

# Gestion Bibliothèque

RAPPORT DE PROJET JAVA

Debusschere Mathieu  
Barreteau Marwan

## Table des matières

I.	Introduction.....	2
	UML du projet .....	3
II.	Utilisation du programme.....	4
	Interface .....	4
	Règles de modification et possibilités. ....	6
III.	Présentation des classes .....	8
	Paquet Modele.....	8
	Paquet Gestion .....	8
	Paquet Graphic .....	8
IV.	Algorithmes (pseudocode).....	9
	A.    ProlongationPret .....	9
	B.    CalculerRemboursement .....	10
	C.    requetePret .....	10
	D.    ChargerDocument.....	11

## Tables des illustrations

Figure 1 : Première ouverture.....	4
Figure 2 : Barre d'outils (fenêtre contextuelle) <sup>2w</sup> .....	4
Figure 3 : Exemple de retour (ligne souligné) .....	5
Figure 4 : Sauvegarde des données.....	5
Figure 5 : Augmenter Quota Exemple.....	6
Figure 6 : Avant Prolongation .....	6
Figure 7 : Après Prolongation .....	7
Figure 8 : Pénalité en cas de perte .....	7

## I. INTRODUCTION

Dans l'objectif d'apprendre, ce projet consiste en la conception et le développement d'une application de gestion pour un centre de documentation universitaire. L'objectif est de fournir un outil doté d'une interface graphique, permettant de piloter l'inventaire, les usagers et prêts.

### 1. Données de la bibliothèque

- **Documents** : Distinction entre **Livres** (caractérisés par un auteur, un taux de remboursement et un stock variable) et **Périodiques** (existant en exemplaire unique, identifiés par leur numéro et année de parution).
- **Lecteurs** : Distinction entre **Enseignants** (avec téléphone de bureau) et **Étudiants** (avec adresse postale). Chaque catégorie possède ses propres quotas de prêt et délais initiaux, bien que ces paramètres soient modifiables individuellement.

### 2. Fonctionnalités attendues

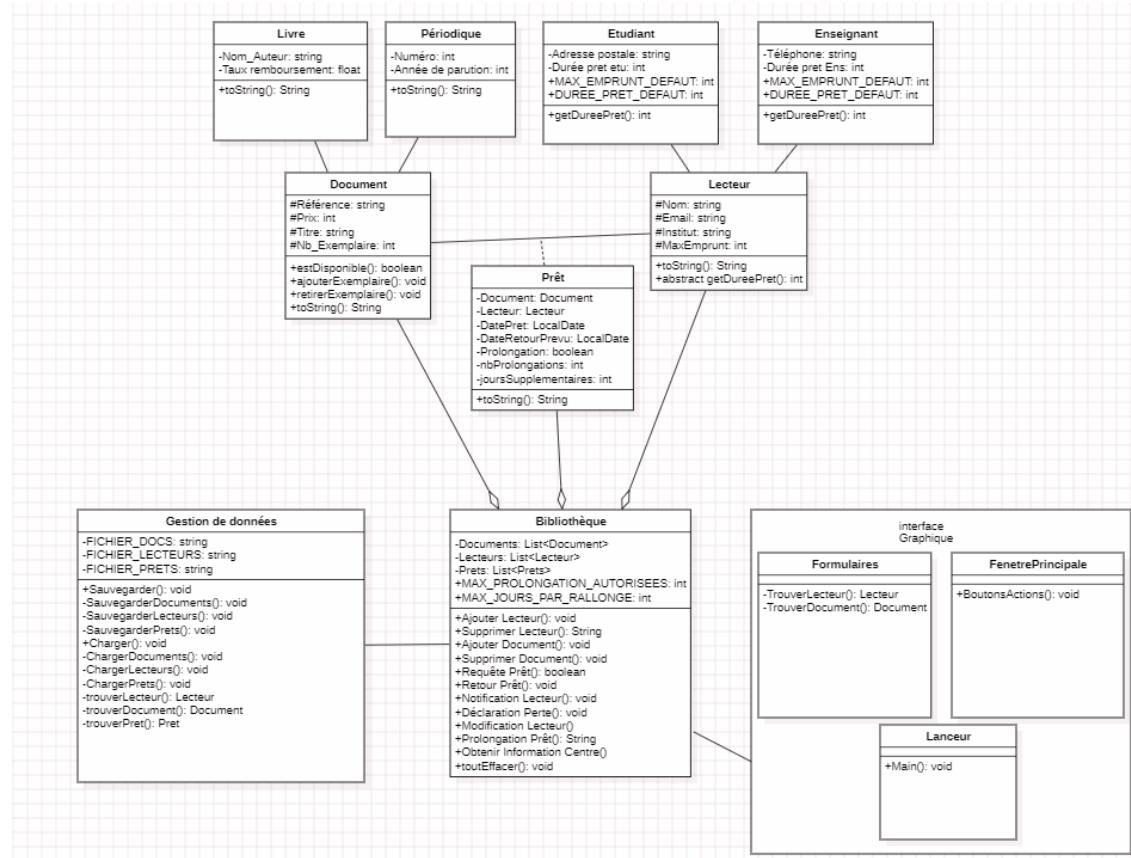
L'application doit assurer les cycles de gestion suivants :

