

API Annuaire de l'administration et des services publics
Service-public.gouv.fr

Documentation technique

Rédaction			
<i>Nom</i>	<i>Fonction</i>	<i>Date</i>	<i>Visa</i>
Direction de l'information légale et administrative		06/12/20 23	
Confidentialité			
<input checked="" type="checkbox"/> Public	<input type="checkbox"/> Interne	<input type="checkbox"/> Restreint	

Table des matières

1.	Objet du document.....	3
2.	Base URL et endpoints	3
3.	Paramètres	4
3.1.	Clauses.....	4
3.2.	Fonctions.....	4
3.2.1.	LIKE.....	4
3.2.2.	startswith()	5
3.2.3.	suggest().....	5
3.3.	Opérateurs logiques	Erreur ! Signet non défini.
4.	Cas pratiques	Erreur ! Signet non défini.
4.1.	Nom de l'organisme.....	Erreur ! Signet non défini.
4.2.	Type de service local	Erreur ! Signet non défini.
4.3.	Code INSEE commune	Erreur ! Signet non défini.
4.4.	Code postal	Erreur ! Signet non défini.
4.5.	Département.....	Erreur ! Signet non défini.
4.6.	Site internet.....	Erreur ! Signet non défini.
4.7.	SIREN/SIRET.....	Erreur ! Signet non défini.
4.8.	Région	Erreur ! Signet non défini.
4.9.	Type d'organisme	Erreur ! Signet non défini.
4.10.	Adresse.....	Erreur ! Signet non défini.
4.11.	Exemple de combinaison de fonctions et d'opérateurs logiques.....	Erreur ! Signet non défini.

1. Objet du document

Ce document a pour objectif de décrire de manière succincte les bases d'utilisation de l'API Annuaire de l'administration et des services publics au travers une description technique appuyée par des exemples et des cas d'usage.

2. Base URL et endpoints

L'URL de base de toutes les requêtes est :

`https://api-annuaire.service-public.gouv.fr/api/explore/v2.1/catalog/datasets/<id_dataset>/<endpoint>`

id_dataset doit prendre la valeur suivante : **api-annuaire-administration**

endpoint peut prendre les valeurs suivantes :

- **/records** pour la recherche de données, idéal pour développer une application et faire de la recherche rapide et paginée, pour alimenter une liste, une table etc...
À noter : cet endpoint limite le nombre de résultats à 100. Pas pertinent au-delà, sinon utiliser l'API d'export.
- **/exports/<format>** pour l'export et donc le téléchargement de données, idéal pour récupérer
 - **format** peut prendre les valeurs les plus demandées suivantes : "json" "csv" "xlsx" mais également les valeurs "geojson" "shp" "kml" "jsonl" "jsonld" "rdfxml" "turtle" "n3"

Exemple de requête d'export du jeu de données api-annuaire-administration au format json :

<https://api-annuaire.service-public.gouv.fr/api/explore/v2.1/catalog/datasets/api-annuaire-administration/exports/json>

3. Paramètres

3.1. Clauses

- **select** : permet de choisir les champs à retourner, séparés par une virgule. Si aucun champ n'est précisé, alors tous les champs disponibles seront rentrés.
- **where** : permet de filtrer la requête, via une recherche textuelle en utilisant des fonctions de recherche décrites ci-dessous
- **order-by** : permet de trier les résultats
- **limit** : permet de limiter le nombre de résultats attendus en réponse de la requête

Exemple de requête de recherche du jeu de données api-lannuaire-administration pour récupérer les 20 premiers résultats et les champs nom, adresse et pivot :

<https://api-lannuaire.service-public.gouv.fr/api/explore/v2.1/catalog/datasets/api-lannuaire-administration/records?select=nom,adresse,pivot&limit=20>

3.2. Fonctions

Dans la clause **where**, on peut utiliser des fonctions de recherche qui auront chacune leur spécificité :

3.2.1. LIKE

Syntaxe: <text_fields> LIKE <text_literal>

Permet de faire une recherche précise sur un champ. Sans approximation ou recherche préfixée.

Particularité :

La fonction LIKE ne prend pas en compte la proximité et l'ordre entre les mots. Par exemple pour la recherche :

nom LIKE "mairie Bains"

Trouvera : "Mairie - Bains", "Mairie - Bains-sur-Oust" mais aussi "Mairie - Cambo-les-Bains", "Mairie - Vals-les-Bains", "Mairie - Gamarde-les-Bains", etc...

