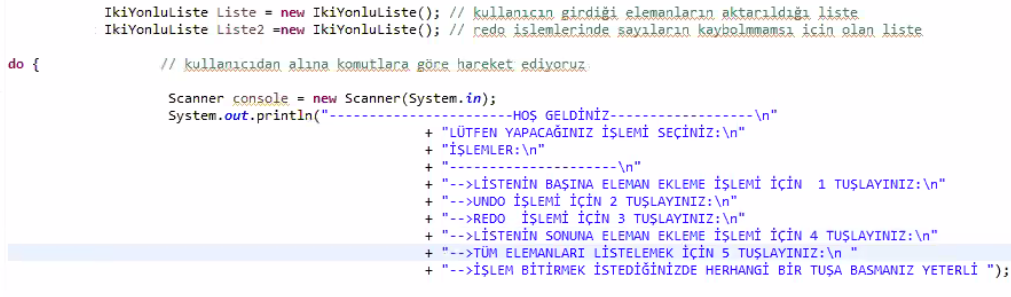
ELİF ÖKSÜZALİ 180303010

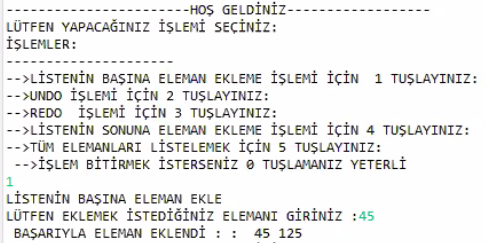
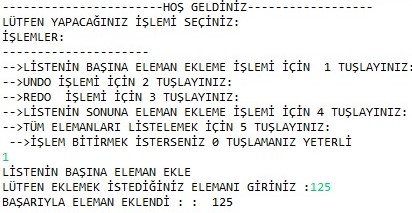
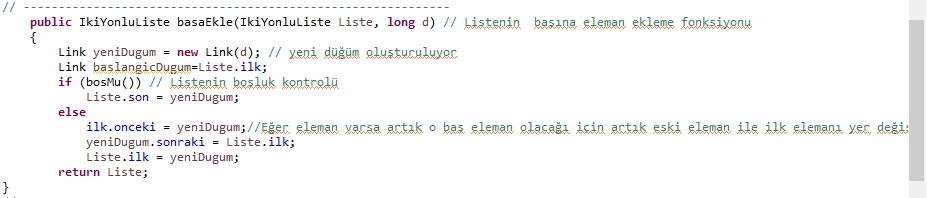
SELMA GÜNDOĞDU 190303054

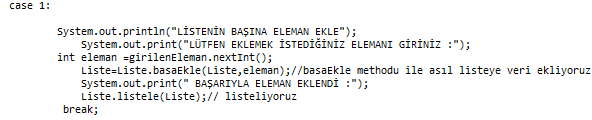
VERİ YAPILARI VE ALGORİTMALARI

Amaç= Bu proje kapsamında kullanıcıdan alınan girdileri bağlantılı liste veri yapısında saklayacak ve kullanıcın **undo ve** **redo** işlemlerini yapmasına olanak sağlayacak bir program yazmak.



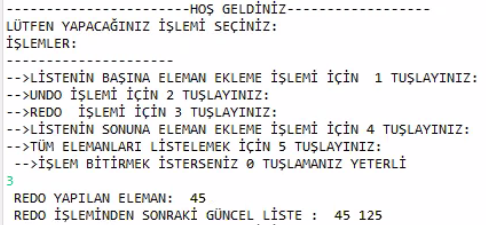
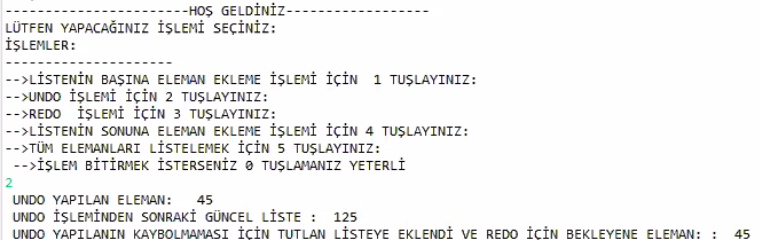
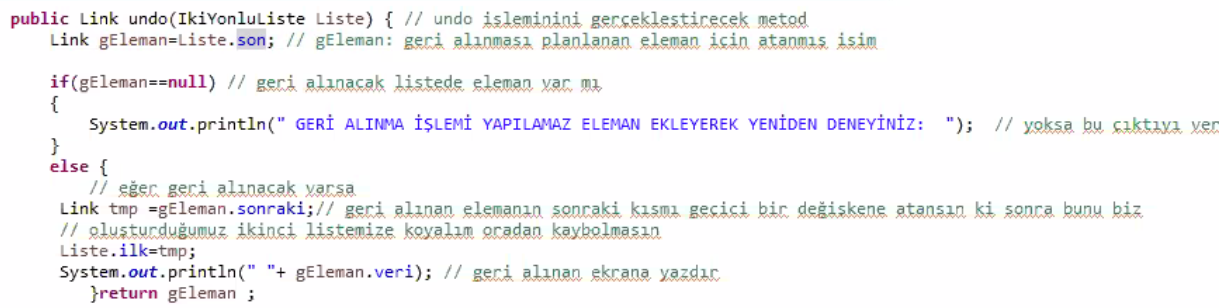
Kullanıcıdan alınan komutlara göre hareket etmesi için switch case yapısı için kullanıcı ekranında çıkan yönlendirme bloklarını oluşturduk. Dikkat edeceğimiz gibi iki tane iki yönlü liste tanımlayın. Bunlardan liste olan kullanıcıdan alınan verilerin kaydolduğu, liste2 ise undo işlemi yapıldığında bu elemanlar kaybolmasın redo metoduyla geri alabilmek için listeden çıkarılan ve liste2 ye aktarılan verilerin tutulduğu yerdir.

