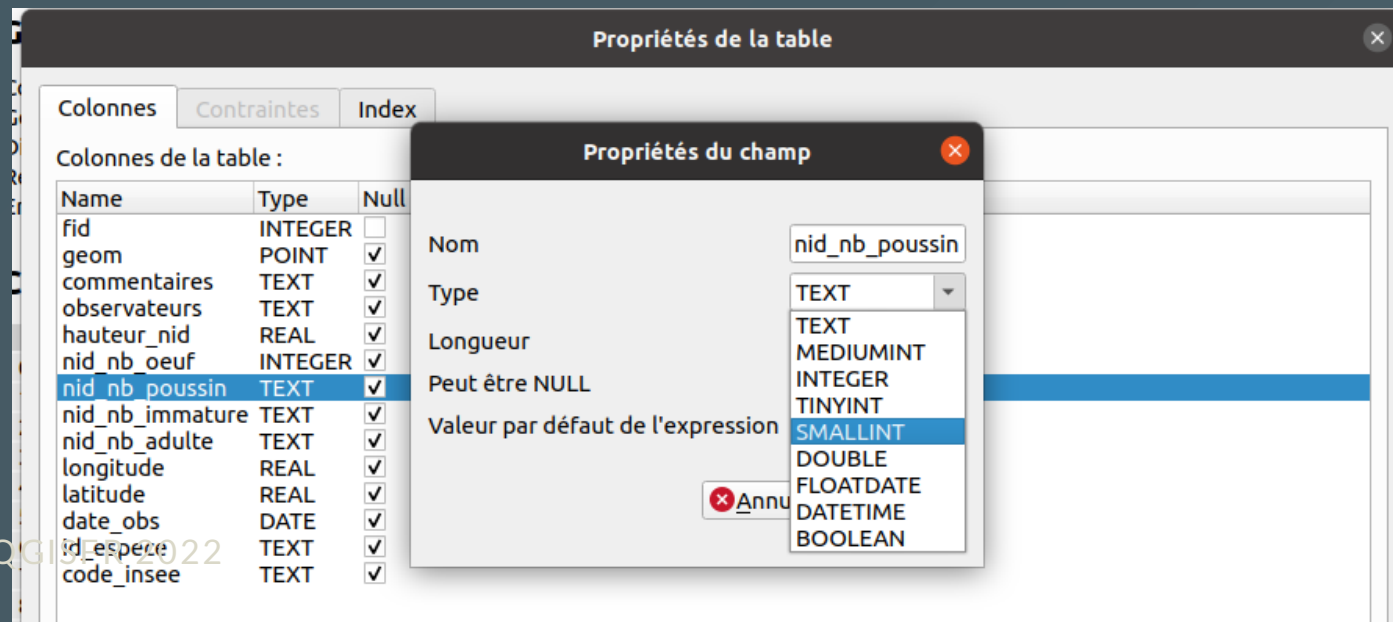
row_number

- ▶ Agrégats
- ▶ Chaîne de carac...
- ▶ Champs et Vale...
- ▶ Conditions
- ▶ Conversions

On nettoie les valeurs des champs avec des entiers 3/3

On doit utiliser le **Gestionnaire de bases de données** pour changer le type des champs `nid_nb_XXXX` de **Texte** à **Entier**

- On crée une **connexion** vers le GeoPackage
- Sur la table `observations`, on ouvre le menu **Table > Modifier une table**, et on choisit le type `SMALLINT` ou `INTEGER`



On transforme un champ texte en Date 1/2

Exemple du champ `date_observation` de type texte, qui contient par exemple un texte `26/06/2013`:

- On crée un nouveau champ `date_obs` de type `Date` via la **calculatrice de champ** avec:

```
to_date("date_observation", 'dd/MM/yyyy')
-- on aura donc: 2013-06-26
```

- On obtient

date_observation	commentaire	observateurs	auteur_n	nb_oe	nb_pou	nb_imma	nb_adu	longitude	latitude	date_obs
15/05/2013	Progr...	NULL	2,5	NULL	1	NULL	NULL	714346,73...	1630612,1...	2013-05-15
13/06/2013	Progr...	DESCHAM...	4,46	NULL	NULL	NULL	NULL	715925,64...	1630965,5...	2013-06-13