3.2.2. **startswith()**

Syntaxe: `startswith(<text_field>|*, <text_literal>)`

Permet de rechercher **sans approximation, sans prise en compte de la racine des mots**, en début d'un champ.

Exemple de recherche préfixée ("commence par"):

`startswith(title, "SECRET")`

Permet de trouver "SECRET DEFENSE" mais ne trouve pas "THE BOOK OF SECRETS" ni "book of secret"

Particularité :

La fonction **startswith** permet d'assurer que le terme est en début de champ, et n'autorise aucune approximation, on cherche donc le terme exact.

C'est pour cette raison que "startswith" est conseillé pour la recherche par département (on recherche les 2 premiers chiffres du code INSEE)

3.2.3. **suggest()**

Syntaxe: `suggest(<text_field>|*, <text_literal>)`

Permet la recherche préfixée en se basant sur la racine du mot (Stemming). Ce qui signifie que les recherches **permettent de remonter des résultats approximatifs** :

`suggest(nom, "rennes")` va donc remonter "Rennes" mais aussi : "Renédale", "Renescure", "René", "Renève", etc...

Exemple de recherche préfixée ("commence par"):

`suggest(*, "film")`

Permet de trouver les mots film, filmer, filmographie etc... dans le jeu de données (dans tous les champs)

Particularité :

La fonction suggest permet d'assurer la précédence (proximité) des mots. Par exemple pour la recherche :

`suggest(nom, "mairie Bains")`

Trouvera : "Mairie - Bains", "Mairie - Bains-sur-Oust", etc...

Mais ne trouvera pas : "Mairie - Cambo-les-Bains", "Mairie - Vals-les-Bains", "Mairie - Gamarde-les-Bains", etc...

La recherche **suggest(nom, "Bains mairie")** ne trouvera pas d'organisme de service

⚠ Déconseillé pour les recherches exactes.

3.3. Opérateurs logiques

l'API permet la recherche via des opérateurs logiques comme OR, AND et NOT. Afin de construire des requêtes plus avancées. Toutes les fonctions vues précédemment peuvent donc être combinées via ces opérateurs.

4. Cas pratiques

Exemples de recherche pour le jeu de données **api-lannuaire-administration**

4.1. Nom de l'organisme

where=nom LIKE "Rennes"

Recherche de l'organisme dont le **nom** contient exactement le terme **"Rennes"**, affichage des 10 premiers résultats :

<https://api-lannuaire.service-public.gouv.fr/api/explore/v2.1/catalog/datasets/api-lannuaire-administration/records?where=nom%20LIKE%20%22Rennes%22&limit=10>

Exporte tous les résultats de la recherche précédente :

<https://api-lannuaire.service-public.gouv.fr/api/explore/v2.1/catalog/datasets/api-lannuaire-administration/exports/json?where=nom%20LIKE%20%22Rennes%22>

4.2. Type de service local

where=pivot LIKE "gendarmerie"

Recherche un service local dans le champs pivot :

[https://api-lannuaire.service-public.gouv.fr/api/explore/v2.1/catalog/datasets/api-lannuaire-administration/records?where=pivot LIKE "gendarmerie"](https://api-lannuaire.service-public.gouv.fr/api/explore/v2.1/catalog/datasets/api-lannuaire-administration/records?where=pivot LIKE \)

4.3. Code INSEE commune

where=code_insee_commune LIKE "31555"

Recherche un service local par son code INSEE :

https://api-lannuaire.service-public.gouv.fr/api/explore/v2.1/catalog/datasets/api-lannuaire-administration/records?where=code_insee_commune%20LIKE%20%2231555%22

4.4. Code postal

where=suggest(adresse,"code_postal 75005")

Recherche un service local par son code postal :

[https://api-lannuaire.service-public.gouv.fr/api/explore/v2.1/catalog/datasets/api-lannuaire-administration/records?where=suggest\(adresse,"code_postal 75005"\)](https://api-lannuaire.service-public.gouv.fr/api/explore/v2.1/catalog/datasets/api-lannuaire-administration/records?where=suggest(adresse,\)

Remarque : ici avec la fonction **suggest** l'ordre des mots est pris en compte, c'est-à-dire que seuls les résultats dont les termes **code_postal** et **75005** collés qui se suivent seront remontés. Et non pas les résultats qui contiendraient la valeur **75005** et le terme **code_postal** n'importe où dans le champ (string) **adresse**.

