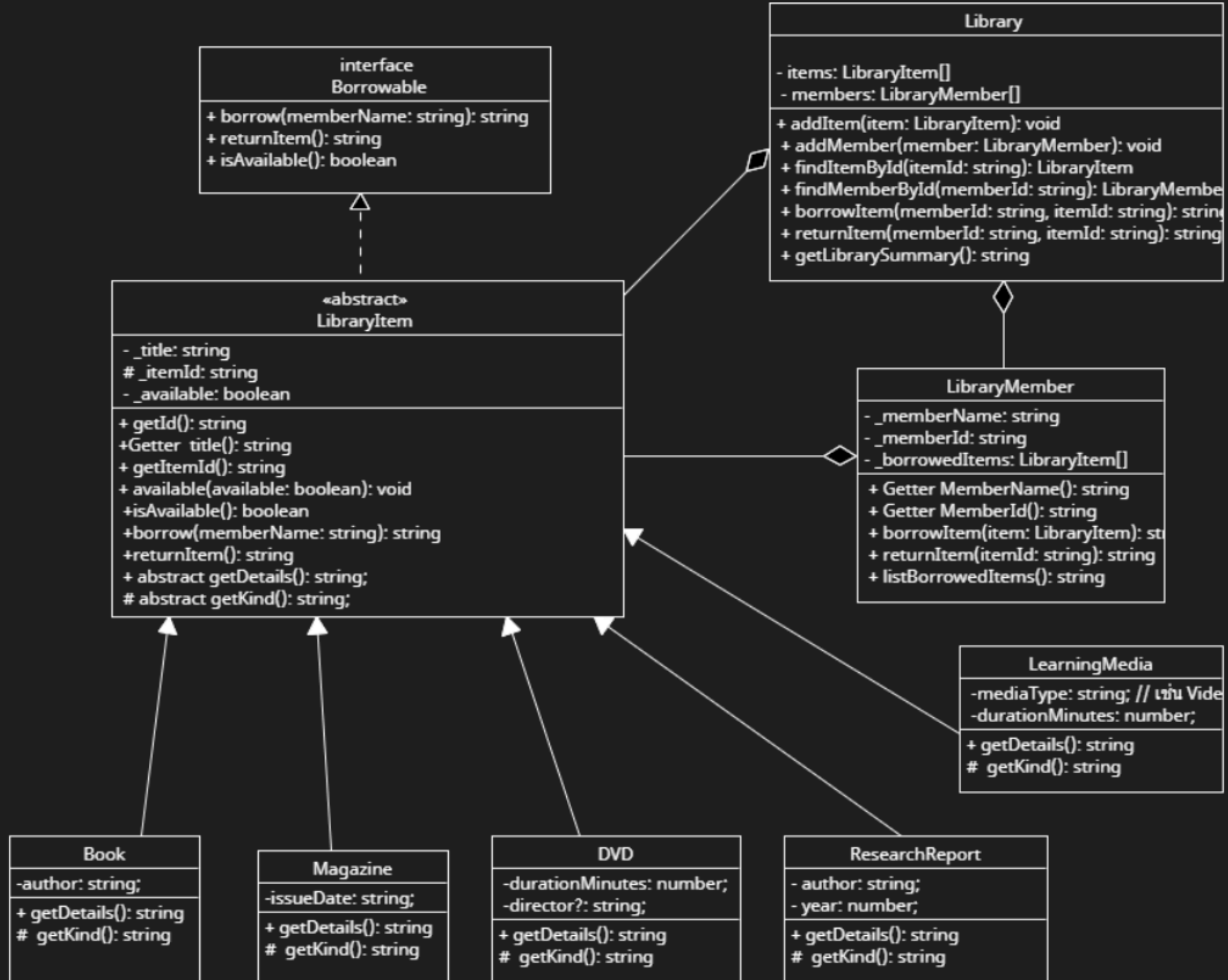
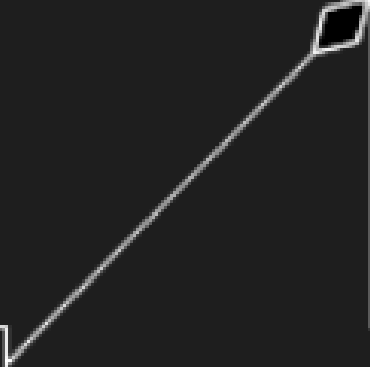
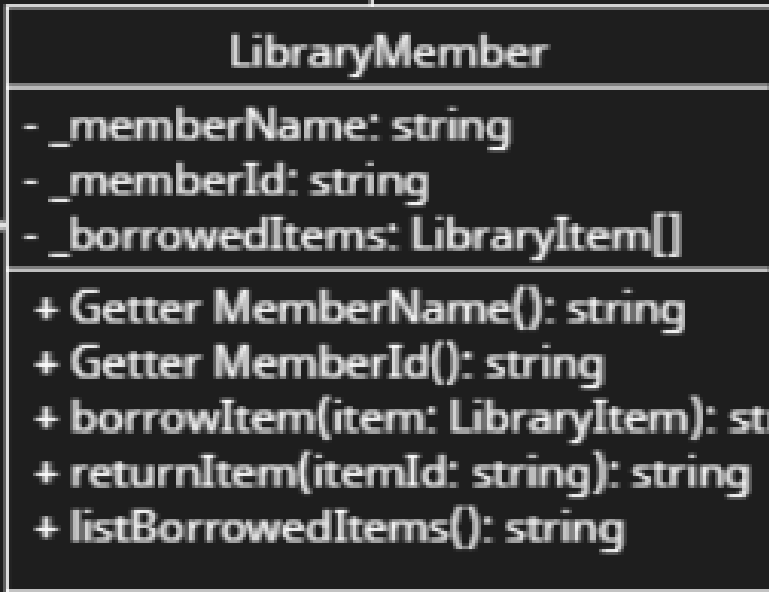