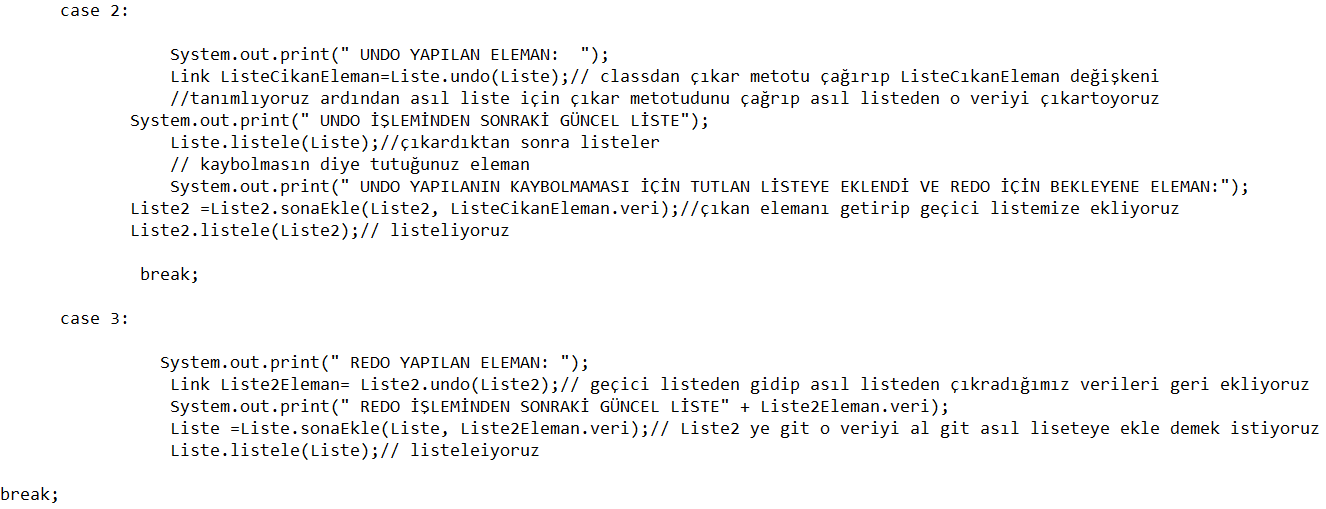




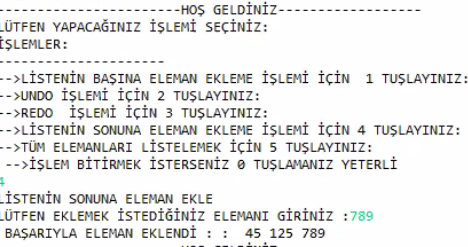
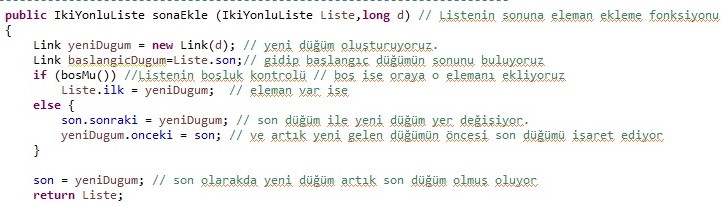
Case 1: yaptığı iş kullanıcı başa eleman eklemek istediğinde class da oluşturduğumuz basaEkle metodunu çağırarak işlem yaptırıp o işlem sonucunda alına veriyi liste adına oluşturmuş olduğumuz kısma ekleyip listeyi güncelleyip ekran çıktısı olarak da ekrana başa eklenmiş hali ile verileri verme işini yapar.