- **Gestion des données** : Ajout, suppression et affichage des documents et des lecteurs.
- **Gestion des transactions** : Enregistrement des emprunts (avec vérification des stocks et des quotas) et des retours.
- **Suivi et alertes** : Identification des retards de restitution et traitement des prolongations de prêt.
- **Gestion des pertes** : Calcul automatique des frais de remboursement selon la nature du document (prix fixe pour les périodiques, prix majoré pour les livres).
- **Persistiance** : Sauvegarde intégrale et récupération des données via des fichiers texte.

Le programme est structuré en 3 paquets distincts :

- **Gestion** : Les classes Bibliothèques et GestionDonnees y sont incluses. Elles ont la charge de la persistiance et des sauvegardes des données.
- **Graphic** : Toutes les classes permettant l'affichage graphique y sont incluses.
- **Modele** : Tous les constructeurs et données à insérer dans la bibliothèque sont présentes dedans

## UML du projet



## II. Utilisation du programme

L'interface du logiciel a été pensée de manière simple. Au lancement avec la classe Lanceur, les données sont automatiquement récupérées des fichiers textes (Documents, Lecteurs, Prets) à la racine. Si inexistant, il est possible d'ajouter manuellement avec l'application les données et de les sauvegarder, les fichiers seront alors automatiquement générés.

### Interface

- Navigation et consultation** : L'utilisateur navigue entre trois onglets principaux permettant de visualiser en temps réel l'état du stock de documents, la liste des lecteurs inscrits et l'ensemble des prêts en cours.

The screenshot shows a window titled "La tour de Babel !". The menu bar includes "Ajouter Document", "Modifier Document", "Ajouter Lecteur", "Modifier Lecteur", "Supprimer", "Nouveau prêt", "Prolonger prêt", "Retourner Document", "Déclarer Perte", and "Voir Retards". Below the menu is a toolbar with three buttons: "Documents", "Lecteurs", and "Prêts en cours". A table displays the following data:

Type	Ref	Titre	Prix	Stock	Info	Taux/Année
Livre	LIV_002	Le Seigneur des Anneaux	35.0€	3	J.R.R. Tolkien	30.000002%
Livre	LIV_003	Apprendre Java en 21 ...	40.0€	2	Michel Martin	10.0%
Périodique	PER_100	Science et Vie	6.0€	1		2023
Périodique	PER_101	Le Monde Informatique	5.0€	1		2024
Livre	LIV_004	Les Misérables	15.0€	4	Victor Hugo	50.0%
Périodique	PER_102	National Geographic	8.0€	1	500	2022
Livre	LIV_005	Clean Code	45.0€	4	Robert C. Martin	15.000001%
Périodique	PER_103	Vogue	7.0€	1	854	2024
Livre	LIV_006	1984	12.0€	6	George Orwell	25.0%
Livre	LIV_007	Crimes et Châtiment	5.0€	6	Fiodor Dostoïevski	25.0%

At the bottom are two buttons: "Par défaut" and "Sauvegarder".