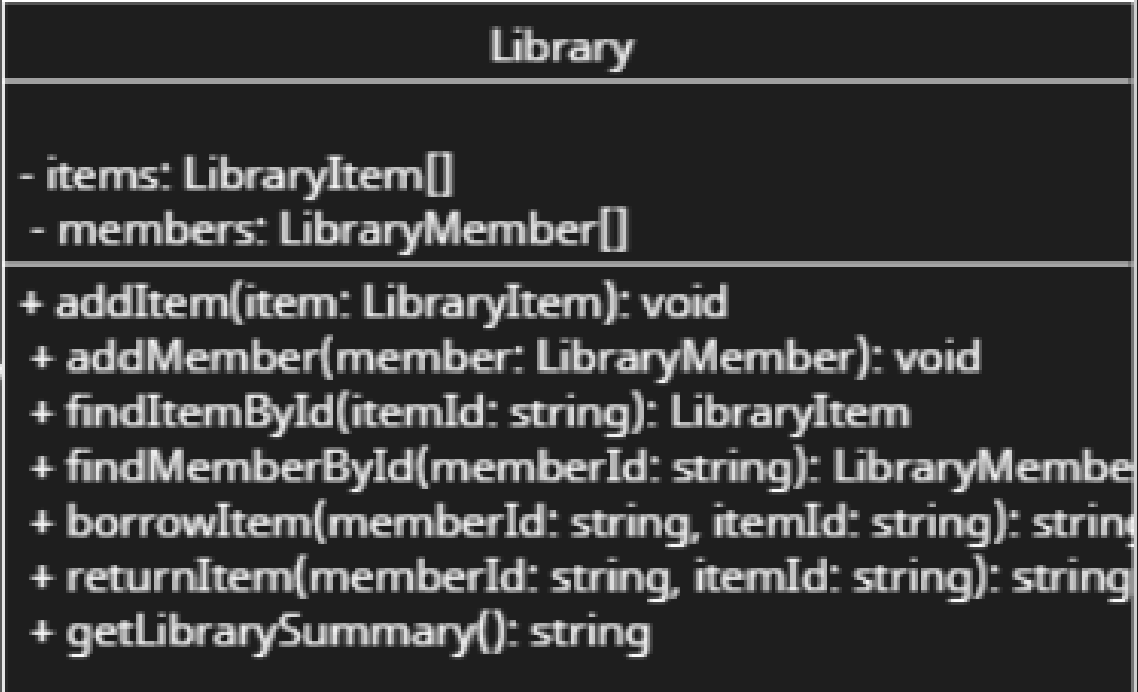
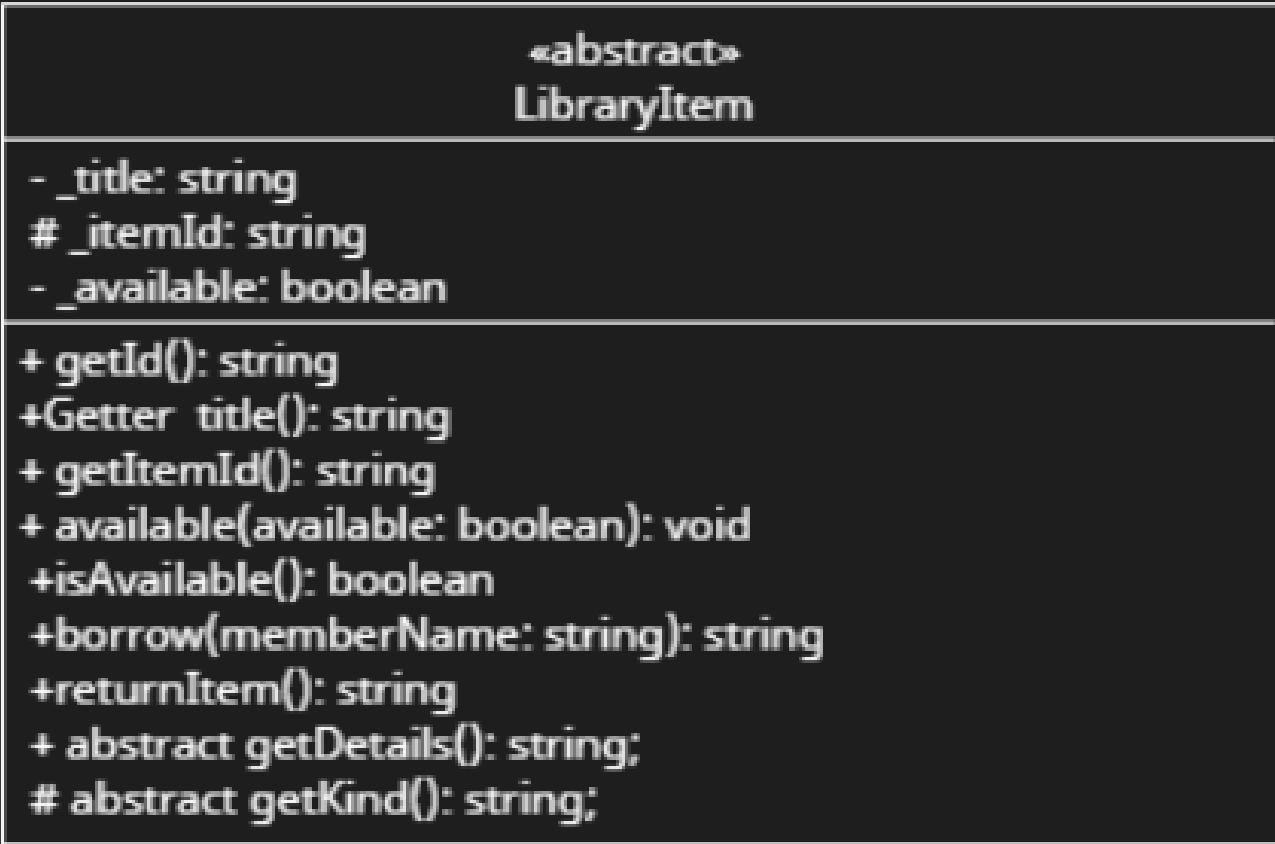
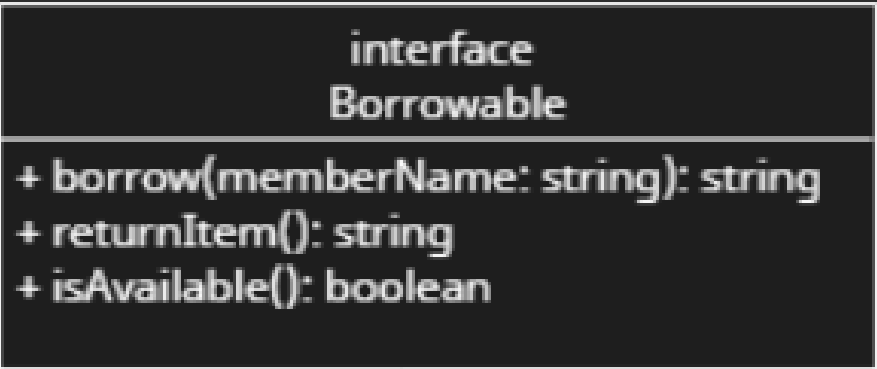