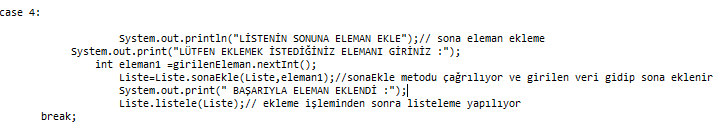
basaEkle: Önce yeni bir düğüm oluşturduk oluşan bu düğüm için önce listenin boş olup olmadığını yani hiç eleman olup olmadığını kontrol ettirdik. Eğer hiç eleman yoksa yeni oluşan bu düğüm baş eleman olmasını söyledik. Eğer daha önceden eklenmiş eleman varsa mantık olarak şöyle o elemanın önceki kısmı yeni gelen elemanı işaret edecek yeni gelenin sonrası ise ilk elemanın öncesini , listenin ilk elemanı da artık yeni gelen düğümü işaret edeceğini söylüyor .



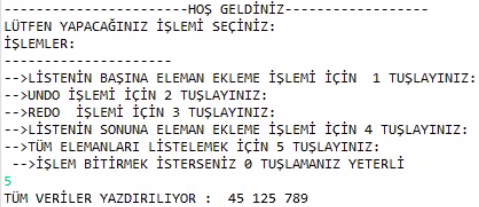
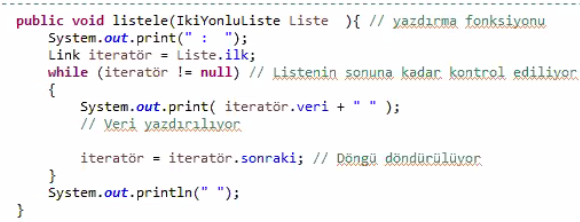


gEleman: Geri alınması planlanan eleman için atanmış isimdir bu değişken ile class da undo metoduna şu işi yaptırıyoruz. Git listeyi kontrol et (asıl liste = Liste) eleman var mı yoksa print içindeki ifadeyi ekran ver eğer o listede geri alınacak eleman varsa geri alınan elemanın sonraki kısmı geçici bir değişkene atansın ki sonra bunu biz oluşturduğumuz ikinci listemize koyalım oradan kaybolmasın geri aldığımız bu sayının Liste.ilk=tmp atıyoruz ki bu sayede artık tmp değişkeninin sonrakide ilk oluyor böylece gelecek olan eleman listeden geri alınmış oluyor. Veriyi yazdırıyoruz return ifadesiyle tekrar tekrar geri alınma işlemin, tekrar edebilmesi için çıkarılacak olan elemanı döndürüyoruz. Kullanıcı istediği kadar geri alma işlemi yapar.Case 2: Asıl listeye git o elamanı çıkar sonra listeyi güncelle ve listele metodu ile yeniden ekrana son halini çıktısını veriyoruz.Case 3: Geçici olarak çıkartıp tutuğumuz o elemanı git oradan çıkar ve asıl listeye eklemesi gerektiğini söylüyoruz bu case kısmı o işe yarıyor.





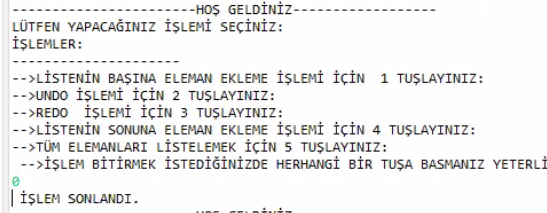
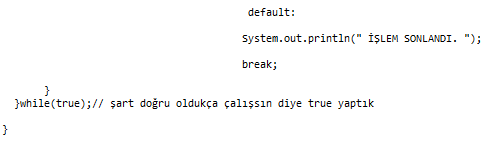
Listenin sonuna eleman ekleme fonksiyonda öncelikle bir düğüm oluştururuz. Elimizdeki listenin dolu yada boş olma durumunu kontrol ederek sona eleman ekleme motodunu çağırırız ve listemizin sonuna elemanı düğümü ekleriz. Yeni eklenen eleman listemizin son elemanı düğümü olmuş olur.



C:\Users\LENOVO\Pictures\Screenshots\Ekran Görüntüsü (138).png

Case 5: Kullanıcı eklene tüm elemanları listelemesi için yaptığı çağrışımı alana bunu ekrana çıktı olarak veren kısmı

Listele metodu: Tüm listeyi gezerek o elemanları tek tek ekrana çıktı verebilmesi için oluşturulmuş metod.



Kullanıcı işi bittiğinde programı sonlandırma isteği için yapılmış kısımdır. Program eksiklikleri:

* 5 redo işlemi yapılmadı
* Undo işlemlerinde çıkarılan eleman listeleme metodu çağrılınca normal listede gözüküyor
* Eklenme sırasına göre listele işlemi tam anlamıyla sona elemen sadece ekler iseniz çalışır. Araya veya bir başa bir sona eleman eklemek isteyip bunların ekleniş sırasına göre listeleme yapamıyoruz.
* Undo işleminde son eleman çıkıyor ama basaEkle metodu çalışırsa basa eklene son eleman ne ise onu çıkartıyor siz en son sona eklerseniz case deki undo metodunu basaEkle yerini değişmeniz gerekir.