Figure 1 : Première ouverture

- Barre d'outils** : La barre d'outils supérieure centralise les actions de création de données et de transactions (Nouveau prêt par exemple). Ces actions ouvrent des fenêtres de dialogue interactives qui guident la saisie et valident immédiatement la cohérence des données.

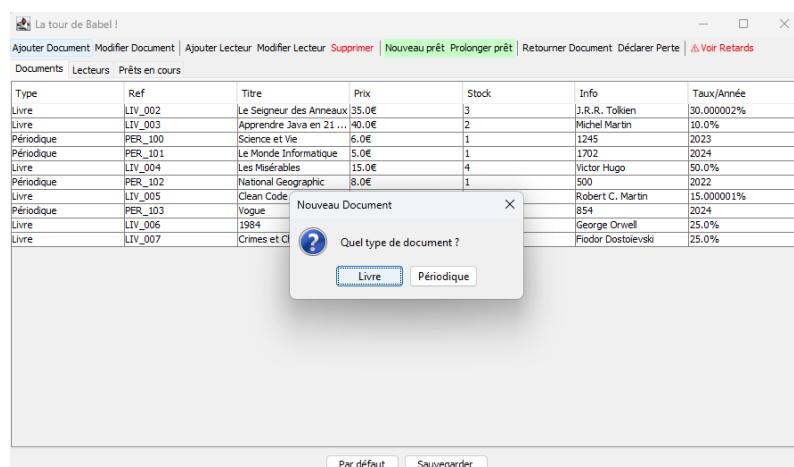


Figure 2 : Barre d'outils (fenêtre contextuelle)<sup>2w</sup>

- **Gestion et maintenance** : Une simple sélection d'une ligne du tableau de bord et un click d'un bouton permet d'effectuer un retour, de déclarer une perte, de modifier un document ou d'accéder au système de prolongation dynamique. Un bouton d'alerte spécifique offre une vue immédiate sur les retards de restitution, facilitant le suivi quotidien de la bibliothèque.

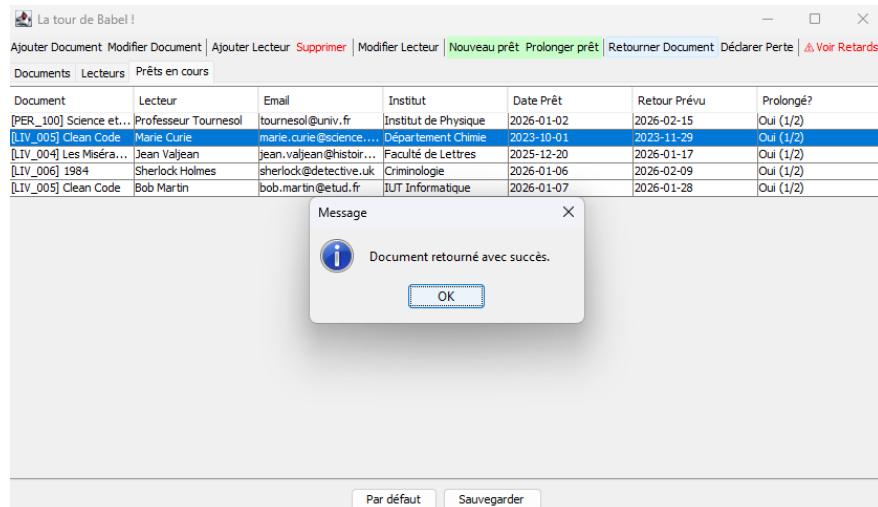


Figure 3 : Exemple de retour (ligne souligné)

- **Persistance des données** : Le bouton sauvegarder permet d'enregistrer les nouvelles données dans nos fichiers textes. Le bouton par défaut permet de revenir aux données depuis la dernière sauvegarde (entre deux sessions de logiciels).