4.5. Département

where=startswith(code_insee_commune,"32")

Remarque : la fonction **startswith** permettant la recherche préfixée, c'est-à-dire dont les mots commençant par le terme recherché, cela va donc remonter les codes INSEE commençant par **32**. Et donc toutes les communes du département **32**.

4.6. Site internet

where=suggest(site_internet,"www.defense.gouv.fr")

À noter : le moteur de recherche ne découpe pas les termes par le séparateur point. Ce qui a pour impact de considérer **www.defense.gouv.fr** comme un seul et unique terme. La recherche sur une partie du domaine n'est donc pas possible. Par exemple, le terme **gouv**, ou encore **defense**, ne permettra pas de rechercher dans le domaine d'un site internet.

4.7. SIREN/SIRET

where=siren:"246900575"

Recherche un service spécifique par son SIREN :

<https://api-lannuaire.service-public.gouv.fr/api/explore/v2.1/catalog/datasets/api-lannuaire-administration/records?where=siren%3A%22246900575%22>

4.8. Région

La donnée ne contient pas de champ région, il faut donc rechercher les départements de la région :

where=startswith(code_insee_commune,"22") OR startswith(code_insee_commune, "29") OR startswith(code_insee_commune, "35") OR startswith(code_insee_commune, "56")

4.9. Type d'organisme

where=type_organisme:"Service déconcentré"

4.10. Adresse

**where=adresse_courriel LIKE "defense.gouv.fr"
where=suggest(adresse_courriel,"defense.gouv.fr")**

À noter : comme pour le site internet, il n'y a pas de découpage des termes par le séparateur point. Donc **defense.gouv.fr** doit être vu comme un seul mot. Et la recherche doit donc s'effectuer sur le nom de domaine de l'e-mail complet.

Exporter au format JSON, les champs **nom, adresse, site_internet et adresse_courriel**, de tous les organismes qui ont comme adresse de contact un e-mail en **@defense.gouv.fr** :

[https://api-lannuaire.service-public.gouv.fr/api/explore/v2.1/catalog/datasets/api-lannuaire-administration/exports/json?where=adresse_courriel LIKE "defense.gouv.fr"&select=nom,adresse,site_internet,adresse_courriel](https://api-lannuaire.service-public.gouv.fr/api/explore/v2.1/catalog/datasets/api-lannuaire-administration/exports/json?where=adresse_courriel%20LIKE%20%22defense.gouv.fr%22&select=nom,adresse,site_internet,adresse_courriel)

4.11. Exemple de combinaison de fonctions et d'opérateurs logiques

Pour combiner la recherche des types de service local en fonction du code INSEE commune (exple la mairie, la prefecture, le tgi, le dd_femmes et le bav de Bordeaux)

where=code_insee_commune LIKE "33063" and (pivot LIKE "mairie" or pivot LIKE "prefecture" or pivot LIKE "tgi" or pivot LIKE "dd_femmes" or pivot LIKE "bav")

[https://api-lannuaire.service-public.gouv.fr/api/explore/v2.1/catalog/datasets/api-lannuaire-administration/records?where=code_insee_commune%20LIKE%20%2233063%22%20and%20\(pivot%20LIKE%20%22mairie%22%20or%20pivot%20LIKE%20%22prefecture%22%20or%20pivot%20LIKE%20%22tgi%22%20or%20pivot%20LIKE%20%22dd_femmes%22%20or%20pivot%20LIKE%20%22bav%22\)&limit=20](https://api-lannuaire.service-public.gouv.fr/api/explore/v2.1/catalog/datasets/api-lannuaire-administration/records?where=code_insee_commune%20LIKE%20%2233063%22%20and%20(pivot%20LIKE%20%22mairie%22%20or%20pivot%20LIKE%20%22prefecture%22%20or%20pivot%20LIKE%20%22tgi%22%20or%20pivot%20LIKE%20%22dd_femmes%22%20or%20pivot%20LIKE%20%22bav%22)&limit=20)