- On supprime le champ `date_observation`
- On enregistre

On transforme un champ texte en Date 2/2

- On pourra donc faire des **calculs de date**, par exemple l'âge en jours de l'observation, via

```
to_int(  
    day(age(now(), "date_obs"))  
)
```

qui renverra par exemple **128** jours

- Ou tester si la date donnée est bien supérieure à la date du jour

```
"date_obs" >= now()
```

qui renverra **Vrai** ou **Faux**

On veut créer une table de nomenclature pour les espèces

On souhaite faire une nouvelle table non spatiale dans le Geopackage:

- table `especies` : pour stocker la liste des espèces support (arbres ou arbustes)

Dans la table source, on a pour l'instant un nom d'espèce dans le champ `espece_support` :

espece_support
Damburneya coriacea
Myrcia citrifolia
Ouratea guildingii

On pourra ensuite créer des **relations** entre les **observations** et les **espèces**

Créer une table avec les espèces uniques 1/2

- Utiliser l'algorithme **Ajouter un champ d'index de valeur unique** qui va produire une **nouvelle table** contenant les valeurs distinctes du champ **espece_support**. Ouvrir la table via l'**explorateur**

The screenshot shows the 'Ajouter un champ d'index de valeur unique' (Add a unique value index field) processing tool in QGIS. The tool is configured with the following parameters:

- Paramètres** (Parameters) tab is selected.
- Couche source** (Source layer): 'observations [EPSG:5490]'.
- Entité(s) sélectionnée(s) uniquement** (Only selected entity(ies)): ☐.
- Champ de classification** (Classification field): 'abc espece_support'.
- Nom** (Name): 'id_espece'.
- Couche avec un champ d'index [optionnel]** (Optional layer with an index field): '[Créer une couche temporaire]' (Create a temporary layer).
- Ouvrir le fichier en sortie après l'exécution de l'algorithme** (Open output file after algorithm execution): ☐.
- Synthèse de la classe [optionnel]** (Optional class synthesis): '2022-01-QGISFR_2022_expressions/qgis/donnees_suivi_moqueur.gpkg' table="especes" (with a file icon).
- Ouvrir le fichier en sortie après l'exécution de l'algorithme** (Open output file after algorithm execution): ☒.

On the right, a help panel titled 'Ajouter un champ d'index de valeur unique' explains the algorithm: 'Cet algorithme prend un couche vectorielle et un attribut et ajoute un nouveau champ numérique. Les valeurs de ce champ correspondent aux valeurs de l'attribut spécifié. Par conséquent, les entités ayant la même valeur pour l'attribut auront la même valeur dans le nouveau champ numérique. Cela crée un équivalent numérique de l'attribut spécifié, qui définit les mêmes classes. Le nouvel attribut n'est pas ajouté à la couche source mais à une nouvelle qui est générée. En option, une table distincte peut être sortie, contenant un résumé des valeurs de champ de classe associées à la nouvelle valeur numérique unique.'

At the bottom, there is a progress bar at 0%, and buttons for 'Aide' (Help), 'Exécuter comme processus de lot...' (Execute as batch process...), 'Annuler' (Cancel), 'Fermer' (Close), and 'Exécuter' (Execute).

Créer une table avec les espèces uniques 2/2

- La table `especies` a bien été créée dans le GeoPackage `donnees_suivi_moqueur.gpkg`. On l'ouvre dans le projet
- Elle contient les champs `fid`, `id_espece` et `espece_support`
- On va conserver uniquement le `fid` et le champ `espece_support`, qu'on renomme en `nom_scientifique`, en modifiant la table depuis les propriétés de la couche, onglet `Champs`
- On ajoute un identifiant unique `uid` de type `Texte` avec la caculatrice de champ et l'expression `regexp_replace(uuid(), '[{}]', '')`

fid	nom_scientifique	uid
1	Myrcia citrifolia	2e79d82e-5bea-4750-bb47-a6c71184ed52
2	Pisonia fragans	82865a40-9ec2-4c37-b61f-89fed11a784d
3	NULL	a699ff96-b49d-43f3-8254-d1e3987ff414
4	Damburneya coriacea	bca4e8a3-b4f7-4937-8fb1-e99b9c7e35b6
5	Ouratea guildingii	9785b7b8-7de0-4876-ba57-05e72ee9210f
6	Maytenus laevigata	b4f74947-0bb8-4d30-b813-541d094975db
7	Eugenia ligustrina	cf31fbae-5f5e-413b-973c-a4027f06ef94

