

# Les vélos de Nantes

## Détail de la partie "Programmation Orientée Objets" (R2.01)

R. Fleurquin, I. Borne, L. Naert

2022 - 2023

### 1 Architecture Modèle-Vue-Contrôleur (MVC)

L'application finale suivra une architecture logicielle "Modèle-Vue-Contrôleur" (voir cours R2.02). Cette architecture permet de définir trois modules relativement indépendants :

- Le **modèle** qui décrit le modèle de données choisi ainsi que les traitements (mathématiques par exemple) à effectuer sur les données ;
- La **vue** qui gère les éléments visibles de l'interface graphique (graphes, tableaux, boutons...) et la logique permettant d'afficher ces éléments ;
- Le **contrôleur** qui traite les actions de l'utilisateur en faisant le lien entre modèle et vue (mise à jour d'un affichage de la vue, ajout de données dans le modèle...).

En R2.01, nous travaillerons sur la partie "modèle" : vous transformerez votre base de données en code Java en utilisant une architecture orientée objets. Il s'agit ici de stocker vos données, non plus dans une base MySQL ou Oracle, mais sous forme d'objets Java de façon à pouvoir exécuter des traitements sur ces derniers.

### 2 Travaux à réaliser : Traduction BDD vers POO

1. Le diagramme de la Figure 1 (ci-dessous) est une modélisation de la base de données des vélos de Nantes. Ce diagramme UML représente donc des tables. En vous basant sur cette modélisation des tables, proposer un diagramme de classe permettant de manipuler ces différentes tables sous forme d'objets. Choisissez des types pertinents pour vos attributs<sup>1</sup>. Ajoutez des méthodes (constructeurs, getters/setters, manipulation des objets...). Chaque classe devra avoir au moins deux méthodes (hors constructeur, getter et setter).
2. Choisissez une méthode et faites le diagramme de séquence détaillé de cette méthode (ne pas représenter la manipulation des tableaux et/ou listes et les appels à `System.out.println`). Évidemment, la complexité de la méthode choisie sera prise en compte.
3. Écrire le code correspondant à l'ensemble de votre diagramme de classe. Les classes correspondantes devront être dans un package `modele` (dans lequel vous pouvez ajouter des packages si besoin).
4. Réaliser des tests de vos méthodes ainsi qu'un scénario illustrant des manipulations possibles de vos objets Java.

---

<sup>1</sup>Par exemple, les champs avec un domaine prédéfini réduits peuvent être représentés sous forme de type Enum. Une énumération en Java permet de lister le nombre fini de valeurs que peut prendre une variable. Les énumérations sont créées dans un fichier séparé, comme une classe (cf. <https://docs.oracle.com/javase/tutorial/java/java00/enum.html>)

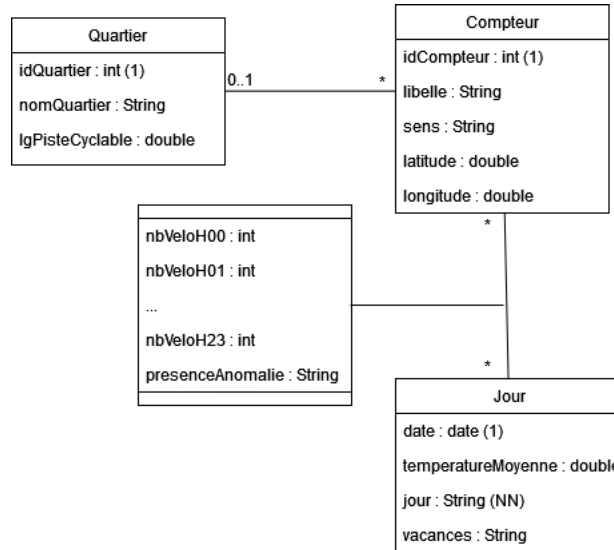


Figure 1: Diagramme de conception de la base de données

#### Contraintes et informations :

- `lgPisteCyclable` représente la longueur des pistes cyclables par quartier en mètre.
- `nbVeloHXX` correspond au nombre de vélo par heure, il y a donc 24 attributs de ce type numérotés de 00 à 23.
- `domaine(presenceAnomalie) = {Faible, Forte}`. Cette valeur peut être nulle.
- `sens` donne le sens de comptage (ex : le compteur "Coteaux vers Ouest" aura pour libellé "Coteaux" et pour sens "Ouest", "VN vers Sucé sur Erdre" aura pour libellé "VN" et pour sens "Sucé sur Erdre")
- `domaine(jour) = {Lundi, Mardi, Mercredi, Jeudi, Vendredi, Samedi, Dimanche}`
- `domaine(vacances) = {Noel, Ascension, Hiver, Ete, Toussaint, Printemps}`. Si null alors le jour ne tombe pas pendant les vacances.

## 3 Livrables

Chaque groupe doit rendre une archive contenant :

- le diagramme de classe correspondant au diagramme de conception de la partie BDD (Figure 1) avec une explication textuelle des différents choix qui ont été faits (explication des méthodes rajoutées, des associations, des types des attributs quand ceux-ci ne sont pas évidents, etc.)
- le diagramme de séquence de la méthode choisie
- le code complet de la partie "modèle" commenté en français ou en anglais
- un fichier de test et un fichier de scenario

Cette archive devra être remise au plus tard 1 semaine après votre dernière séance de R2.01 sur [l'espace Moodle de la SAE](#) en **un exemplaire** par groupe (voir la date de rendu en fonction du groupe sur l'espace Moodle). Les noms de tous les membres du groupe doivent être présents sur le rendu.