



YAPISAL PROGRAMLAMA

KÜTÜPHANE OTOMASYONU

AHMET ELBİR

ENES ÖZDOĞAN

20011056

MENÜLER

Program 3 alt menüden oluşmaktadır. Program açıldığında kullanıcıdan hangi alt menüye dâlanılacağı bilgisi alınır. Kitap işlemleri için 1. Menüye gidilir. Öğrenci işlemleri için 2, yazar işlemleri için 3. Menüye gidilir.

```
KITAP : OGR : YAZAR işlemleri için sirasiyla 1 : 2 : 3
```

Kitap Alt Menüsü

7 farklı işlemten oluşan bu menüde kitap ekle/sil/güncelleme ve listeleme işlemleri yapılır.

```
===KITAP ISLEMLERI===  
Kitap Ekleme:1  
Kitap Silme:2  
Kitap Guncelleme:3  
Kitap Bilgi Goruntuleme:4  
Kitap Listeleme:5  
Zamaninda Teslim Edilmeyen Kitaplar:6  
Yazar Eslestir/Guncelle:7
```

Öğrenci Alt Menüsü

9 farklı işlemten oluşan bu menüde öğrenci ekle/sil/güncelleme, listeleme ve kitap ödünç/teslim işlemleri gerçekleştirilmiştir.

```
===OGR ISLEMLERI===  
OGR Ekle:1  
OGR Sil:2  
OGR Guncelle:3  
OGR Bilgi Goruntuleme:4  
Kitap Teslim Etmemis OGR Listesi:5  
Cezali OGR listesi:6  
Tum OGR Listesi:7  
Kitap Odunc:8  
Kitap Teslim:9
```

Yazar Alt Menüsü

5 farklı işlemten oluşan bu menüde yazar ekle/sil/güncelleme ve listeleme işlemleri gerçekleştirilmiştir.

```
===YAZAR ISLEMLERI===  
Yazar Ekle:1  
Yazar Sil:2  
Yazar Guncelle:3  
Yazar Bilgisi Goruntule:4  
Yazar Listesi:5
```

KİTAP İŞLEMLERİ

Kitap Ekleme İşlemi

```
KITAP* kitap_ekle(KITAP* head,KITAP* tmp,int choice);
```

Öncelikle parametre olarak linkli listenin baş elemanını, eklenecek kitabı ve tercihi alır. Fonksiyon içerisinde linkli listenin sonuna kadar ilerlenir ve kitap eklenir. Eğer ilerleme esnasında aynı ISBN bilgisine sahip kitap bulunursa iki kitabın adedi toplanır ve linkli listeye yeni eleman eklenmez. Sıralamanın doğru olabilmesi ve kitap adedince etiket oluşturulabilmesi için 3 farklı fonksiyon ile birlikte çalışır.

```
void sortByBookName(KITAP **head);
```

```
void call_kitorn(KITAP* head,KITAP_ODUNC* arr_odunc,int M);
```

```
KITAP_ORN* ornek_ekle(KITAP_ORN* head,int ord,char* isbn,KITAP_ODUNC* arr_odunc,int M);
```

sortBybookName fonksiyonunda kitaplar isme göre sıralanır sonrasında aynı isimli kitaplar varsa ISBN kontrolü yapar ve ona göre değerler değiş tokuş edilir.

Call_kitorn fonksiyonunda her kitap için adedince etiket oluşturulur ve etiket no, durum bilgisi doldurulur. Parametre olarak KitapOdunc.csv dosyasını tutan diziyi alır. Böylelikle kitapların durumları dosya ile uyumlu hale getirilir.

Ornek_ekle() fonksiyonunda bir kitabın adedi kadar etiket oluşturulur ve durumları da KitapOdunc.csv dosyasındaki işlem tipi kontrolüne bakarak uyumlu bir şekilde doldurulur.

```
===KITAP EKLE===  
Ad:Jumbo Karides  
ISBN:19191919193  
Adet:3  
Programi Sonlandirmek icin :1  
0
```

Java progr	1.23E+12	5	Algoritma	9.78E+12	2
Algoritma	9.78E+12	2	Fizik 1	1.23E+12	4
Veri Yapı	9.78E+12	3	Fizik 2	1.23E+12	3
Fizik 1	1.23E+12	4	Java progr	1.23E+12	5
Fizik 2	1.23E+12	3	Jumbo Kar	1.92E+12	3
			Veri Yapı	9.78E+12	3