Ajouter une clé étrangère dans la table observations avec l'id de l'espèce

Avec la calculatrice de champs, pour la couche `observations`, on ajoute un champ `id_espece` de type `Texte` et on va le **remplir à l'aide d'une expression** basée sur:

- la méthode `get_feature` qui permet de récupérer une ligne d'une autre couche, ici `especes` en faisant la correspondance entre les valeurs (comme pour une jointure)
- la méthode `attributes` qui permet de récupérer les valeurs des champs pour cette ligne, ici pour récupérer l'`uid` créé précédemment

```
attributes(  
    get_feature(  
        'especes',  
        'nom_scientifique',  
        "espece_support"  
    )  
)['uid']
```

Ajouter une clé étrangère dans la table observations avec l'id de l'espèce

- On a bien l'`uid` de l'espèce ajouté dans le champ `id_espece`

espece_support	commentaire	servateur	auteur_uid	nb_oe	nb_pouv	nb_imma	nb_adu	longitude	latitude	date_obs	id_espece
Damburneya coria...	Progr...	DES...	2,16	NULL	NULL	NULL	NULL	7152...	1630...	2013...	bca4e8a3-b4f7-4937-8fb1-e99b9c7e
Myrcia citrifolia	Progr...	DES...	2,69	NULL	NULL	NULL	NULL	7170...	1631...	2013...	2e79d82e-5bea-4750-bb47-a6c7118

- On peut supprimer le champ `espece_support` qui ne servira plus

Récupérer automatiquement la commune de chaque observation

Les **expressions** permettent de gérer les données en relation, par exemple via la méthode `aggregate` : on va récupérer le **code INSEE** de la commune de chaque observation, par **intersection** entre leurs géométries

- on crée un nouveau champ `code_insee` de type `Texte` avec l'expression

```
aggregate(
  layer:='communes',
  aggregate:='max', expression:="code_commune",
  filter:=intersects($geometry, geometry(@parent))
)
```

- On obtient bien le nouveau champ `code_insee` dans la table

longitude	latitude	date_obs	id_espece	code_insee
714346,7338...	1630612...	2013-05-15	2e79d82...	97230
715925,6404...	1630965...	2013-06-13	NULL	97230
715523,0675...	1629879...	2013-06-18	NULL	97230
714620,8252...	1632941...	2013-06-18	2e79d82...	97228

Créer des relations entre les observations, les communes et les espèces

Dans les **propriétés du projet QGIS**, onglet **Relations**, créer

- Une relation entre les **observations** et les **communes**
- Une relation entre les **observations** et les **especes**

Propriétés du projet — Relations

	Nom	Couche de référenc...	Champ de référen...	Id de référence	Champ référençant	Id	Force
1	Obs par ...	communes	code_com...	observations	code_insee	observatio...	Association

Ajouter une nouvelle relation

Nom: Obs par espèces

Couche référencée (Parente): especes

Couche référençante (Enfant): observations

Id: [Généré automatiquement]

Force de la relation: Association

Aide Annuler OK

Visualiser via la table attributaire les données en relation

- Ouvrir la **table attributaire** des **especes**
- Passer en **vue "Formulaire"** via le petit bouton en bas à droite du tableau
- Sélectionner une **commune** et voir les données d'observations liées

communes — Total des entités: 34, Filtrées: 34, Sélectionnées: 0

Expression

- ☐ Le Morne-Vert
- ☐ Le Prêcheur
- ☐ Le Robert
- ☐ Le Vauclin
- ☐ Les Anses-d'Arlet
- ☐ Les Trois-Îlets
- ☐ Macouba
- ☐ Rivière-Pilote
- ☐ Rivière-Salée
- ☐ Saint-Esprit
- ☐ Saint-Joseph
- ☐ Saint-Pierre
- ☐ Sainte-Anne
- ☐ Sainte-Luce

