Bases de données avancées
Projet

Modalités

Le projet est à réaliser en binômes.

Une modélisation préliminaire (avec une brève explication) doit être rendue sur Moodle le 24 mars. Elle fera l'objet d'une pré-soutenance du 24 au 26 mars. L'inscription aux pré-soutenances se fera sur Moodle et celle-ci est obligatoire. Il s'agit de s'assurer que vous avez une bonne modélisation et que vous ne suivrez pas une mauvaise direction. Elle fera partie de la notation.

Le projet sera à rendre en fin de semestre ; les modalités définitives vous seront communiquées via Moodle.

I) Présentation générale

L'objectif du projet est la modélisation, le peuplement, et la mise en place de fonctions, triggers et index adéquats pour une base de données en lien avec la pandémie de COVID-19. Le choix précis de ce que cette base de données va modéliser est laissé à chaque binôme ; on ne donnera ici que des indications, et une certaine dose d'originalité dans votre projet sera bienvenue.
Après un an d'une pandémie qui ne donne pas signe d'amélioration, il est notable que des *données ouvertes* sur le COVID-19 sont largement disponibles et utilisées dans le discours public par les journalistes, politicien·ne·s, et citoyen·ne·s. Par exemple, des sites comme `http://covidtracker.fr` ou `https://ourworldindata.org/coronavirus` qui offrent des visualisations des données sont largement repris dans la presse.

Données ouvertes

> « Les *données ouvertes* ou *open data* sont des données numériques dont l'accès et l'usage sont laissés libres aux usagers. Elles peuvent être d'origine publique ou privée, produites notamment par une collectivité, un service public, un collectif citoyen ou une entreprise. Elles sont diffusées de manière structurée selon une méthode et une licence ouverte garantissant leur libre accès et leur réutilisation par tous, sans restriction technique, juridique ou financière.
>
> L'ouverture des données est à la fois un mouvement, une philosophie d'accès à l'information, une politique publique et une pratique de publication de données librement accessibles et exploitables.
>
> Elle s'inscrit dans une tendance qui considère l'information publique comme un bien commun [. . .] dont la diffusion est d'intérêt public et général. »

Source : Article *Données ouvertes* de Wikipédia en français. Contenu soumis à la licence CC-BY-SA 3.0.

Cela signifie pour cet enseignement de bases de données avancées qu'un vaste choix de données du monde réel est disponible et peut être intégré dans la base de votre projet. Ces données concernent par exemple les nombres de cas, les nombres d'hospitalisations, de décès, de vaccinations, de livraisons de vaccins, *etc.*, en France et dans d'autres pays, avec des informations temporelles ou différents niveaux de granularité géographique. Les données sont disponibles via plusieurs plate-formes, par exemple les suivantes (mais de nombreuses autres plates-formes peuvent être trouvées par des recherches Internet) :

- sur la plateforme *open data* de la région Île-de-France `https://data.iledefrance.fr/pages/home-covid/`
- sur la plateforme ouverte des données publiques françaises `https://www.data.gouv.fr/fr/pages/donnees-coronavirus`
- sur la plateforme *our world in data* `https://ourworldindata.org/coronavirus`
- sur la plateforme de l'OMS `https://covid19.who.int`
- ...

II) Travail à effectuer

a) Travail préliminaire

Le travail pour ce projet commence par explorer les données disponibles et décider de ce que vous allez chercher à modéliser. Par exemple, après avoir parcouru `https://www.data.gouv.fr/fr/pages/donnees-coronavirus`, vous voyez qu'on dispose de données sur les personnes vaccinées, les lieux de vaccination et leurs rendez-vous, les stocks et livraisons de doses et vous pourriez décider de centrer votre projet sur la vaccination [1].

Il faut alors décider du périmètre exact de votre projet. Quels aspects seront modélisés ? Est-ce que vous intégrez des données issues de plusieurs sources ? *Etc.*

b) Modélisation

La première étape du projet consiste en une modélisation suffisamment riche pour qu'elle soit intéressante. Quand viendra sa réalisation, vous pourrez vous restreindre à une partie raisonnable de votre modèle.

Pour vous donner un ordre de grandeur de ce qui est attendu, le nombre de tables résultantes ne devrait pas dépasser 12, suivant les choix de modélisation et les simplifications entre le modèle et l'implémentation.

Cette modélisation fait l'objet d'un rendu préliminaire et est défendue lors d'une pré-soutenance.

1. Faites preuve d'originalité, il serait dommage que la moitié des projets portent sur la vaccination en France.

Rendu de modélisation

Vous devez pour le **24 mars**

1. fournir un schéma entité/relation (voir le cours de BD de L2/L3 d'Amélie Gheerbrant pour la syntaxe graphique) et un schéma relationnel (qui correspondent bien l'un à l'autre),
2. indiquer les sources des données que vous souhaitez importer (par exemple en utilisant des attributs de couleurs différentes pour identifier quelle source sera utilisée) et identifier quelles données seront au contraire écrites manuellement ou générées automatiquement,
3. identifier un maximum de contraintes (cardinalité, dépendances fonctionnelles, règles de gestion) dès la modélisation et préciser comment chacune pourra être gérée (par exemple `CHECK`, `NOT NULL`, `UNIQUE`, trigger, fonctions d'insertion/mise à jour sûre, ...) ; un récapitulatif doit être fourni.