Kitap Silme İşlemi

```
KITAP* kitap_sil(KITAP* head, char* isbn);
```

Bu fonksiyonda ISBN numarası alınan kitap, kitap linki listesinde aranır. Adet kontrolü yapılır ve eğer adedi varsa adet miktarından bir azaltılır eğer kitabın 1 örneği varsa tamamen dosyadan ve linkli listeden kaldırılır.

```
===KITAP ISLEMLERI===
Kitap Ekleme:1
Kitap Silme:2
Kitap Guncelleme:3
Kitap Bilgi Goruntuleme:4
Kitap Listeleme:5
Zamaninda Teslim Edilmeyen Kitaplar:6
Yazar Eslestir/Guncelle:7
2
silinmek istenen kitap ISBN:1234567891011
AD:Java programlama, ISBN:1234567891011, Adet:4
EtiketNo:1234567891011_0 , Durum:Rafta
EtiketNo:1234567891011_1 , Durum:Rafta
EtiketNo:1234567891011_2 , Durum:Rafta
EtiketNo:1234567891011_3 , Durum:Rafta
```

Java progr	1.23E+12	5	Algoritma	9.78E+12	2
Algoritma	9.78E+12	2	Fizik 1	1.23E+12	4
Veri Yapı	9.78E+12	3	Fizik 2	1.23E+12	3
Fizik 1	1.23E+12	4	Java progr	1.23E+12	4
Fizik 2	1.23E+12	3	Jumbo Kai	1.92E+12	3
			Veri Yapı	9.78E+12	3

Kitap Güncelleme İşlemi

```
KITAP* kitap_guncelle(KITAP* head, KITAP_ODUNC* arr_odunc, int M);
```

Parametre olarak kitap listesini ve KitapOdunc.csv bilgilerini alır. ISBN bilgisi ile listede kitap aranır. Switch case yapısı ile ad, ISBN ve adet güncellemesi yapılır.

```
===KITAP GUNCELLE===
Guncellenecek kitap ISBN:9780140449335
Ad,ISBN,Adet icin sirasiyla 1,2,3:2
Yeni ISBN:1818181818182
Programi Sonlandirmak icin :1
0
```

```
===KITAP GUNCELLE===
Guncellenecek kitap ISBN:9780140449335
Ad,ISBN,Adet icin sirasiyla 1,2,3:1
Yeni Ad:Veri
Programi Sonlandirmak icin :1
0
```

```

===KITAP GUNCELLE===

Guncellenecek kitap ISBN:1818181818182
Ad,ISBN,Adet icin sirasiyla 1,2,3:3
Yeni Adet:4
2Programi Sonlandirmak icin :1

```

Algoritma	9.78E+12	2	B5					1818181818182
Fizik 1	1.23E+12	4		A	B	C	D	E
Fizik 2	1.23E+12	3	1	Algoritma	9.78E+12	2		
Java progr	1.23E+12	5	2	Fizik 1	1.23E+12	4		
Jumbo Ka	1.92E+12	3	3	Fizik 2	1.23E+12	3		
Veri Yapı	9.78E+12	3	4	Java progr	1.23E+12	5		
			5	Veri	1.82E+12	4		

Kitap Bilgi Görüntüleme

```
void kitap_bilgi(KITAP* head);
```

Parametre olarak kitap linkli listesini alır. Verilen kitap adına göre arama yapar ve bilgileri ekrana yazdırır.

```

===KITAP BILGI GORUNTULEME===
Kitap Adi:Veri
EtiketNo:1818181818182_0 , Durum:Rafta
EtiketNo:1818181818182_1 , Durum:Rafta
EtiketNo:1818181818182_2 , Durum:Rafta
EtiketNo:1818181818182_3 , Durum:Rafta
Programi Sonlandirmak icin :1

