



TP isPretEnRetard

Classe UtilitaireBookstore

Sujet: java.time, java.time.format, LocalDate, DateTimeFormatter, @test

Conseil : Pour gérer le temps, vous utilisez soit les classes historiques des package java.util, java.text, soit l'API Date-Time (JSR 310) de Java8 (package java.time, java.time.format). Un bon site avec des exemples : javatpoint.com

Enrichissez le FrameworkAfp avec la classe fr.afpa.tool.UtilitaireBookstore qui comportera les méthodes statiques qui serviront au projet bookstore.

1 La méthode statique isPretEnRetard

Dans le contexte d'une bibliothèque municipale, une personne peut emprunter un livre pendant un nombre maximum de jours ; c'est la donnée dureeMaxPret (15 jours par exemple). Au-delà de cette durée, l'emprunteur est dit « en retard ». dureeMaxPret est chargée au démarrage de l'appliquatif à partir de la base de données. Les calculs se font au jour près.

Vous développez une méthode statique qui sera appelée au moment de la restitution du livre et qui renverra un booléen indiquant si la personne rend le livre en retard (booléen) ou non.

Voici la signature de la méthode testable:

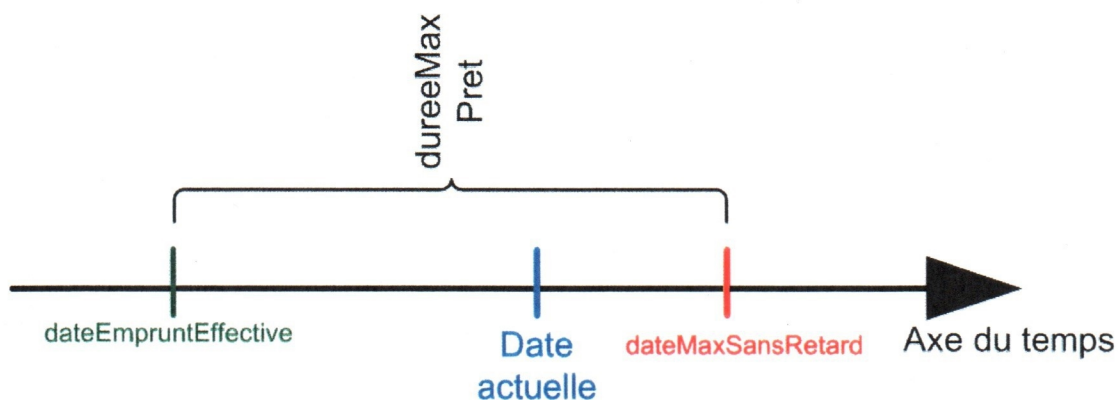
```
static boolean isPretEnRetard(LocalDate dateEmpruntEffective,
    LocalDate maintenant, int dureeMaxPret)
```

Conseils : Il y a des 2 logiques possibles

Première logique:

Vous instanciez une date actuelle et vous lui ajoutez (plusDays) le nombre de jour maximum pour le prêt (dureeMaxPret) afin d'obtenir la date limite maximum au-delà de laquelle, on est considéré comme en retard.

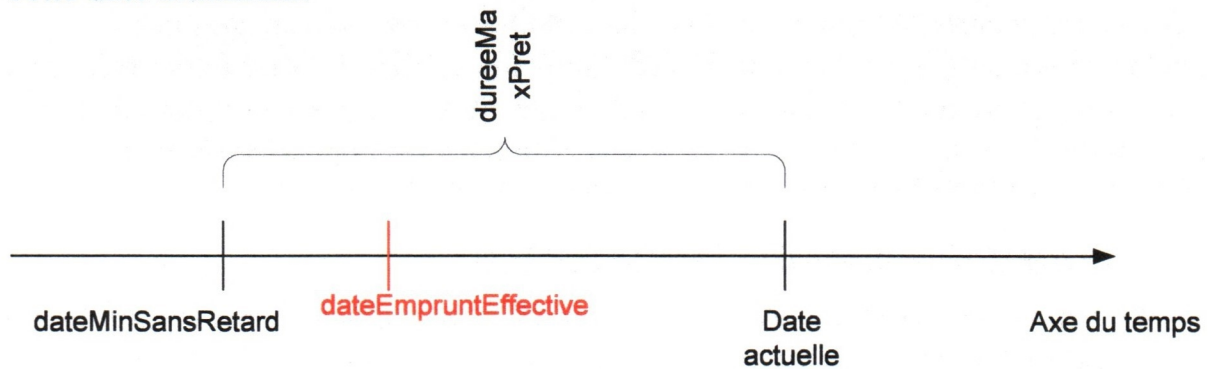
Vous comparez (isBefore ou isAfter) la date actuelle avec cette date limite.



Deuxième logique:

Vous instanciez une date actuelle et vous lui retirez (`minusDays`) le nombre de jour maximum pour un prêt (`dureeMaxPret`) afin d'obtenir la date minimum d'emprunt pour ne pas être considéré en retard.

Vous comparez (`isBefore` ou `isAfter`) la date effective d'emprunt du livre avec cette date minimum.



2 Tests unitaires (avec ou sans JUnit)

Pour ceux qui avancent vite, codez des `@tests` unitaires pour les cas normaux et les cas aux limites.