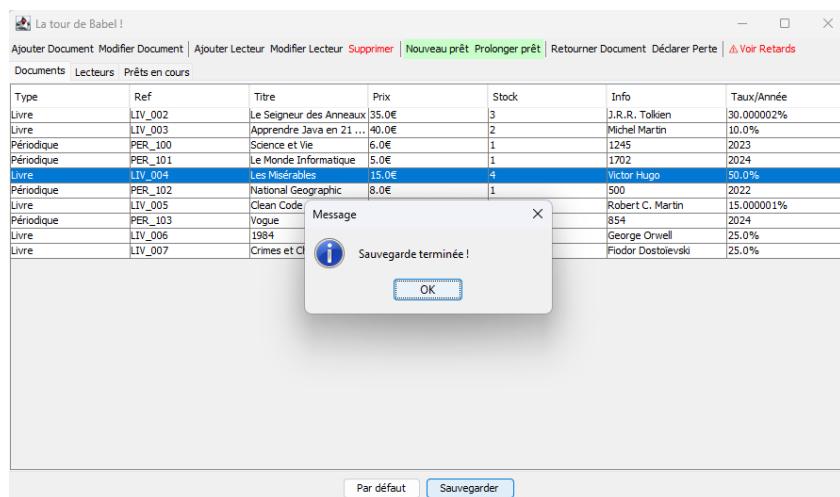


Figure 4 : Sauvegarde des données

## Règles de modification et possibilités.

Il est possible de régler dynamiquement plusieurs paramètres concernant tant bien les lecteurs que les documents.

### Système de quotas et durée.

Par défaut, le système applique des règles différencierées selon le profil du lecteur :

- **Étudiants** : Disposent initialement d'un quota de 1 documents et d'une durée de prêt de 14 jours.
- **Enseignants** : Bénéficient d'un quota de 5 documents et d'une durée de 20 jours.

Toutefois, ces valeurs ne sont pas définitives. Via l'interface de modification, l'administrateur peut ajuster ces plafonds pour un lecteur spécifique (par exemple, augmenter le quota d'un élève ponctuel).

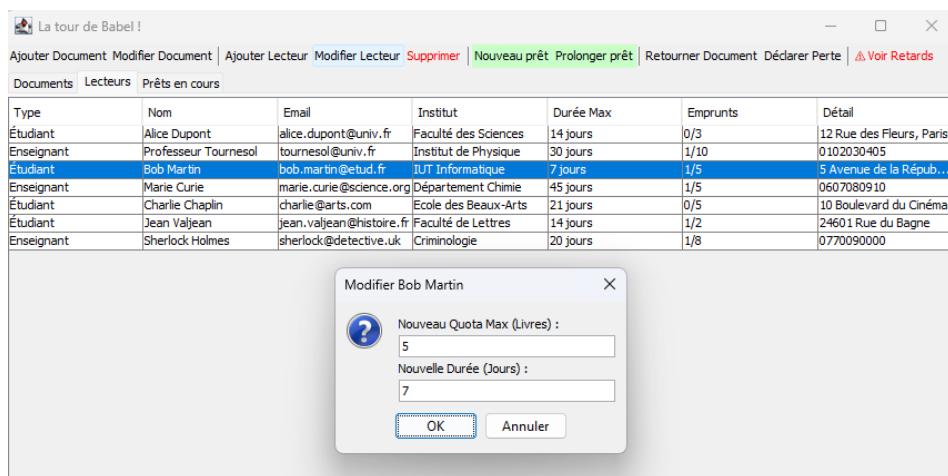


Figure 5 : Augmenter Quota Exemple

### Prolongations intelligentes (Bonus)

La méthode de prolongation ne se contente pas seulement d'ajouter des jours, elle valide chaque requête selon trois critères définis :