```

Kitap Listeleme

```
void print_kitap(KITAP* head)
```

Kitap linkli listesinde sona kadar ilerlenir ve değerler bastırılır.

```

KITAP
AD:Algoritma Analizi, ISBN:9780140449334, Adet:2
EtiketNo:9780140449334_0 , Durum:Rafta
EtiketNo:9780140449334_1 , Durum:Rafta
KITAP
AD:Fizik 1, ISBN:1234567891012, Adet:4
EtiketNo:1234567891012_0 , Durum:Rafta
EtiketNo:1234567891012_1 , Durum:18011001
EtiketNo:1234567891012_2 , Durum:17011020
EtiketNo:1234567891012_3 , Durum:Rafta
KITAP
AD:Fizik 2, ISBN:1234567891013, Adet:3
EtiketNo:1234567891013_0 , Durum:Rafta
EtiketNo:1234567891013_1 , Durum:18011010
EtiketNo:1234567891013_2 , Durum:Rafta
KITAP
AD:Java programlama, ISBN:1234567891011, Adet:5
EtiketNo:1234567891011_0 , Durum:Rafta
EtiketNo:1234567891011_1 , Durum:Rafta
EtiketNo:1234567891011_2 , Durum:Rafta
EtiketNo:1234567891011_3 , Durum:Rafta
EtiketNo:1234567891011_4 , Durum:Rafta
KITAP
AD:Veri, ISBN:1818181818182, Adet:4
EtiketNo:1818181818182_0 , Durum:Rafta
EtiketNo:1818181818182_1 , Durum:Rafta
EtiketNo:1818181818182_2 , Durum:Rafta
EtiketNo:1818181818182_3 , Durum:Rafta

```


B5						1818181818182
	A	B	C	D	E	
1	Algoritma	9.78E+12	2			
2	Fizik 1	1.23E+12	4			
3	Fizik 2	1.23E+12	3			
4	Java progr	1.23E+12	5			
5	Veri	1.82E+12	4			

Zamanında Teslim Edilmemiş Kitapların Listelenmesi

Bu işlemde özel herhangi bir fonksiyon kullanılmaz. Kullanıcı kitap teslim işlemi yaparken eğer gecikme yaşandıysa outdated linkli listesine bu kitap eklenir. Sonrasında print_kitap() fonksiyonu ile yazdırılır.

D5					28.12.2022
	A	B	C	D	E
1	123456789	18011001	0	12.12.2022	
2	123456789	18011010	0	12.12.2022	
3	123456789	18011020	1	22.12.2022	
4	978014044	17011010	1	27.12.2022	
5	123456789	17011020	0	28.12.2022	

```

===KITAP IADE===
Gun ay yil sirasiyla:03 03 2024
OGR NO:17011020
Teslim Edilecek Kitap EtiketNo:1234567891012_2
Teslim Basarili

```

Soldaki ekran görüntüsünde kitabın ödünç alınma tarihi verilmiştir. Sağdaki görüntüde bu kitap gecikme olacak şekilde teslim edilmiştir ve outdated listesine eklenmiştir.

```

---SUREYI ASAN IADELER---

===KITAP LISTESI===

KITAP
AD:Fizik 1, ISBN:1234567891012, Adet:4

```

İlişki Ekleme ve Güncelleme

```
void iliski_ekle(char* isbn,int id,YAZ* head1,KITAP* head2)
```

Parametre olarak kitap ISBN, yazar id, yazar listesi ve kitap listesini alır. Veri tutarlılığı kontrolü yapıldıktan sonra KitapYazar.csv dosyasına ilişki eklenir/güncellenir. Yazar silinmesi durumu yazar silinme fonksiyonunda ele alınacaktır.

```

===ILISKI EKLE===

Iliskilenecek kitap ISBN:1818181818182
Iliskilenecek Yazar ID:6

```

A10						1818181818182
	A	B	C	D	E	
1	1.23E+12	2				
2	9.78E+12	3				
3	1.23E+12	1				
4	1.23E+12	4				
5	1.23E+12	1				
6	1.23E+12	5				
7	9.78E+12	6				
8	9.78E+12	7				
9	9.78E+12	8				
10	1.82E+12	6				

ÖĞRENCİ İŞLEMLERİ

ÖĞRENCİ EKLE

```
OGR* ogr_ekle(OGR* head, OGR* tmp, int choice)
```

Parametre olarak öğrenci linkli listesinin baş elemanını, eklenecek yeni öğrenciyi ve tercihi alır. Doubly linkli liste içerisinde öğrenci numarasına göre sıralı ekler.

```
===OGR EKLE===
```

```
OGR ID:19011105
Ad:Oguz
Soy Ad:Topalak
Puan:100
```

```
OGR
ID:19011105, Ad:Oguz , Soy Ad: Topalak
```