Identifiant auto 32

Code INSEE 97228

Nom Sainte-Marie

▼ Obs par commune

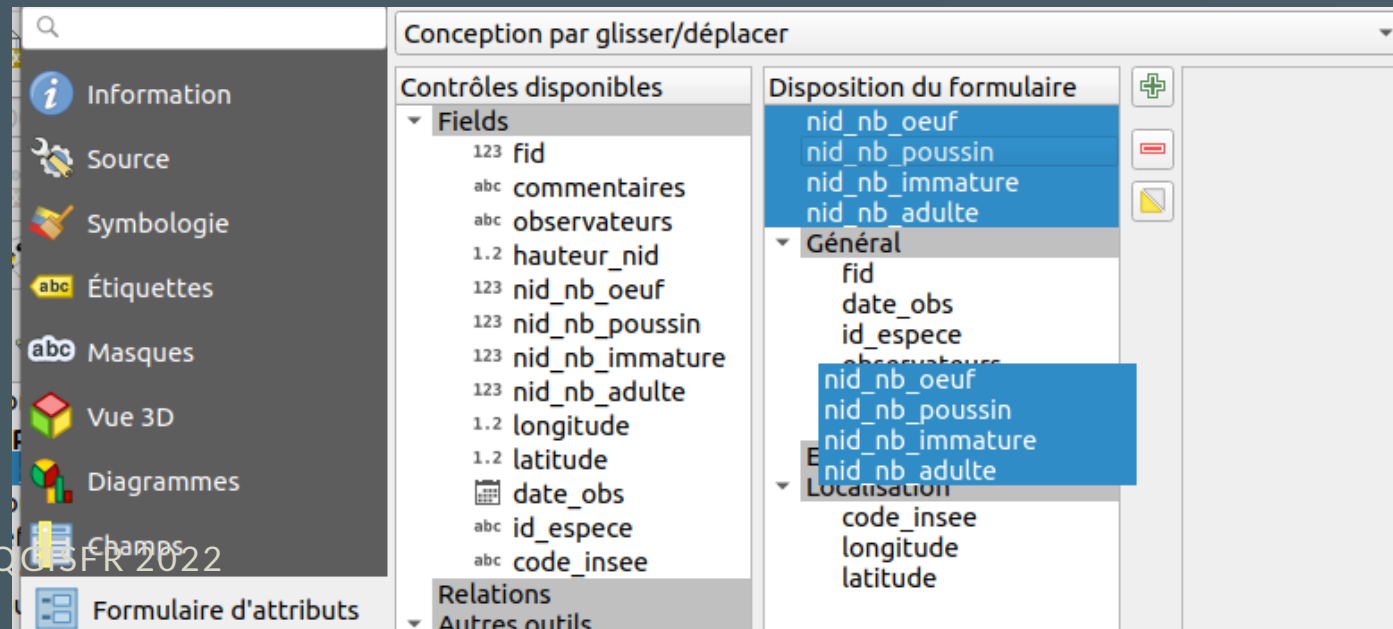
	fid	espece_suppor	commentaires	observateurs	hauteur_nid	nid_nb_oeuf	nid_nb_poussir	d_nb_i
1	3	Pisonia fra...	Programm...	NULL	3,8	NULL	NULL	NULL
2	4	NULL	Programm...	NULL	NULL	NULL	NULL	NULL
3	5	Myrcia citri...	Programm...	NULL	2,96	2	NULL	NULL
4	8	Damburne...	Programm...	ATICIEN Jé...	3,16	NULL	NULL	NULL
5	12	Myrcia citri...	Programm...	ATICIEN Jé...	3	NULL	NULL	NULL
6	13	Damburne...	Programm...	DESCHAM	2,16	NULL	NULL	NULL

QCIS 2022

Construire un formulaire de saisie

On souhaite **contrôler la saisie** des observations. On ouvre les propriétés de la couche `observations`, menu **Formulaire**.

