

TP PHP 7 - Interrogation d'API

Décompressez l'archive déposée sur Moodle pour ce TP. Le dossier résultant contient différents fichiers à réutiliser ou à compléter. Pensez à consulter le [manuel PHP](#) et pour visualiser ce qui est attendu, ce [démonstrateur](#).

Exercice 1. API Wikipedia

Wikipedia fournit une [API](#) (complexe !) permettant d'échanger des données programmatiquement.

1. Chargez tout d'abord cet [URL](#) dans votre navigateur, analysez la chaîne de requête, et inspectez la réponse obtenue au format JSON.

L'API interprète la chaîne de requête comme suit :

- extraire (`action=query&prop=extracts`)
- au format texte (`explaintext=true`)
- les 5 premières phrases (`exsentences=5`)
- de l'introduction (`exintro=true`)
- de l'article de Wikipedia concernant "Paris" (`titles=Paris`),
- combiner ce contenu avec des méta-données et renvoyer le tout au format JSON (`format=json`).

Pour plus d'informations sur le sens et les valeurs autorisées pour ces paramètres, consultez les liens [action=query](#) et [prop=extracts](#).

2. En vous inspirant des exemples du cours, complétez le fichier `api-wikipedia.php` pour

- (1) exécuter la requête HTTP GET ci-dessus,
- (2) extraire le contenu texte de l'objet JSON renvoyé, cad. l'introduction de l'article,
- (3) et l'afficher dans la page Web tel qu'illustré en Figure 1.

Paris (/pa.ʁi/) est la capitale de la France et une collectivité à statut particulier. Divisée en vingt arrondissements, elle est le chef-lieu de la région Île-de-France et le siège de la métropole du Grand Paris. Paris est situé au centre du Bassin parisien, sur une boucle de la Seine, entre les confluent avec la Marne et l'Oise. Le site est occupé à partir du III^e siècle avant notre ère sous le nom Lutèce par le peuple gaulois des Parisii, qui donne son nom à la ville. Au début du VI^e siècle, Clovis choisit Paris comme capitale de son royaume.

FIGURE 1 – Extraction partielle de l'article "Paris" à l'aide de l'API Wikipedia

Testez ensuite l'extraction d'articles "uniques" de votre choix.

Notez que les appels d'API peuvent aussi s'exécuter (et se tester)

- (1) dans un shell avec la commande `curl`
`curl "https://fr.wikipedia.org/w/api.php?action=query&format=json&titles=Paris& ..."`
- (2) en PHP avec la [bibliothèque Curl](#) ou le paquet [guzzle](#) par exemple,
- (3) ou encore avec une application graphique telle [Postman](#).

Exercice 2. API Adresse

L'API Adresse permet d'effectuer une recherche d'adresses selon différents critères (numéro, nom de rue, lieu-dit ou commune, code postal) et de récupérer leurs données de géolocalisation. Par exemple, la requête

<https://api-adresse.data.gouv.fr/search/?q=Evariste+Galois&type=street&limit=2>

recherche 2 rues au maximum dont le nom comporte Evariste Galois, et elle renvoie cet objet JSON

```
{
  {
    "type": "FeatureCollection",
    "version": "draft",
    "features": [
      {
        "type": "Feature",
        "geometry": {
          "type": "Point",
          "coordinates": [
            -4.481212,
            48.413073
          ]
        },
        "properties": {
          "label": "rue Evariste Galois 29200 Brest",
          "score": 0.7016318181818181,
          "id": "29019_1607",
          "banId": "0058f235-127a-4124-82cc-d6976ac0bb58",
          "name": "rue Evariste Galois",
          "postcode": "29200",
          "citycode": "29019",
          "x": 147231.63,
          "y": 6838736.95,
          "city": "Brest",
          "context": "29, Finistère, Bretagne",
          "type": "street",
          "importance": 0.71795,
          "street": "rue Evariste Galois"
        }
      },
      {
        "type": "Feature",
        "geometry": {
          "type": "Point",
          "coordinates": [
            2.357755,
            47.077523
          ]
        },
        "properties": {
          "label": "Allée Evariste Galois 18000 Bourges",
          "score": 0.6974054545454544,
          "id": "18033_2025",
          "name": "Allée Evariste Galois",
          "postcode": "18000",
          "citycode": "18033",
          "x": 651268.62,
          "y": 6664339.57,
          "city": "Bourges",
          "context": "18, Cher, Centre-Val de Loire",
          "type": "street",
          "importance": 0.67146,
          "street": "Allée Evariste Galois"
        }
      }
    ],
    "attribution": "BAN",
    "licence": "ETALAB-2.0",
    "query": "Evariste Galois",
    "filters": {
      "type": "street"
    },
    "limit": 2
  }
}
```