17011001	Andrew	Zisserman	100
17011005	Francisco	Matorras	100
17011010	Ross	Girshick	100
17011015	Trevor	Hastie	100
17011020	Yann	LeCun	80
18011001	Geoffrey	Hinton	100
18011005	Yoshua	Bengio	100
18011010	Robert	Tibshirani	100
18011015	Kaiming	He	100
18011020	Ilya	Sutskever	100
19011105	Oguz	Topalak	100

ÖĞRENCİ SİL

```
OGR* ogr_sil(OGR* head)
```

Parametre olarak öğrenci listesinin baş elemanını alır. Öğrenci numarasına göre silinmek istenen öğrenciyi arar ve doubly linkli listeden kaldırır.

```
===OGR SİL===  
Silinecek Eleman ID:17011020
```

17011001	Andrew	Zisserman	100
17011005	Francisco	Matorras	100
17011010	Ross	Girshick	100
17011015	Trevor	Hastie	100
18011001	Geoffrey	Hinton	100
18011005	Yoshua	Bengio	100
18011010	Robert	Tibshirani	100
18011015	Kaiming	He	100
18011020	Ilya	Sutskever	100
19011105	Oguz	Topalak	100

Görüldüğü üzere başlangıçta var olan 17011020 numaralı öğrenci silinmiştir.

ÖĞRENCİ GÜNCELLE

```
void ogr_gun(OGR* head)
```

Parametre olarak öğrenci listesinin baş elemanını alır. Öğrenci numarasına göre silinmek istenen öğrenciyi arar ve switch case yapısı ile ad, soyad ve puan güncellemesi yapılır.

```
===OGR GÜNCELLE===  
Guncellenecek OGR ID:19011105  
Ad,soyad,puan,hepsi sirasiyla 1,2,3,4  
1  
Yeni AD:Mert
```

```
===OGR GÜNCELLE===  
Guncellenecek OGR ID:19011105  
Ad,soyad,puan,hepsi sirasiyla 1,2,3,4  
2  
Yeni Soyad:Yuvarlak
```

```
===OGR GÜNCELLE===  
Guncellenecek OGR ID:19011105  
Ad,soyad,puan,hepsi sirasiyla 1,2,3,4  
3  
Yeni Puan:40
```



```

===OGR GORUNTULEME===

Goruntulenmek istenen Ogr ID:19011105
OGR
ID:19011105, Ad:Mert , Soy Ad: Yuvarlak, Puan:40

```

17011001	Andrew	Zisserman	100
17011005	Francisco	Matorras	100
17011010	Ross	Girshick	100
17011015	Trevor	Hastie	100
18011001	Geoffrey	Hinton	100
18011005	Yoshua	Bengio	100
18011010	Robert	Tibshirani	100
18011015	Kaiming	He	100
18011020	Ilya	Sutskever	100
19011105	Mert	Yuvarlak	40

ÖĞRENCİ BİLGİSİ GÖRÜNTÜLE

```
void ogr_bilgi(OGR* head,KITAP_ODUNC* arr_odunc,int M)
```

Parametre olarak öğrenci listesinin baş elemanını ve KitapOdunc bilgilerini alır. Öğrenci numarasına göre istenen öğrenciyi her ikide yapıda arar. Bilgilerini ekrana yazdırır. Örnek csv dosyası aşağıdadır.

```

===OGR GORUNTULEME===

Goruntulenmek istenen Ogr ID:18011001
OGR
ID:18011001, Ad:Geoffrey , Soy Ad: Hinton, Puan:100
18011001 1234567891012 1 0 12.12.2022

```

123456789	18011001	0	12.12.2022
123456789	18011010	0	12.12.2022
123456789	18011020	1	22.12.2022
978014044	17011010	1	27.12.2022
123456789	17011020	1	3.3.2024

KİTAP TESLİM ETMEMİŞ ÖĞRENCİLERİ LİSTELEME

```
void ogr_teslim0(OGR* head,KITAP_ODUNC* arr_odunc,int M);
```

Parametre olarak öğrenci listesinin baş elemanını, KitapOdunc.csv dosyası bilgilerinin depolandığı struct dizisini alır. İşlem tipi 0 olan kayıtları öğrenci listesinde arayarak ekrana yazdırır. Örnek KitapOdunc.csv dosyası için çıktılar aşağıdadır.

