

Tervezési minta:  
Látogató (Könyv-Példányszám)

```
for i = 0; i < |kölcsönzések|; i++  
for j = 0; j < |kölcsönzések[i].könyvek|; j++  
if kölcsönzések[i].könyvek[j].title == title then  
k.Visszavesz(this, kölcsönzések[i].könyvek[j])  
kölcsönzések[i].RemoveBook(kölcsönzések[i].könyvek[j])
```

```
kivettKönyvek = k.Kiad(this,könyvek)  
if !( |kivettKönyvek.könyvek| ) > 0 then  
NoBooksToBorrowException  
kölcsönzések.Add(kivettKönyvek)
```

```
if tagok.Contains(s2) then  
AlreadyMemberException  
else  
tagok.Add(s2)
```

```
if !(tagok.Contains(s2)) then  
NotMemberException()  
  
if |könyvek| > 5 then  
TooManyBooksException()  
  
kölcsönzöttKönyvek := new List<Könyv>()  
currentDate := DateTime.Today  
lejárat := currentDate.AddMonths(borrowTimeInMonths)  
  
for i = 0; i < |könyvek|; i++  
for j = 0; j < |this.könyvek|; j++  
if könyvek[i] == this.könyvek[j].title then  
if !(könyvek[j].kölcsönzött) then  
kölcsönzöttKönyvek.Add(this.könyvek[j])  
this.könyvek[j].kolcsonzott = true;  
else  
this.könyvek[j] + "is already borrowed"
```

```
k.id = |könyvek| + 1  
k.kölcsönzött = false  
könyvek.Add(k)
```

```
for i = 0; i < |könyvek|; i++;  
if könyvek[i].title == k.title then  
könyvek[i].kolcsonzott = false  
done
```

```
return getDiffDays(DateTime.Parse(visszaDate)) * (példányszám.Érték(this))
```

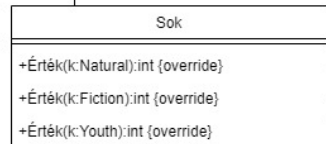
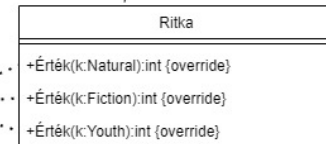
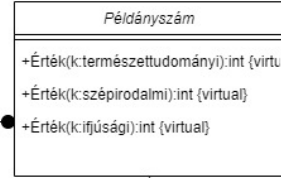
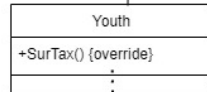
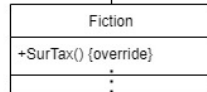
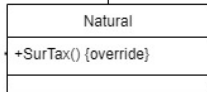
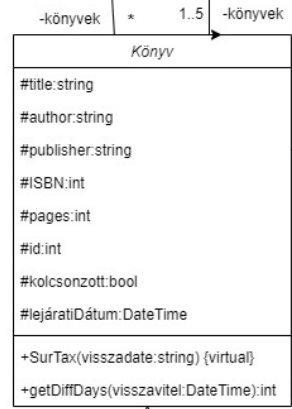
```
return getDiffDays(DateTime.Parse(visszaDate)) * (példányszám.Érték(this))
```

```
return getDiffDays(DateTime.Parse(visszaDate)) * (példányszám.Érték(this))
```

k.Beirat(this)

```
-név:string  
+Beiratkozik(k:Könyvtár)  
+Kölcsönöz(k:Könyvtár, könyvek:List<Könyv>  
)  
+Visszavisz(k:Könyvtár, title:string)
```

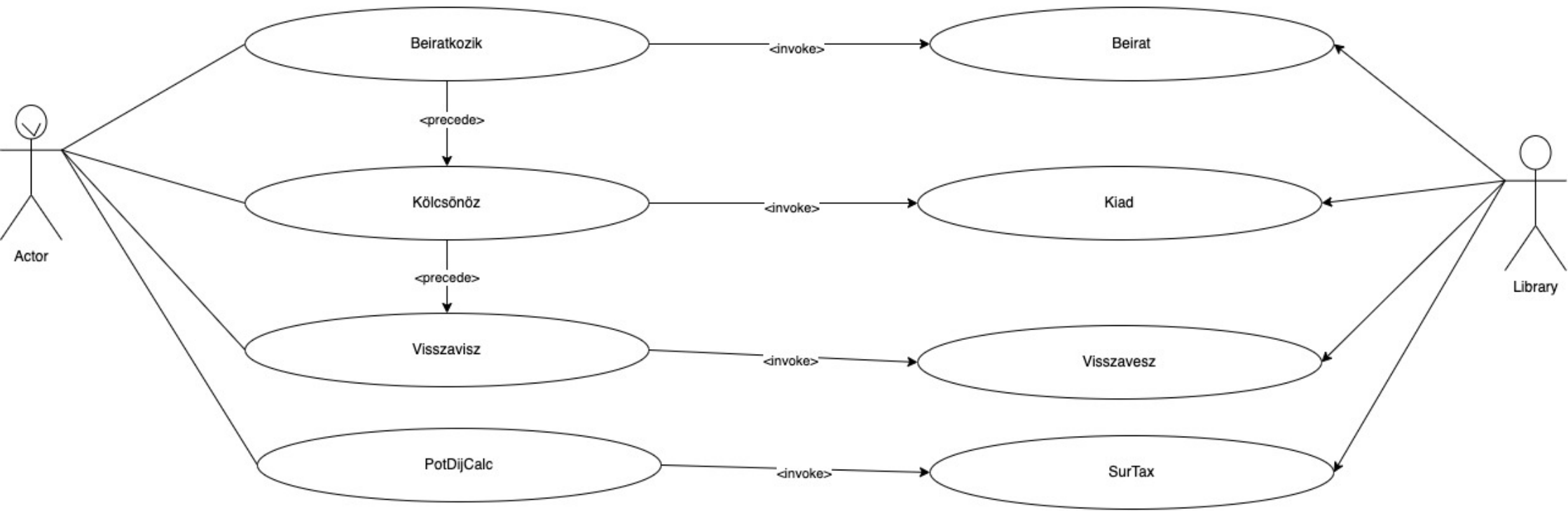
```
this.könyvek.Remove(k)
```



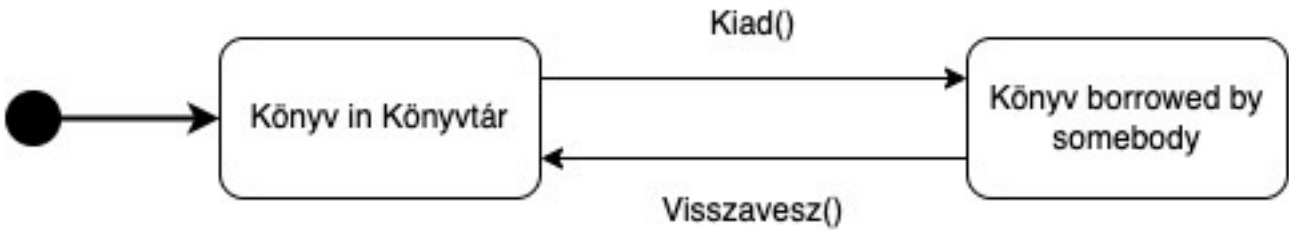
return 30  
return 50  
return 100

return 10  
return 10  
return 20

# use-case diagram



Állapotgép



Objektum diagram

