

## [미션 1] 나만의 책장 만들기

### 학습 목표

- 클래스를 사용하여 객체를 만들고, 정보 출력 메소드를 활용할 수 있다.

### 문제 설명

서점에서 책 정보를 관리하기 위해 Book 클래스를 만들려고 해.  
책의 제목(title)과 저자(author)를 저장하고, 책 정보를 출력하는 기능을 만들어 보자.

### 미션 과제

1. Book 클래스를 만들고, 제목과 저자를 저장할 수 있도록 필드를 작성하자.
2. 생성자(Constructor)를 이용하여 값을 초기화할 수 있게 하자.
3. 책 정보를 출력하는 printInfo() 메소드를 만들자.
4. Main 클래스에서 책 두 권을 생성하고 정보를 출력해보자.

### 추가 도전 과제

- 사용자의 입력을 받아서 책 정보를 동적으로 추가해보자.  
(힌트: Scanner 사용)

## [미션 2] 나만의 은행 계좌 만들기

### 학습 목표

- 필드 값을 변경하는 메소드 작성법을 익히고, 조건문을 활용할 수 있다.

### 문제 설명

간단한 은행 계좌 시스템을 만들자.

계좌 소유자의 이름과 잔액 정보를 관리하고, 입금과 출금 기능을 구현해보자.

### 미션 과제

1. BankAccount 클래스를 만들고, 이름(owner)과 잔액(balance)을 저장하자.
2. 입금 메소드 deposit(int amount)와 출금 메소드 withdraw(int amount)를 구현하자.
3. 현재 잔액을 출력하는 printBalance() 메소드를 만들자.
4. Main 클래스에서 계좌를 하나 만들고, 입금과 출금을 실행해보자.

### 추가 도전 과제

- 출금할 때 잔액보다 많은 금액을 출금하려 하면, "잔액 부족" 메시지를 출력하게 만들자.
- 여러 계좌를 배열로 만들어서 관리해보자.

## [미션 3] 영화 추천 시스템 만들기

### 학습 목표

- 여러 객체를 배열로 관리하고, 조건문으로 분류하는 방법을 익힌다.

### 문제 설명

Movie 클래스를 만들어서 영화 제목과 평점을 관리하자.

평점이 8 점 이상인 영화는 “추천 영화”로 출력되도록 프로그램을 만들어보자.

### 미션 과제

1. Movie 클래스를 만들고 제목(title)과 평점(rating)을 저장하자.
2. 영화 정보를 출력하는 showInfo() 메소드를 만들자.
3. 평점이 8 점 이상일 경우 isRecommended() 메소드가 true 를 반환하게 하자.
4. Main 클래스에서 영화 3 편을 만들고 반복문을 이용해 모든 영화 정보를 출력하자.

### 추가 도전 과제

- 사용자로부터 영화 제목과 평점을 받아서 Movie 객체를 동적으로 추가해보자.
- 추천 영화를 따로 출력하는 기능을 추가해보자.