Les deux valeurs de la propriété `coordinates` de chaque adresse trouvée correspondent à sa longitude et sa latitude. Par exemple, l'allée Evariste Galois à Bourges a pour longitude 2.357755 et latitude 47.077523.

On se propose de développer une page web permettant d'interroger cette API indirectement à partir de "clés" renseignées dans un champ de formulaire pour ensuite afficher les adresses obtenues sur une carte tel qu'illustré en Figures 2 et 3.

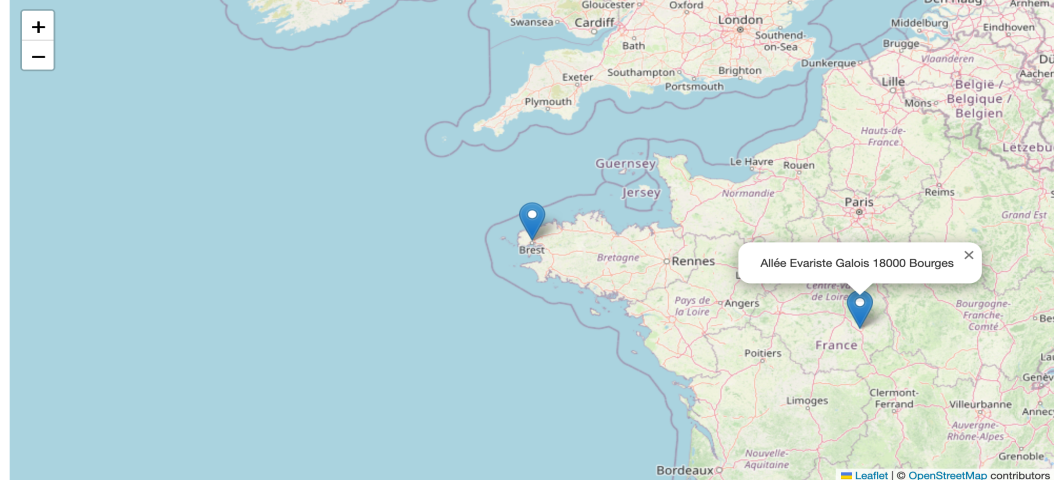


FIGURE 2
— Clés de
recherche

FIGURE 3 – Géolocalisation de deux rues "Evariste Galois"

Le fichier **cartes.php** produit le formulaire et utilise la librairie de cartes interactives **leaflet.js** pour l'affichage des adresses. Il échange aussi avec le fichier **api-adresses.php** en HTTP POST (via le code Javascript de **cartes.js**) en lui communiquant la/les clé(s) saisie(s). **api-adresses.php** doit alors interroger l'API Adresse pour extraire les adresses qui correspondent aux clés, et les filtrer pour ne renvoyer que leur libellé, longitude et latitude au format JSON. Par exemple, pour la requête précédente, **api-adresses.php** doit renvoyer le tableau JSON

```
[
  {
    "address": "rue Evariste Galois 29200 Brest",
    "lng": -4.481212,
    "lat": 48.413073
  },
  {
    "address": "Allée Evariste Galois 18000 Bourges",
    "lng": 2.357755,
    "lat": 47.077523
  }
]
```

1. Complétez le fichier **api-adresses.php** pour obtenir le comportement attendu. Dans un premier temps, codez en dur les paires `type=street` et `limit=2` de la chaîne de requête dans le fichier. Dans un second temps, extrayez-les du fichier de configuration **api-adresses.ini** à l'aide de la fonction **parse_ini_file**. Modifiez ensuite ce fichier pour re-paramétrer vos requêtes.