123456789	18011001	0	12.12.2022	<pre> ===KITAP TESLİM ETMEMİS OGR GORUNTULEME=== OGR ID:18011001, Ad:Geoffrey , Soy Ad: Hinton, Puan:100 OGR ID:18011010, Ad:Robert , Soy Ad: Tibshirani, Puan:100 </pre>
123456789	18011010	0	12.12.2022	
123456789	18011020	1	22.12.2022	
978014044	17011010	1	27.12.2022	
123456789	17011020	1	3.3.2024	

CEZALI ÖĞRENCİLERİN LİSTELENMESİ

```
void cezali_ogr(OGR* head){
```

Parametre olarak öğrenci listesinin baş elemanını alır. Puanı 0 dan düşük olan öğrencileri listede arar ve yazdırır.

17011001	Andrew	Zisserman	100
17011005	Francisco	Matorras	100
17011010	Ross	Girshick	100
17011015	Trevor	Hastie	-10
18011001	Geoffrey	Hinton	100
18011005	Yoshua	Bengio	-10
18011010	Robert	Tibshirani	100
18011015	Kaiming	He	100
18011020	Ilya	Sutskever	-10
19011105	Mert	Yuvarlak	40

```

===CEZALI OGR GORUNTULEME===
OGR
ID:17011015, Ad:Trevor , Soy Ad: Hastie, Puan:-10
OGR
ID:18011005, Ad:Yoshua , Soy Ad: Bengio, Puan:-10
OGR
ID:18011020, Ad:Ilya , Soy Ad: Sutskever, Puan:-10

```

TÜM ÖĞRENCİLERİN LİSTELENMESİ

```
void print_liste(void* head,int choice);
```

Void pointer türünde bir linkli liste baş elemanı alır. Tercih değeri 1 ise öğrenci listesi yazdırılır.

17011001	Andrew	Zisserman	100
17011005	Francisco	Matorras	100
17011010	Ross	Girshick	100
17011015	Trevor	Hastie	-10
18011001	Geoffrey	Hinton	100
18011005	Yoshua	Bengio	-10
18011010	Robert	Tibshirani	100
18011015	Kaiming	He	100
18011020	Ilya	Sutskever	-10
19011105	Mert	Yuvarlak	40

```
OGR
ID:17011001, Ad:Andrew , Soy Ad: Zisserman
OGR
ID:17011005, Ad:Francisco , Soy Ad: Matorras
OGR
ID:17011010, Ad:Ross , Soy Ad: Girshick
OGR
ID:17011015, Ad:Trevor , Soy Ad: Hastie
OGR
ID:18011001, Ad:Geoffrey , Soy Ad: Hinton
OGR
ID:18011005, Ad:Yoshua , Soy Ad: Bengio
OGR
ID:18011010, Ad:Robert , Soy Ad: Tibshirani
OGR
ID:18011015, Ad:Kaiming , Soy Ad: He
OGR
ID:18011020, Ad:Ilya , Soy Ad: Sutskever
OGR
ID:19011105, Ad:Mert , Soy Ad: Yuvarlak
```

KİTAP ÖDÜNC ALMA

```
KITAP_ODUNC* kitap_al(OGR* heado,KITAP* headk,KITAP_ODUNC* arr_odunc,int* M);
```

Parametre olarak öğrenci listesinin baş elemanını, kitap listesinin baş elemanını, ödünç teslim bilgilerinin tutulduğu diziyi alır. Fonksiyon içerisinde öncelikle kullanıcıdan alınan bilgilerin doğruluğu tespit edilir. Eğer öğrenci no ve kitap ISBN bulunduysa eşleştirme yapılarak dosyaya arr_odunc dizisi üzerinden yazdırılır. arr_odunc dizisinin boyutu 1 arttırılır ve return edilir. Böylelikle ödünç alınan kitabın bilgileri program içerisinde güncellenir.

```
===KITAP ODUNC===

Gun ay yil sirasiyla:01 01 2024
OGR NO:18011001
Odunc Alinacak Kitap ISBN:1818181818182
```

```
===OGR GORUNTULEME===

Goruntulenmek istenen Ogr ID:18011001
OGR
ID:18011001, Ad:Geoffrey , Soy Ad: Hinton, Puan:100
18011001 1234567891012_1 0 12.12.2022
18011001 1818181818182_0 0 1.1.2024
```