# FEATURES

- จัดเก็บสื่อหลายประเภท: BOOK, MAGAZINE, DVD, LEARNING MEDIA, RESEARCH REPORT
- จัดการสมาชิกห้องสมุด (LIBRARYMEMBER)
- ฟังก์ชันยืม – คืน พร้อมตรวจสอบสถานะ
- แสดงสรุปข้อมูลสื่อและสมาชิกทั้งหมด





```
interface Borrowable {  
    borrow(memberName: string): string;  
    returnItem(): string;  
    isAvailable(): boolean;  
}
```

- กำหนดสัญญาว่า class ใดที่ implement ต้องมี method borrow(), returnItem(), isAvailable()
- LibraryItem implements Borrowable → คือการใช้ Interface

```
abstract class LibraryItem implements Borrowable  
class Book extends LibraryItem  
class Magazine extends LibraryItem  
class DVD extends LibraryItem  
class LearningMedia extends LibraryItem  
class ResearchReport extends LibraryItem
```

- LibraryItem เป็น abstract class
- Subclass ทั้งหมด (Book, Magazine, DVD, LearningMedia, ResearchReport) มาจาก LibraryItem

```
private _borrowedItems: LibraryItem[] = [];
```

- LibraryMember มีการยืมของ LibraryItem อยู่ แต่ไม่ได้ถาวร

```
private items: LibraryItem[] = [];  
private members: LibraryMember[] = [];
```

- ตรงนี้คือ Aggregation เพราะเป็นการเก็บ reference ของ object อื่นมาใช้

```
private _title: string;  
protected itemId: string;  
private _available: boolean = true;
```

```
private _memberName: string;  
private _memberId: string;  
private _borrowedItems: LibraryItem[] = [];
```

- ใช้ private / protected เพื่อให้เข้าถึงตรงๆ

```
public get title(): string {  
    return this._title;  
}
```

```
public getItemId(): string {  
    return this.itemId;  
}
```

```
public isAvailable(): boolean {  
    return this._available;  
}
```

- ตรงนี้คือการใช้ Encapsulation เพื่อควบคุมการเข้าถึงข้อมูล