```
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>
#include <string.h>
#include <ctype.h>
struct Book
    char title[50];
    char authors[50];
    char press[50];
    int page;
    int price;
    char borrow[50];
};
void BookList(struct Book books[], int size)
    printf("\nTitle\tauthors\textra tPress\textra tPage\textra tPrice\textra tBorrow\textra n");
    printf("----₩t-----₩t-----₩t-----₩t-----₩n");
    int i;
    for (i = 0; i < size; ++i)
         printf("%s\t%s\t%s\t%d\t%d\t%d\t%d\tn", books[i].title, books[i].authors,
             books[i].press, books[i].page, books[i].price, books[i].borrow);
    }
}
void searchBook(struct Book books[], int size)
    char searchTitle[50];
    printf("₩n검색할 도서를 선택하세요: ");
    scanf("%s", searchTitle);
    int booksearch = 0;
    int i;
    for (i = 0; i < size; ++i)
         if (_stricmp(books[i].title, searchTitle) == 0)
             printf("₩n도서 정보:\n");
             printf("Title: %s\mathbb{\text{Mn}Authors: \mathbb{\text{%s}\mathbb{\text{Mn}Press: \mathbb{\text{Mn}Price: \mathbb{\text{Mm}Borrow: \mathbb{\text{Mn}}\mathbb{\text{N}}",
                  books[i].title, books[i].authors, books[i].press, books[i].page,
                  books[i].price, books[i].borrow);
             booksearch = 1;
             break;
         }
    }
    if (!booksearch)
```

```
{
       printf("₩n보유하고 있지 않은 도서입니다.₩n");
   }
}
void borrowBook(struct Book books[], int size)
   char borrowTitle[50];
   printf("₩n대출할 책의 이름을 선택하세요: ");
   scanf("%s", borrowTitle);
   int bookborrow = 0;
   int i;
   for (i = 0; i < size; ++i)
       if (_stricmp(books[i].title, borrowTitle) == 0)
           if (strcmp(books[i].borrow, "available") == 0)
               printf("₩n대출 되었습니다.₩n");
               strcpy(books[i].borrow, "borrowing");
           }
           else
           {
               printf("₩n대출 중이라 대출 할 수 없습니다₩n");
           }
           bookborrow = 1;
           break;
       }
   }
   if (!bookborrow)
       printf("₩n보유하고 있지 않은 도서입니다.₩n");
   }
}
void returnBook(struct Book books[], int size)
   char returnTitle[50];
   printf("₩n반납할 도서를 선택하세요: ");
   scanf("%s", returnTitle);
   int returnbook = 0;
    int i;
   for (i = 0; i < size; ++i)
       if (_stricmp(books[i].title, returnTitle) == 0)
           if (strcmp(books[i].borrow, "borrowing") == 0)
```

```
{
                  printf("₩n책이 반납 되었습니다.₩n");
                  strcpy(books[i].borrow, "available");
              }
              else
              {
                  printf("₩n이 책은 대출 중이 아닙니다. 반납 할 수 없습니다.₩n");
              }
              returnbook = 1;
              break;
         }
    }
    if (!returnbook)
         printf("₩n보유하고 있지 않은 도서입니다.₩n");
    }
}
int main()
{
    struct Book books[] =
        {"Truth", "John", "Century", 300, 20000, "available"}, {"Love", "Paul", "Goods", 200, 15000, "available"}, {"Joy", "James", "Cookie", 250, 18000, "available"},
         {"Thanks", "Mark", "Saejong", 240, 21000, "available"}, {"God", "Johnson", "Jungjo", 450, 35000, "available"}
    };
    int Booknumber = sizeof(books) / sizeof(books[0]);
    int choice;
    do {
         printf("₩n[도서목록]: 1 [검색]: 2 [대출]: 3 [반납]: 4 [종료]: 5₩n");
         printf("메뉴를 선택하세요: ");
         scanf("%d", &choice);
         switch (choice) {
         case 1:
              BookList(books, Booknumber);
              break;
         case 2:
              searchBook(books, Booknumber);
              break;
         case 3:
              borrowBook(books, Booknumber);
              break;
```

```
case 4:

returnBook(books, Booknumber);
break;

case 5:

printf("프로그램을 종료합니다.\n");
break;

default:
printf("잘못된 선택입니다. 다시 선택하세요.\n");
}

while (choice != 5);

return 0;
}
```