- On passe en mode `Conception par glisser/déplacer`
- Pour chaque champ, on clique sur son nom, on ajoute un `alias`
- On **réordonne les champs** dans l'ordre souhaité.
- On peut créer des **groupes** et y déplacer les champs



Construire un formulaire de saisie

- Pour certains champs, on les marque en **non éditables**: `fid`, `longitude`, `latitude`
- On choisit pour chacun le `type d'outil` pour chaque champ:
 - **Édition de texte** pour `fid`, `commentaire`, `observateurs`
 - **Référence de la relation** pour `id_espece` et `code_insee` avec `Autoriser la valeur NULL`, `Trier par valeur` et
 - **Date/Heure** pour `date_obs`
 - **Plage** pour les champs `hauteur_nid`, `nid_nb_oeuf`, `nid_nb_poussin`, `nid_nb_immature`, `nid_nb_adulte` entre 0 et 100
- On ajoute des **contraintes**: `Non null` et `Renforcer la contrainte non null` pour les **champs obligatoires**
- On peut ajouter des contraintes spécifiques: `"date_obs" > '2000-01-01'` pour forcer une date après 2000
- On peut utiliser des expressions pour les **valeurs par défaut**: `now()` pour la date, `x($geometry)` et `y($geometry)` pour les **coordonnées**

- Disposition du formulaire
 - ▼ Observation
 - ▼ Général
 - fid
 - date_obs
 - id_espece
 - observateurs
 - commentaires
 - hauteur_nid
 - ▼ Effectifs
 - nid_nb_oeuf
 - nid_nb_poussin
 - nid_nb_immaturation
 - nid_nb_adulte
 - ▼ Localisation
 - code_insee
 - longitude
 - latitude

▼

Affichage

☒ Montrer l'étiquette

▼

Général

Alias

Date

...

Commentaire

☒ Éditable
 ☐ Étiquette au sommet

▼

Type d'outil

Date/Heure

▼

Format du champ

Date

▼

yyyy-MM-dd

Affichage

Défaut

▼

yyyy-MM-dd

☒ Calendrier
 ☒ Autoriser les valeurs NULL

Aperçu

2022-01-19

▼

▼

Contraintes

☒ Non nul
 ☒ Renforcer la contrainte non null

☐ Unique
 ☐ Renforcer la contrainte de valeur unique

Expression

"date_obs" >= '2000-01-01'

ⓧ

▼

ℰ

Description de l'expression

La date doit être au 21^è siècle

☐ Renforcer la contrainte par expression

▼

Défauts

Valeur par défaut

now()

