

Test Technique // Backend



ANNEXES

Tu trouveras en PJ deux fichiers :

- 1/ Une liste de quelques milliers d'offres d'emploi, avec pour chacune :

```
profession_id,contract_type,name,office_latitude,office_longitude  
2,FULL_TIME,Dev Full Stack,48.8768868,2.3091203
```

- 2/ La liste des "professions" (cf. "profession_id" ci-avant) :

```
id,name,category_name  
16,Développement Fullstack,Tech
```

EXERCICES / QUESTIONS

01 / 03 . Exercice : regroupements par continents

En utilisant les données en annexe, l'objectif est de développer un script qui renvoie en output le nombre d'offres par catégorie de profession par continent.

Exemple :

	TOTAL	TECH
TOTAL	1	1
EUROPE	1	1
ASIE	0	0

02 / 03 . Question : scaling ?

Maintenant si on imagine que l'on a 100 000 000 d'offres et qu'on a 1000 nouvelles offres qui arrivent par seconde. Qu'implémentes-tu en sachant que l'on souhaite le résultat de l'exercice n°1 en quasi-temps réel ?

NB: pas de code nécessaire pour cette question.

03 / 03 . Exercice : mise à disposition d'une API

On souhaite maintenant que ces données soient utilisables ! L'objectif est de développer une API, avec un unique endpoint, permettant de récupérer les offres ci-avant fonction d'une position donnée et d'un radius prédéfini.

Paramètres en entrée de cet endpoint :

- `latitude` (ex: 48.8659387)
- `longitude` (ex: 2.34532)
- `radius` (ex: 10 (km))

En output, on veut la liste des offres correspondant aux critères de recherche. Préciser pour chaque offre la proximité rapport aux coordonnées définies en entrée serait bien évidemment un plus !

FAQ

Quelles technos dois-je utiliser ?

Comme détaillé dans notre annonce, on recherche quelqu'un ayant des connaissances en Ruby / Elixir et potentiellement Go, mais surtout on recherche quelqu'un de PRAGMATIQUE et CURIEUX qui sait utiliser les bonnes technos au bon moment et pour les bons usages. Du coup, on te laisse choisir ! (et n'hésite pas à nous détailler pourquoi ce choix :))

Quel est le rendu attendu ?

On s'attend tout simplement à voir un dépôt git bien rangé avec un bel historique qui nous permettra de mieux comprendre ta réflexion.