123456789	18011001	0	12.12.2022
123456789	18011010	0	12.12.2022
123456789	18011020	0	22.12.2022
978014044	17011010	0	27.12.2022
123456789	17011020	0	3.3.2024
181818181	18011001	0	1.1.2024

```
KITAP
AD:Veri, ISBN:1818181818182, Adet:4
EtiketNo:1818181818182_0 , Durum:18011001
```

KİTAP TESLİM ETME

```
void kitap_ver(KITAP_ODUNC* arr_odunc,int M, OGR* heado,KITAP* headk,KITAP** outdated)
```

Parametre olarak ödünç, teslim dizisini ve kitap, öğrenci ve geç teslim edilen kitapların tutulduğu outdated listesini alır. Arr_odunc dizisinde verilen Etiket no ve öğrenci numarasını eşleşmesini arar. Verilerin tutarlılığını kontrol eder. Eşleşme bulunursa gün farkı hesaplanır. Eğer 15 günden fazla ise öğrencinin 10 azaltılır ve ilgili ISBN'e sahip kitap outdated listesine eklenir. Son olarak da arr_odunc dizisi ile dosyaya bilgiler tekrardan yazılır.

```
===KITAP İADE===  
  
Gun ay yil sirasiyla:03 03 2024  
OGR NO:18011010  
Teslim Edilecek Kitap EtiketNo:1234567891013_1  
  
Teslim Basarili
```

```
===OGR GORUNTULEME===  
  
Goruntulenmek istenen Ogr ID:18011010  
OGR  
ID:18011010, Ad:Robert , Soy Ad: Tibshirani, Puan:90  
18011010 1234567891013_1 1 3.3.2024
```

```
KITAP  
AD:Fizik 2, ISBN:1234567891013, Adet:3  
EtiketNo:1234567891013_0 , Durum:Rafta  
EtiketNo:1234567891013_1 , Durum:Rafta
```

Teslimden Sonra

123456789	18011001	0	12.12.2022
123456789	18011010	1	3.3.2024
123456789	18011020	1	22.12.2022
978014044	17011010	1	27.12.2022
123456789	17011020	0	28.12.2022

Teslimden Önce

123456789	18011001	0	12.12.2022
123456789	18011010	0	12.12.2022
123456789	18011020	1	22.12.2022
978014044	17011010	1	27.12.2022
123456789	17011020	0	28.12.2022

YAZAR İŞLEMLERİ

YAZAR EKLE

```
YAZ* yazar_ekle(YAZ* head,YAZ* tmp,int choice);
```

Parametre olarak yazar listesinin baş elemanını, eklenecek yazarı ve tercihi alır. ID değeri otomatik olarak artan bir şekilde yazara atanır.

```
===YAZAR EKLE===
```

```
Ad:Ramazan  
Soy Ad:Dodurga
```

```
YAZAR  
ID:9, Ad:Ramazan , Soy Ad: Dodurga
```

1	Michael	Jordan	
2	Stephen	Boyd	
3	Kalyanmo	Deb	
4	David	Johnson	
5	Scott	Kirkpatrick	
6	Lieven	Vandenberghe	
7	Fabian	Pedregosa	
8	Jorge	Nocedal	
9	Ramazan	Dodurga	

1	Michael	Jordan	
2	Stephen	Boyd	
3	Kalyanmo	Deb	
4	David	Johnson	
5	Scott	Kirkpatrick	
6	Lieven	Vandenberghe	
7	Fabian	Pedregosa	
8	Jorge	Nocedal	

YAZAR SİL

```
YAZ* yazar_sil(YAZ* head,int n);
```

```
void iliski_sil(KITAP_YAZ* book_auth,int N,int id,YAZ* head);
```

Parametre olarak yazar listesinin baş elemanını, silinecek yazar numarasını alır. Linkli liste içerisinde arama yaparak eleman bulunur ve silinir. İlişki_sil() fonksiyonu silinmek istenen id ye KitapYazar.csv dosyasında ulaşarak -1 güncellemesi yapar.

```
===YAZAR SİL===
```

```
Silinecek Yazar ID:4
```

```
===YAZAR SİL===
```

```
ID:3, Ad:Kalyanmoy , Soy Ad: Deb  
YAZAR  
ID:5, Ad:Scott , Soy Ad: Kirkpatrick  
YAZAR
```