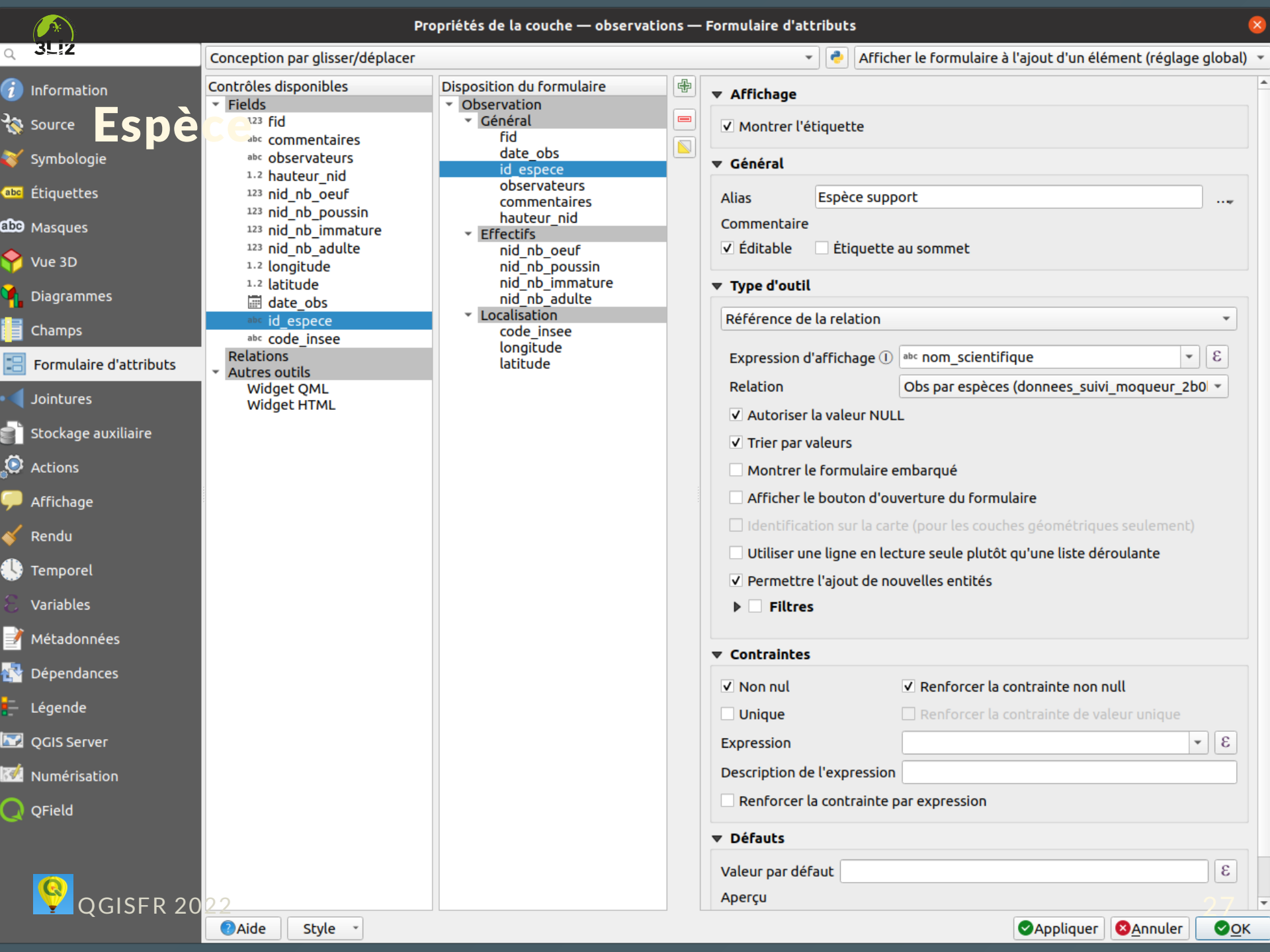
ⓧ

ℰ

Aperçu

2022-01-19

☐ Appliquer la valeur par défaut sur la mise à jour



Le formulaire de saisie

observations - Attributs d'entités

Observation

Général

Identifiant unique
60

Date
2021-11-30

Espèce support
Ardisia obovata

Observateurs
RIVIERE Georges (ACME), DULOUP Juliette (ACME)