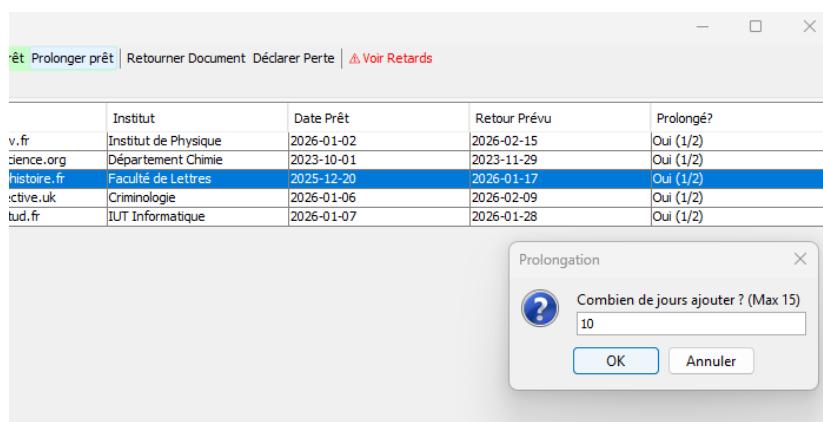


Figure 6 : Avant Prolongation

- Limite par opération** : Le nombre de jours ajoutés lors d'une seule demande ne peut excéder un plafond de sécurité (fixé à 15 jours).
- Quota de répétition** : Chaque prêt possède son propre compteur de prolongations. Une fois la limite atteinte (fixé à 2), le système verrouille et empêche toute nouvelle demande pour ce prêt spécifique.
- Cumul du temps** : La nouvelle date de retour est calculée en additionnant la durée initiale et le cumul de tous les jours supplémentaires accordés.

The screenshot shows a software interface for managing loans. At the top, there are buttons for 'Prolonger prêt' (Extend loan), 'Retourner Document' (Return document), 'Déclarer Perte' (Declare loss), and 'Voir Retards' (View delays). Below this is a table listing five loans:

	Institut	Date Prêt	Retour Prévu	Prolongé?
r	Institut de Physique	2026-01-02	2026-02-15	Oui (1/2)
nce.org	Département Chimie	2023-10-01	2023-11-29	Oui (1/2)
toire.fr	Faculté de Lettres	2025-12-20	2026-01-27	Oui (2/2)
ve.uk	Criminologie	2026-01-06	2026-02-09	Oui (1/2)
.fr	IUT Informatique	2026-01-07	2026-01-28	Oui (1/2)

Figure 7 : Après Prolongation

### Calcul des pénalités de pertes

Le système applique une règle de calcul différente selon la nature du document

- Périodiques** : Remboursement à la valeur de l'oeuvre (prix d'achat).
- Livres** : Remboursement du prix d'achat majoré d'un taux différent selon chaque ouvrage (à définir lors de l'ajout du livre).
- Déclaration de Perte** : Lors de la déclaration, une fenêtre contextuelle apparaîtra avec le prix à payer.

The screenshot shows a software interface for managing documents. At the top, there are buttons for 'Ajouter Document' (Add document), 'Ajouter Lecteur' (Add reader), 'Supprimer' (Delete), 'Modifier Lecteur' (Modify reader), 'Nouveau prêt' (New loan), 'Prolonger prêt' (Extend loan), 'Retourner Document' (Return document), 'Déclarer Perte' (Declare loss), and 'Voir Retards' (View delays). Below this is a table listing five documents:

Document	Lecteur	Email	Institut	Date Prêt	Retour Prévu	Prolongé?
[PER_100] Science et Vie	Professeur Tournesol	tournesol@univ.fr	Institut de Physique	2026-01-02	2026-02-15	Oui (1/2)
[LIV_005] Clean Code	Marie Curie	marie.curie@science.org	Département Chimie	2023-10-01	2023-11-29	Oui (1/2)
[LIV_004] Les Misérables	Jean Valjean	jean.valjean@histoire.fr	Faculté de Lettres	2025-12-20	2026-01-17	Oui (1/2)
[LIV_006] 1984	Sherlock Holmes	sherlock@detective.uk	Criminologie	2026-01-06	2026-02-09	Oui (1/2)
[LIV_005] Clean Code	Bob Martin	bob.martin@etud.fr	IUT Informatique	2026-01-07	2026-01-28	Oui (1/2)

A message dialog box titled 'Message' is displayed, containing the text 'PERTE DÉCLARÉE.' and 'Jean Valjean doit payer : 22,5 €'. There is an 'OK' button at the bottom of the dialog.

