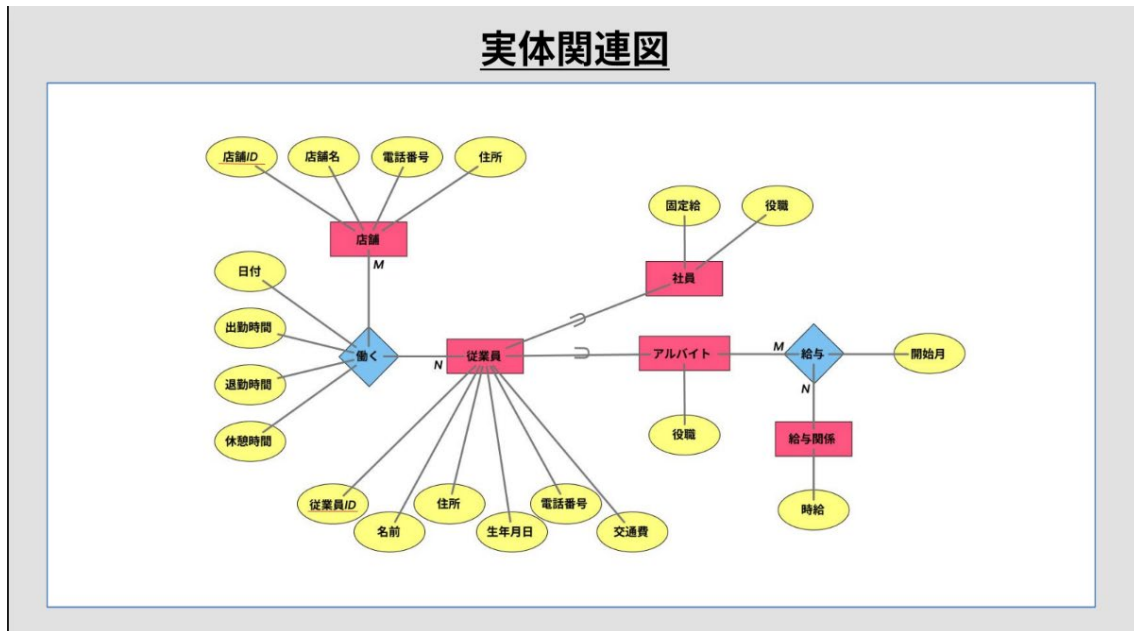


・DB(データベース)の設計書



・データベーススキーマ

店舗(店舗 ID,店舗名,電話番号,住所)

店舗(char,char,char,char)

従業員(従業員 ID,名前,電話番号,住所,生年月日,交通費)

従業員(char,char,char,char,char,int)

アルバイト(従業員 ID,役職)

アルバイト(char,char)

社員(従業員 ID,固定給,役職)

社員(char,int,char)

給与体系(時給)

給与体系(int)

働く(店舗 ID,従業員 ID,日付,出勤時間,退勤時間,休憩時間)

働く(char,char,int,int,int,int)

給与(社員 ID,開始年月)

給与(char,int)

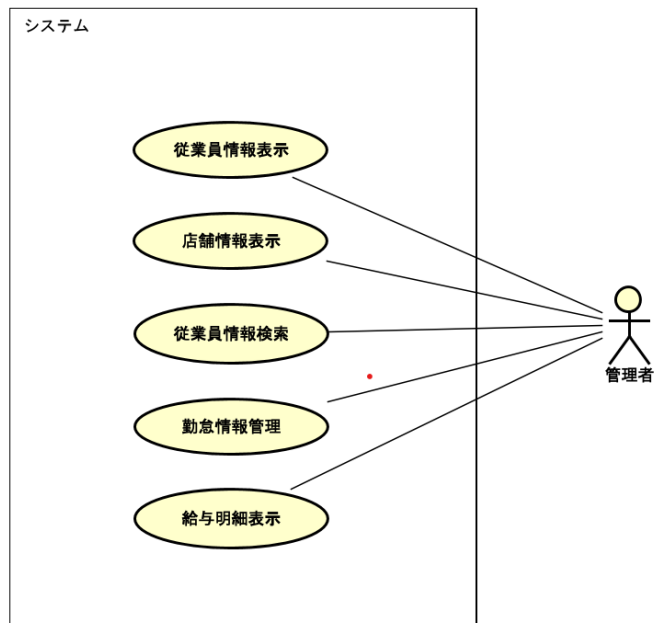
・データベースを作成する際に用いたテーブル一覧

テーブル一覧

shop(店)	shopID(店舗ID)	shopName(店舗名)	phoneNumber(電話番号)	address(住所)		
	CHAR(7)	CHAR(20)	CHAR(15)	CHAR(20)		
staff(従業員)	staffID(従業員ID)	staffName(名前)	phoneNumber(電話番号)	address(住所)	birthDate(生年月日)	fare(交通費)
	CHAR(7)	CHAR(15)	CHAR(15)	CHAR(20)	CHAR(10)	int
ptj(アルバイト)	staffID(従業員ID)	position(役職)				
	CHAR(7)	CHAR(7)				
employee(社員)	staffID(従業員ID)	position(役職)	salary(時給)			
	CHAR(7)	CHAR(7)	int			
wage(給料関係)	hourly(時給)					
	int					
work(働く)	shopID(店舗ID)	staffID(従業員ID)	Date(日付)	start(出勤時間)	end(退勤時間)	break(休憩時間)
	CHAR(7)	CHAR(7)	int	int	int	int
payment	staffID	startDate	hourly			
	CHAR(7)	int	int			
	(従業員ID)	(開始月)	(時給)			