1	Michael	Jordan
2	Stephen	Boyd
3	Kalyanmo	Deb
5	Scott	Kirkpatrick
6	Lieven	Vandenberghe
7	Fabian	Pedregosa
8	Jorge	Nocedal
9	Ramazan	Dodurga

1	Michael	Jordan
2	Stephen	Boyd
3	Kalyanmo	Deb
4	David	Johnson
5	Scott	Kirkpatrick
6	Lieven	Vandenberghe
7	Fabian	Pedregosa
8	Jorge	Nocedal
9	Ramazan	Dodurga

A4				1234567891012	
	A	B	C	D	E
1	1.23E+12	2			
2	9.78E+12	3			
3	1.23E+12	1			
4	1.23E+12	-1			
5	1.23E+12	1			
6	1.23E+12	5			
7	9.78E+12	6			
8	9.78E+12	7			
9	9.78E+12	8			
10	1.82E+12	6			

A4				1234567891012		
	A	B	C	D	E	F
1	1.23E+12	2				
2	9.78E+12	3				
3	1.23E+12	1				
4	1.23E+12	4				
5	1.23E+12	1				
6	1.23E+12	5				
7	9.78E+12	6				
8	9.78E+12	7				
9	9.78E+12	8				

YAZAR GÜNCELLE

```
void yazar_gun(YAZ* head);
```

Parametre olarak yazar listesinin baş elemanını alır. Alınan id değerine göre listede arama yaparak isim ve soy isime üzerinde güncelleme yapılır.

```
===YAZAR GUNCELLE===
Guncellemek istediginiz yazar id:1
Yeni isim:Mike
Yeni Soyad:Tyson
```

```
YAZAR
ID:1, Ad:Mike , Soy Ad: Tyson
```

1	Mike	Tyson
2	Stephen	Boyd
3	Kalyanmo	Deb
5	Scott	Kirkpatrick
6	Lieven	Vandenberghe
7	Fabian	Pedregosa
8	Jorge	Nocedal

1	Michael	Jordan
2	Stephen	Boyd
3	Kalyanmo	Deb
4	David	Johnson
5	Scott	Kirkpatrick
6	Lieven	Vandenberghe
7	Fabian	Pedregosa
8	Jorge	Nocedal

YAZAR BİLGİSİ GÖRÜNTÜLEME

```
void print_iliski(KITAP* headk, YAZ* heady, KITAP_YAZ* book_auth, int N)
```

Parametre olarak yazar listesinin baş elemanını, kitap listesinin baş elemanını kitap yazar ilişkisinin tutulduğu diziyi alır. id üzerinden arama yapar ve book_auth dizisi üzerinde o id ye sahip eşleşmelerin kitap bilgilerini kitap listesinde arayarak tüm bilgileri yazdırır.

```
===YAZAR BİLGİSİ===  
  
Kitaplari Goruntulenecek yazar ID:1  
ID:1, Ad:Mike , Soy Ad: Tyson  
KITAP  
AD:Fizik 1, ISBN:1234567891012, Adet:4  
EtiketNo:1234567891012_0 , Durum:Rafta  
EtiketNo:1234567891012_1 , Durum:18011001  
EtiketNo:1234567891012_2 , Durum:17011020  
EtiketNo:1234567891012_3 , Durum:Rafta  
KITAP  
AD:Fizik 2, ISBN:1234567891013, Adet:3  
EtiketNo:1234567891013_0 , Durum:Rafta  
EtiketNo:1234567891013_1 , Durum:Rafta  
EtiketNo:1234567891013_2 , Durum:Rafta
```

PROJE HAKKINDA GERİ BİLDİRİM

Dönem içerisindeki yoğunlukla birlikte uğraştırıcı bir proje olduğunu düşünüyorum. Fonksiyonların uyumlu bir şekilde tasarlanması gerekiyordu. Bundan dolayı ara ara geriye dönüp düzeltmeler yapma ihtiyacı duydum. Bu konudaki eksikliklerimi fark ettim. Kaynak olarak ders materyalleri yeterliydi. Ahmet Hoca'ya ve diğer Asistan Hoca'lara bu dönem için teşekkür ederim.