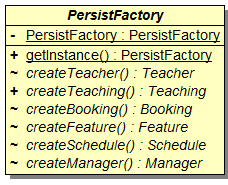
# Abstract Factory

## PersistFactory



**Brève description**

La classe PersistFactory représente la fabrique abstraite. Elle fournit une interface permettant de créer différents produits (Enseignant, Enseignement, Réservation, Caractéristique, Créneau et Manager).

**Description des attributs**

- PersistFactory : PersistFactory

: représente la seule instance de PersistFactory (singleton).

**Description des méthodes**

createTeacher() : Teacher

: renvoie un objet de type enseignant.

createTeaching() : Teaching

: renvoie un objet de type enseignement.

createBooking() : Booking

: renvoie un objet de type réservation.

createFeature() : Feature

: renvoie un objet de type caractéristique.

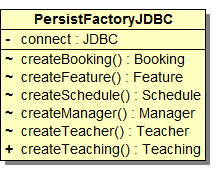
createSchedule() : Schedule

: renvoie un objet de type créneau horaire.

createManager() : Manager

: renvoie un objet de type gestionnaire.

## PersistFactoryJDBC



**Brève description**

La classe PersistFactoryJDBC hérite de la classe PersistFactory.

Le système possède des versions concrètes dérivées de la classe PersistFactory, comme par exemple

PersistFactoryJDBC. Celle-ci possède sa propre implémentation des fonctions citées ci-dessus, pouvant créer des objets tels que RéservationJDBC, CaractéristiqueJDBC, CréneauJDBC, ManagerJDBC, EnseignantJDBC, EnseignementJDBC.

Chacun de ces produits dérive d'une classe abstraite comme Teacher, Teaching…

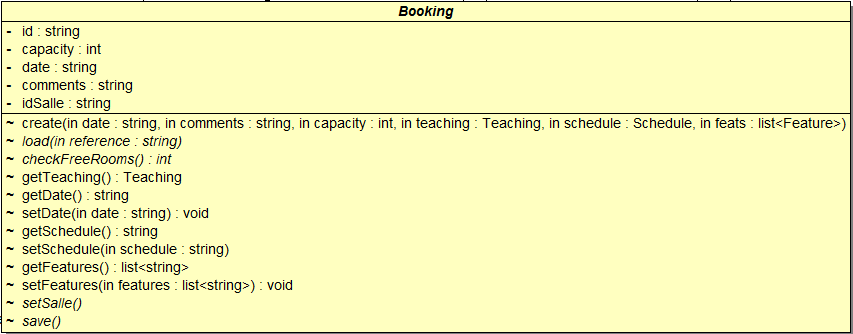
**Description des attributs**

connect

: Objet qui contient la connection à la base de données.

**Booking / Schedule / Feature**

## Booking

****

**Brève description**

La classe abstraite Booking représente la classe des réservations.

Il est important de noter qu’on ne réserve pas un créneau de 3H, mais on réserve créneau par créneau. Une réunion est une réservation qui relie un enseignement mais qui n’a pas de groupe.

**Description des attributs**

id : String

: représente l’identifiant de la réservation

capacity : int

: indique la capacité maximale en nombre de places

date : String

: indique la date à laquelle l’enseignement a lieu

comments : String

: représente les commentaires que peut faire un enseignant sur la réservation de l’enseignement

idSalle : int

: indique le numéro de la salle dans laquelle l’enseignement aura lieu

Elle possède de plus un attribut référençant un enseignement, un créneau et une liste de caractéristiques.

**Description des méthodes**

create(String date, String comments, int capacity, Teaching teaching, Schedule schedule, List<Feature> feats)

: crée une nouvelle réservation.

load(String reference) : void

: charge la réservation qui a pour identifiant reference.

checkFreeRooms() : int

: retourne le nombre de salles de disponibles en fonction des paramètres de cette réservation.

getTeaching() : Teaching

: retourne l'enseignement lié à cette réservation.

getDate() : String

: retourne la date de réservation.

setDate(String date) : void

: remplace la date de reservation.

getSchedule() : String

: retourne le créneau de réservation.

setSchedule(String schedule) : void

: remplace le créneau de réservation.

getFeatures() : List<String>

: retourne la liste des caractéristiques de la réservation.

setFeatures(List<String> features) : void

: remplace la liste des caractéristiques de la réservation.

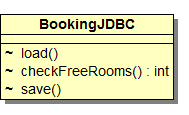
setSalle() : void

: associe une salle à la réservation.

save() : void

: sauvegarde la réservation.

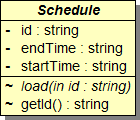
## BookingJDBC



**Brève description**

La classe BookingJDBC hérite de la classe Booking (elle hérite donc des attributs et méthodes).

## Schedule



**Brève description**

La classe abstraite Schedule représente la classe des créneaux.

Les créneaux sont fixes (il n’y a donc pas de créneaux décalés).

**Description des attributs**

id : String

: représente l’identifiant du créneau.

endTime : String

: représente le début du créneau.

startTime : String

: représente la fin du créneau.

**Description des méthodes**

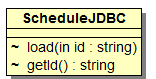
load(String id) : void

: charge le créneau avec l'identifiant correspondant.

getId() : String

: retourne l'identifiant du créneau.