Figure 8 : Pénalité en cas de perte

### III. PRESENTATION DES CLASSES

Comme présenté dans l'introduction, le projet est regroupé en trois paquets différents. Cette section permet de présenter de manière synthétique l'ensemble des classes, par paquets.

#### Paquet Modele

**Document (classe abstraite)** : Définit le socle commun de tout ouvrage (référence, titre, prix, stock). Elle possède deux classes filles, permettant de définir selon le type d'ouvrage :

- **Livre** : Étend les attributs de Document en y ajoutant le nom d'auteur et le taux de remboursement spécifique pour les cas de perte.
- **Periodique** : Étend les attributs de Document pour les publications sérielles, incluant un numéro et une année de parution.

**Lecteur (classe abstraite)** : Définit l'identité de l'usager (nom, email, institut) et ses droits d'emprunt (quota, durée, nombre de prolongations autorisées). Elle possède deux classes filles, permettant de définir selon le type de lecteur :

- **Etudiant** : Spécialise le lecteur en ajoutant une adresse postale.
- **Enseignant** : Spécialise le lecteur en ajoutant un numéro de téléphone.

**Pret** : Créer le lien entre un Lecteur et un Document. Elle stocke la date d'emprunt et suit l'historique des prolongations (nombre de fois et jours cumulés).

#### Paquet Gestion

Ce paquet permet de gérer la persistance et les chargements des données :

- **Bibliotheque** : Agit comme le contrôleur central. Elle gère les listes en mémoire, valide les règles d'emprunt (vérification des stocks et des quotas) et traite les retours ou les pertes.
- **GestionDonnees** : Responsable de la lecture et de l'écriture des fichiers textes. Elle assure la transformation des données brutes en objets Java et vice-versa lors de la sauvegarde.

#### Paquet Graphic

Ce paquet assure l'interaction entre l'utilisateur et le système via la bibliothèque Swing.

- **FenetrePrincipale** : Constitue l'interface graphique majeure. Elle présente les données sous forme de tableaux et centralise les boutons d'action (retour, perte, prolongation).
- **Formulaires** : Fournit des fenêtres de dialogue statiques pour la saisie de nouveaux lecteurs, documents ou prêts, garantissant une séparation entre l'affichage principal et la saisie.

- **Lanceur** : Contient la méthode main qui démarre l'application.

## IV. ALGORITHMES (PSEUDOCODE)

Cette partie présente certains algorithmes sous forme de pseudocode, afin de mieux expliquer le processus et leur fonctionnement.

### A. ProlongationPret

```

ALGORITHME ProlongationPret
ENTRÉES : Lecteur L, Document D, Entier JoursDemandes
SORTIE : Message de confirmation ou d'erreur (String)

DEBUT
// 1. On commence par vérifier les entrées (Sécurité)
SI JoursDemandes <= 0 ALORS
    RETOURNER "REFUS : Nombre de jours invalide"
FIN SI

// 2. Vérification du plafond de jours
SI JoursDemandes > MAX_JOURS_PAR_RALLONGE ALORS
    RETOURNER "REFUS : Limite de jours dépassée"
FIN SI

// 3. Recherche du prêt concerné (actif)
POUR CHAQUE Pret P DANS ListeDesPrets FAIRE

    // Identification unique par Email et Référence
    SI (P.Lecteur.Email == L.Email) ET (P.Document.Reference == D.Reference)
    ALORS

        // 4. Vérification du quota de répétition
        SI P.NbProlongations >= MAX_PROLONGATIONS_AUTORISEES ALORS
            RETOURNER "REFUS : Quota de prolongations atteint"
        FIN SI

        // 5. Application de la prolongation
        P.AjouterProlongation(JoursDemandes)
        RETOURNER "SUCCÈS"

    FIN SI
FIN POUR

// Si on trouve aucun prêt
RETOURNER "ERREUR : Prêt introuvable"
FIN