Ces éléments peuvent tenir sur environ 3 pages, c'est ce petit rapport qu'il vous faudra déposer sur Moodle. Le nom du fichier sera `rapport_XXX_YYY.pdf` en précisant vos noms dans le nom du fichier mais aussi sur la première page au début du rapport.

Présoutenance

D'une durée courte (une dizaine de minutes), elle aura lieu du **24 au 26 mars** et devra montrer que vous avez réfléchi en amont à tous les aspects du sujet, et que vous vous êtes organisé·e·s pour la phase de développement. Elle sert aussi à rajuster la portée de votre projet si celui-ci semble trop étroit/simple ou trop large/complexe.

c) Création et indexation des tables

Après cette phase de modélisation, vous pouvez implémenter la création de vos tables SQL, en prenant soin d'intégrer vos contraintes et vos triggers le cas échéant. Définissez également les signatures des fonctions PL/pgSQL qui serviront à simplifier les opérations de gestion et les tests, ce sont par exemple des fonctions qui exécuteront des recherches, des comparaisons, des extractions de séries temporelles, *etc.*

Il faut également identifier et implémenter les index que vous poserez sur vos tables pour aider à optimiser vos requêtes les plus fréquentes.

d) Peuplement des tables

Vous devez prévoir un script entièrement automatique qui peuplera vos tables à partir de fichiers source (typiquement du texte au format CSV). Ceci peut se faire en plusieurs étapes, en créant des tables temporaires dans lesquelles vous importerez les données publiques à partir des fichiers, puis en faisant des requêtes dans ces tables pour peupler les tables de votre schéma. Des scripts peuvent également être écrits pour générer des données artificielles.

e) Gestion, tests, démonstration

Mettez en place des fonctions PL/pgSQL pour les opérations courantes : gestion (insertion et mises à jour qui maintiennent bien l'intégrité de la base), requêtes. Pensez aussi à proposer des tests pour vérifier que vos contraintes d'intégrité sont bien garanties.

En vue de la soutenance de projet, vous pouvez préparer aussi des requêtes paramétrées utilisant `PREPARE` et `\prompt`. C'est le moment de mettre en avant votre projet et de vous distinguer des autres rendus : comment exploiter vos données ? Par exemple, des comparaisons temporelles ou géographiques sur les données hospitalières, ou de vaccination, ou sur les retombées économiques de la pandémie, avec plusieurs échelles de temps ou différentes échelles géographiques (communale/départementale/nationale/européenne…) peuvent être pertinentes.

Rendu final

Votre projet doit être rendu dans une archive `projet_XXX_YYY.zip` comprenant :

- Un rapport étendant celui de la pré-soutenance. Il contiendra en plus de votre modélisation (mise à jour pour correspondre au rendu final) la description SQL de votre schéma dans la base, et des considérations générales sur le développement. Donnez aussi une liste commentée des fonctions, des triggers et des règles de gestion, et la liste et la nature des index qui seront éventuellement ajoutés ainsi qu'une justification des bénéfices attendus.
- Un fichier appelé `README` qui décrira brièvement le rôle de chacun des autres fichiers et les instructions permettant de démarrer.
- Un fichier séparé permettant de distinguer clairement l'étape de création des tables de la base appelé `create_all.sql`.
- Un fichier qui créera les triggers appelé `create_trigger.sql`.
- Un fichier `insert_data.sql` qui permettra de peupler les tables.
- Un fichier pour chaque scénario. Chacun permettra de tester comment se comporte votre base de données sur chacune des fonctionnalités, vérifier le déclenchement de chaque trigger, *etc.* Utilisez des commandes `\prompt` et `\echo` pour que l'on voie ce qu'il se passe. Chaque fichier s'appellera `test_XXX.sql`. Vous pouvez répartir les tests dans des sous-répertoires.

Certains des fichiers de scripts peuvent potentiellement utiliser d'autres langages que SQL (par exemple si vous générez automatiquement des données artificielles) ; un `Makefile` peut être une bonne solution pour télécharger des données et lancer des scripts.

Soutenance

La soutenance se déroulera par binôme, mais la notation et les questions pourront être individualisées. Vous devrez *chacun·e* maîtriser l'ensemble de ce qui est présenté, quelle que soit la façon dont vous vous êtes réparti le travail.
Ayez de quoi faire une démonstration (à vous de voir si vous commencez avec la base déjà chargée ou non), avec des scripts préparés à l'avance qui montrent chaque règle de gestion implémentée. Vous pourrez être amené·e·s à modifier une partie de ce que vous avez fait pour que nous puissions apprécier votre réactivité. Dans le barème nous évaluerons cette fluidité. Pensez à des jeux de tests significatifs, améliorez l'interface en écrivant des fonctions utiles (parfois un simple `PREPARE` suffit).