・アプリケーションの仕様書

・ユースケース図



・ユースケース記述

項目	内容
ユースケース	従業員情報表示
概要	全従業員情報を表示(アルバイト版と社員版の2種類)
アクター	管理者
事前条件	
事後条件	
基本系列	1.メニューから選択 2.全アルバイト情報、全社員情報を表示
代替系列	
例外系列	
サブユースケース	

項目	内容
ユースケース	従業員情報検索
概要	管理者が従業員情報を呼び出す
アクター	管理者
事前条件	
事後条件	
基本系列	1. 従業員情報検索メニューの立ち上げ 2. 管理者が従業員の名前を入力する(部分一致) 3. 該当するすべての従業員情報を表示する
代替系列	
例外系列	存在しない名前が入力されたとき
サブユースケース	

項目	内容
ユースケース	店舗情報表示
概要	全店舗の情報を表示
アクター	管理者
事前条件	
事後条件	
基本系列	1.メニューから選択 2.全ての店舗情報を表示
代替系列	
例外系列	
サブユースケース	

項目	内容
ユースケース	給与明細表示
概要	管理者が給与明細を表示させる
アクター	管理者
事前条件	
事後条件	
基本系列	1. メニューの立ち上げ 2. 管理者が従業員IDと年月を指定(入力)する 3. 指定従業員の指定時期の明細(月給、今年の合計金額、扶養控除までの金額)が発行される
代替系列	
例外系列	存在しない従業員、存在しない年月が入力されたとき
サブユースケース	

項目	内容
ユースケース	勤怠情報管理
概要	管理者が勤怠情報を呼び出す
アクター	管理者
事前条件	
事後条件	
基本系列	1.勤怠情報管理メニューを立ち上げる 2.①②の2つの方法から選択 3.①StaffIDと調べたい年月を入力する ②ShopIDと日付を入力する 4.①その人のその月の出勤時間、退勤時間、休憩時間を表示する ②その日のすべての従業員の出勤時間、退勤時間、休憩時間を表示する
代替系列	
例外系列	その日付の働くのデータが存在しなかったとき
サブユースケース	

- ・アプリケーションで利用する問い合わせ SQL(目的と SQL 文)一覧

使用クラス:PrintAllPTJInformation

ユースケースとの対応:従業員情報表示

PTJ の全てのデータを表示

```
SELECT ptj.staffID, staff.staffName, staff.phoneNumber,  
staff.address,staff.birthDate, staff.fare, ptj.position  
FROM ptj  
LEFT JOIN Staff ON ptj.staffID = Staff.staffID
```

使用クラス:PrintAllEmployeeInformation

ユースケースとの対応:従業員情報表示

社員の全てのデータを表示

```
SELECT employee.staffID, staff.staffName, staff.phoneNumber,  
staff.address, staff.birthDate, staff.fare,  
employee.position,employee.salary  
FROM employee  
LEFT JOIN Staff ON employee.staffID = Staff.staffID
```

使用クラス:PrintAllShopInformation

ユースケースとの対応:店舗情報表示

店舗情報をすべて表示

```
SELECT shopID, shopName, phoneNumber, shop.address  
FROM shop
```

使用クラス:SearchStaffName

ユースケースとの対応:従業員情報検索

従業員の名前検索

```
SELECT *  
FROM staff  
WHERE staffName LIKE ?
```

使用クラス:SearchWorkData

ユースケースとの対応:勤怠情報管理

1:StaffID と年月から従業員の勤務情報を検索

```
SELECT shopID, staffID, date, start, end, break
FROM work
WHERE staffID = ?
AND date DIV 100 = ?
```

2:ShopID と日付から従業員の勤務情報を検索

```
SELECT shopID, staffID, date, start, end, break
FROM work
WHERE shopID = ?
AND date = ?
```

使用クラス:Payment

ユースケースとの対応:給与明細表示

該当する従業員・日付時の給与体系を検索

```
SELECT *
FROM payment
WHERE staffID=? AND startDate=?
```

スタッフ名を呼び出す

```
SELECT *
FROM staff
WHERE staffID=?
```

勤怠情報から該当者の該当機関の勤務情報を呼び出す

```
SELECT *
FROM work
WHERE staffID=? AND date>? AND date<?
```

・アプリケーションを実装するためのクラス一覧

・AbstractExecuter:

データベースクエリを実行し、結果を表示するための抽象クラス。

・StaffManagementSystem:

ユーザーインターフェースを提供し、ユーザーの選択に基づいて適切なデータベース操作

を実行するメインクラス。

- ・ PrintAllPTJInformation:

全アルバイト情報をデータベースから取得し、表示するクラス。

- ・ PrintAllEmployeeInformation:

全社員情報をデータベースから取得し、表示するクラス。

- ・ PrintAllShopInformation:

全店舗情報をデータベースから取得し、表示するクラス。

- ・ SearchStaffName:

従業員名で部分一致検索を行い、検索結果を表示するクラス。

- ・ SearchWorkData:

従業員の勤務情報を 2 パターンの検索方法から検索し表示する役割を持つクラス

StaffID と年月による検索、ShopID と日付から検索ができる

- ・ Payment:

データベースから取得した勤務情報と時給についてのデータから給与（交通費含む）を計算し、給与明細として表示する機能を持つクラス

自分の貢献作業

- ・ 自分の実装したクラス

- ・ StaffManagementSystem

- ・ PrintAllPTJInformation

- ・ SearchWorkData