```

## B. CalculerRemboursement

ENTRÉE : Le Document D (Objet)  
SORTIE : Montant à payer (Réel)

DEBUT

VARIABLE Montant : Réel

// Vérification du type dynamique de l'objet

SI D EST DE TYPE Livre ALORS

// On cast en java pour accéder aux méthodes de Livre

VARIABLE L : Livre ← (Livre) D

// Formule : Prix + (Prix \* Taux de majoration)

Montant ← L.getPrix() + (L.getPrix() \* L.getTauxRemboursement())

SINON SI D EST DE TYPE Periodique ALORS

// Pour un périodique, pas de majoration

Montant ← D.getPrix()

SINON

// Cas par défaut (Sécurité)

Montant ← D.getPrix()

FIN SI

RETOURNER Montant

FIN

## C. requetePret

ENTRÉES : Lecteur L, Document D  
SORTIE : Booléen (Vrai si succès, Faux si échec)

DEBUT

// 1. Vérification de la disponibilité matérielle

SI D.NombreExemplaires <= 0 ALORS

AFFICHER "Erreur : Document en rupture de stock"

RETOURNER FAUX

FIN SI

// 2. Vérification des droits administratifs (Quota)

// On compte combien de prêts ce lecteur a déjà en cours

VARIABLE NbPretsActuels : Entier ← CompterPretsEnCours(L)

SI NbPretsActuels >= L.MaxEmpruntsAutorises ALORS

AFFICHER "Erreur : Quota d'emprunt atteint pour ce lecteur"

```
RETOURNER FAUX  
FIN SI
```

```
// 3. Enregistrement de la transaction (Si tout est OK)  
CRÉER un nouvel objet Pret(L, D, DateDuJour)  
AJOUTER ce prêt à la ListeDesPrets
```

```
// 4. Mise à jour de l'inventaire  
D.NombreExemplaires ← D.NombreExemplaires - 1
```

```
AFFICHER "Succès : Prêt validé"  
RETOURNER VRAI  
FIN
```

## D. ChargerDocument

```
ENTRÉE : La Bibliothèque Biblio  
SORTIE : Aucune (Mis à jour interne)
```

```
DEBUT  
Ouvrir le Fichier F à l'adresse "Documents.txt"
```

```
// 1. Vérification d'existence  
SI F n'existe pas ALORS  
    RETOURNER // On arrête, la liste reste vide  
FIN SI
```

```
// 2. Lecture et Traitement  
ESSAYER  
TANT QUE le Fichier a une ligne suivante FAIRE  
    Ligne ← LireLigneSuivante(F)
```

```
// Découpage de la chaîne (Séparateur ";")  
Tableau Infos[] ← Decouper(Ligne, ";")
```

```
// Extraction des données communes  
Type ← Infos[0]  
Ref ← Infos[1]  
Titre ← Infos[2]
```

```
// Conversion: String -> Float -> Int  
Prix ← ConvertirEnEntier(ConvertirEnDecimal(Infos[3]))  
Stock ← ConvertirEnEntier(Infos[4])
```

```
// Crédit de l'objet dynamique (Livre ou Périodique)  
SI Type est "LIVRE" ALORS  
    Auteur ← Infos[5]  
    Taux ← ConvertirEnDecimal(Infos[6])
```

NouveauLivre L ← Créer Livre(Ref, Titre, Prix, Stock, Auteur, Taux)  
Biblio.AjouterDocument(L)

SINON SI Type est "PERIODIQUE" ALORS  
    Numero ← ConvertirEnEntier(Infos[5])  
    Annee ← ConvertirEnEntier(Infos[6])

NouveauPeriodique P ← Créer Periodique(Ref, Titre, Prix, Numero, Annee)  
    P.setStock(Stock)  
    Biblio.AjouterDocument(P)  
FIN SI

FIN TANT QUE

Fermer le Scanner

ATTRAPER Erreur  
    AFFICHER "Erreur lecture documents : " + MessageErreur  
FIN ESSAYER  
FIN