Commentaires
NULL

Hauteur du nid
2m

Effectifs

Nombre d'oeufs
1

Nombre de poussins
NULL

Nombre d'immatures
NULL

Nombre d'adultes
2

Localisation

Commune
La Trinité

Longitude
720310,871978758

Latitude
1627986,27636426

Annuler
OK

Une table attributive mise en forme selon les valeurs des champs

observations — Total des entités: 60, Filtrées: 60, Sélectionnées: 0

abc Commentaires = [] Tout mettre à jour Mettre à jour la sélection

	re d'o	re de po	e d'imm	bre d'ad	Longitude	Latitude	Date	spèce suppo	Commune
24	0	0	0	0	719287,5353...	1629202...	2020-09-23	Pisonia f...	La Trinité
25	0	0	0	0	716186,3309...	1628888...	2020-09-23	Myrcia ci...	La Trinité
26	0	0	0	0	718663,0380...	1629327...	2020-09-23	Myrcia ci...	La Trinité
27	0	0	0	0	718154,5859...	1629292...	2020-09-23	Pisonia f...	La Trinité
28	1	0	0	0	717713,3087...	1630189...	2020-09-23	Eugenia l...	La Trinité
29	0	0	0	0	716691,1143...	1628843...	2020-09-23	Pisonia f...	La Trinité
30	0	0	0	0	718487,4614...	1628844...	2020-09-23	Maytenu...	La Trinité
31	0	0	0	0	716821,6629...	1629797...	2020-09-23	Myrcia ci...	La Trinité
32	0	0	0	0	717727,2682...	1628991...	2020-09-23	Myrcia ci...	La Trinité
33	0	0	0	0	715740,3849...	1628982...	2020-09-23	Myrcia ci...	La Trinité
34	NULL	NULL	NULL	NULL	713908,8239...	1632086...	2013-05-23		Sainte-Marie
35	0	0	0	0	719192,1736...	1632393...	2020-09-23	Myrcia ci...	
36	0	0	0	0	717832,8567...	1630705...	2020-09-23	Myrcia ci...	La Trinité
37	0	0	0	0	717359,4318...	1627665...	2020-09-23	Ardisia o...	La Trinité
38	0	0	0	0	717231,6090...	1628888...	2020-09-23	Myrcia ci...	La Trinité
39	0	0	0	0	720025,6192...	1629296...	2020-09-23	Myrcia ci...	La Trinité
40	0	0	0	0	716117,7178...	1629494...	2020-09-23	Myrcia ci...	La Trinité
41	0	0	0	0	717448,5848...	1630550...	2020-09-23	Ardisia o...	La Trinité
42	0	0	0	0	720118,2448...	1628550...	2020-09-23	Myrcia ci...	La Trinité
43	NULL	0	0	NULL	717903,6513...	1629447...	2020-09-30	Eugenia l...	La Trinité

☒ Champ Date
☐ Ligne complète

abc 123 Obs récentes
 to_int(day(age(now(), "date_obs"))) > 100

+ Nouvelle Règle

Quelques exemples d'expression 1/3

- **Couleur continue** pour un vecteur basé sur une palette de couleur
https://twitter.com/Qgis_Bzh/status/1207974981918973962
- **Rotation ou taille** qui dépend de la **position du curseur**
<https://twitter.com/kgjenkins/status/1298363142070767617>
- Lisser les courbes de niveau SRTM
<https://twitter.com/timlinux/status/1331645809834811394>
- Modifier la symbologie ou les étiquettes des **objets sélectionnés**
`is_selected() = True`
- Couleur d'étiquette basée sur la **visibilité d'une couche**
`is_layer_visible()`:
<https://twitter.com/northroadgeo/status/1208299125604638720>

Quelques exemples d'expression 2/3

- Trouver le **point le plus proche** :
<https://twitter.com/spatialthoughts/status/1421038975955791873>
 et <https://www.youtube.com/watch?v=iCgDIxUGAp0&t=113s>
- Lire les données **EXIF d'une image JPG**
<https://github.com/qgis/QGIS/commit/754328cbd0a4e5251f03c444221988a7031f4cef>
- Récupérer un tableau des éléments d'une **multi-géométrie**:
<https://twitter.com/cartocalypse/status/1386647274416181253>
- Utiliser `eval` pour créer des **expressions dans des expressions**. Ex: somme des 31 colonnes de données pour récupérer la somme par mois:

```
eval(
  array_to_string(
    array_foreach(
      generate_series(1, 31),
      concat("VALUE", @element, ' + ')
    )
  )
)
```

Quelques exemples d'expression 3/3

- Déplacer des points de manière aléatoire (floutage)

```
with_variable(  
    'distance_floutage',  
    1000,  
    translate(  
        $geometry,  
        randf(-1 * @distance_floutage, @distance_floutage),  
        randf(-1 * @distance_floutage, @distance_floutage)  
    )  
)
```


Attention

- Les expressions ne sont pas faites pour travailler sur des données lourdes: pas d'utilisation de l'index spatial par exemple.
- Si un script Processing fait la même chose, le privilégier pour modifier de la donnée (ex: recherche par intersection)

Merci de votre attention