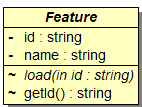
## ScheduleJDBC

****

**Brève description**

La classe ScheduleDBC hérite de la classe Schedule (elle hérite donc des attributs et méthodes).

## Feature



**Brève description**

La classe abstraite Feature représente les caractéristiques d’une salle (exemple : tableau blanc, amphithéâtre, paillasse…).

**Description des attributs**

id : String

: représente l’identifiant de la caractéristique.

name : String

: indique l’intitulé de la caractéristique.

**Description des méthodes**

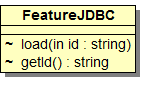
load(String id) : void

: charge la caractéristique correspondant à l'identifiant donné en paramètre.

getId() : String

: retourne l'identifiant de la caractéristique.

## FeatureJDBC

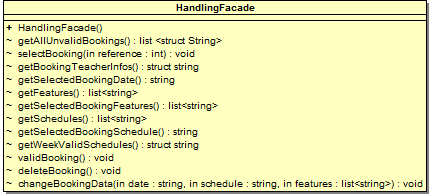
****

**Brève description**

La classe FeatureJDBC hérite de la classe Feature (elle hérite donc des attributs et méthodes).

# Handling Facade – Manager

## Handling Facade

****

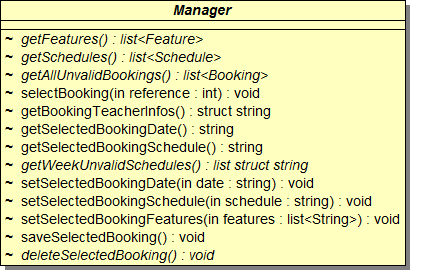
**Brève description**

**Description des attributs**

cf flèches

**Description des méthodes**

## Manager

****

**Brève description**

**Description des attributs**

cf flèches

**Description des méthodes**

getFeatures

: retourne une liste de toutes les caractéristiques existantes.

getSchedules

: retourne une liste de tous les créneaux existants.

getAllUnvalidBookings

: retourne la liste de toutes les réservations qui ne sont pas encore validées, c'est à dire pour lesquelles aucune salle n'est attribuée.

selectBooking

: écoute les événements sur le tableau, lors d'un clique sur celui-ci enregistre la réservation sélectionné puis affiche toutes ses informations.

getBookingTeacherInfos

: retourne sous la forme d'une structure composée de strings toutes les informations sur l’enseignant, c'est à dire nom, prénom, téléphone, mail et enseignement.

getSelectedBookingDate

: retourne la date de la réservation sélectionnée.

getSelectedBookingSchedule

: retourne le créneau de la réservation sélectionnée.

getWeekUnvalidSchedules

: retourne sous forme de liste de structure les dates et créneaux pour lesquelles il y aucune salle libre avec les paramètre de la réservation sélectionnée.

setSelectedBookingDate

: remplace l'ancienne date de la réservation par la nouvelle date.

setSelectedBookingSchedule

: remplace l'ancien créneau de la réservation par le nouveau créneau.

setSelectedBookingFeatures

: remplace les anciennes caractéristiques de la réservation par les nouvelles.

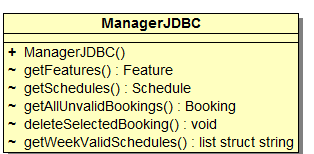
saveSelectedBooking

: sauvegarde la réservation sélectionnée et lui associe une salle.

deleteSelectedBooking

: supprime la réservation sélectionnée.

## ManagerJDBC

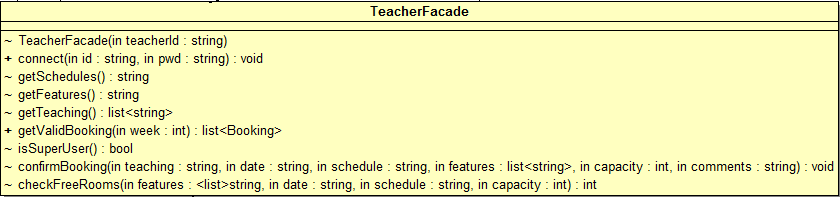
****

**Brève description**

La classe FeatureJDBC hérite de la classe Feature (elle hérite donc des attributs et méthodes).

# TeacherFaçade / Teacher / Teaching

## TeacherFaçade

****

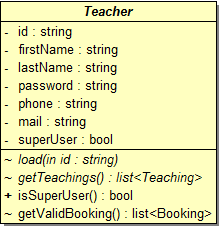
**Brève description**

**Description des attributs**

cf flèches

**Description des méthodes**

## Teacher

****

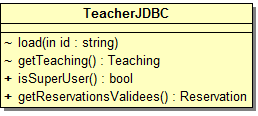
**Brève description**

**Description des attributs**

cf flèches

**Description des méthodes**

## TeacherJDBC

****

**Brève description**

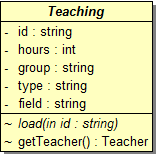
La classe TeacherJDBC hérite de la classe Teacher (elle hérite donc des attributs et méthodes).

**Description des attributs**

cf flèches

**Description des méthodes**

## Teaching



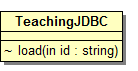
**Brève description**

**Description des attributs**

cf flèches

**Description des méthodes**

## TeachingJDBC

****

**Brève description**

La classe TeachingJDBC hérite de la classe Teaching (elle hérite donc des attributs et méthodes).

**Description des attributs**

cf flèches

**Description des méthodes**