

Kullanıcının programa öncelikle giriş yaptığında göreceği ana menünün fonksiyonu şu şekilde oluşturulup istenilen seçim girildikten sonra oluşturulmuş olan fonksiyonların çağırımı yapılmaktadır. Parametrelili fonksiyonlarda ise önce girdi alınması sağlanmaktadır.

```
//Kullanıcının işlemleri yapacağı ana menüye ait fonksiyon.
void AnaMenu()
{
    string urunum, projelum, donem;
    int secim;
    cout << "Ne işlem yapmak istiyorsunuz?" << endl;
    cout << "1) Urun ekleme" << endl;
    cout << "2) Urun silme" << endl;
    cout << "3) Urun arama" << endl;
    cout << "4) Urunleri listeleme" << endl;
    cout << "5) Proje ekleme" << endl;
    cout << "6) Proje silme" << endl;
    cout << "7) Proje arama" << endl;
    cout << "8) Projeleri listeleme" << endl;
    cout << "9) Hakedis ekleme" << endl;
    cout << "10) Hakedis urun silme" << endl;
    cout << "11) Hakedis urun guncelleme" << endl;
    cout << "12) Hakedis kullanılan urun arama" << endl;
    cout << "13) Hakedis urunlerde arama" << endl;
    cout << "14) Proje no'ya göre hakedislerin total tutar görüntüleme" << endl;
    cout << "15) Doneme göre hakedis total tutari görüntüleme" << endl;
    cout << "16) Cikis" << endl;

    cin >> secim;
    switch (secim)
    {
        case 1:
            UrunEkle();
            break;
        case 2:
            cout << "Silmek istediginiz urun no yaziniz.\n"; cin >> urunum;
            UrunSil(urunum);
            break;
        case 3:
            cout << "Aramak istediginiz urun no yaziniz.\n"; cin >> urunum;
            UrunAra(urunum);
            break;
        case 4:
            UrunListele();
            break;
        case 5:
            ProjeEkle();
            break;
        case 6:
            cout << "Silmek istediginiz proje no yaziniz.\n"; cin >> projelum;
            ProjeSil(projelum);
            break;
        case 7:
            cout << "Aramak istediginiz proje no yaziniz.\n"; cin >> projelum;
            ProjeAra(projelum);
            break;
        case 8:
            Projelistele();
            break;
        case 9:
            HakedisEkle();
            break;
        case 10:
            cout << "Proje no giriniz.\n"; cin >> projelum;
            HakedisUrunSil(projelum);
            break;
        case 11:
            cout << "Proje no giriniz.\n"; cin >> projelum;
            HakedisUrunGuncelle(projelum);
            break;
        case 12:
            cout << "Kullanilan urunleri gormek istediginiz proje no giriniz. (Proje no'ya ait eger birden fazla donem var ise tumu goruntulenecektir.)\n"; cin >> projelum;
            HakedisKullanilanUrunListele(projelum);
            break;
        case 13:
            cout << "Hakedislerde bulunan aramak istediginiz urun no giriniz.\n"; cin >> urunum;
            HakedisUrunlerdeAra(urunum);
            break;
        case 14:
            cout << "Total tutari gormek istediginiz proje no giriniz.\n"; cin >> projelum;
            HakedisProjeNodanTotalTutariListele(projelum);
            break;
        case 15:
            cout << "Total tutari gormek istediginiz donemi giriniz.\n"; cin >> donem;
            HakedisDonemdenTotalTutariListele(donem);
            break;
        case 16:
            exit(0);
            break;
        default:
            cout << "Hatali tuslama yaptiniz. Lutfen tekrar deneyiniz.\n";
            break;
    }
}
```

Görüntüsü ise şu şekilde olmaktadır.

```
Ne islem yapmak istiyorsunuz?
1) Urun ekleme
2) Urun silme
3) Urun arama
4) Urunleri listeleme
5) Proje ekleme
6) Proje silme
7) Proje arama
8) Projeleri listeleme
9) Hakedis ekleme
10) Hakedis urun silme
11) Hakedis urun guncelleme
12) Hakedis kullanılan urun arama
13) Hakedis urunlerde arama
14) Proje no'ya göre hakedislerin total tutar görüntüleme
15) Doneme göre hakedis total tutari görüntüleme
16) Cikis
```

1) Urun Ekleme

```
//insaatmalzemeleri.txt için gerekli olan girdilerin tanımlanması.
struct Urun
{
    string urunNo, urunAdi, urunBirim, urunMarka, urunBirimFiyat;
}urun;
```

Ürün dosyasına girişi yapılacak bilgiler için struct yapısı oluşturulmuş durumda. UrunEkle fonksiyonu ile işlem sağlanmaktadır. Fonksiyonda gerekli kontroller yapılarak yorum satırında yazmakta olan işlemler yapılmaktadır.

```
//insaatmalzemeleri.txt dosyasına urunun eklenmesini saglayan fonksiyon.
void UrunEkle()
{
    string urunNum;
    fstream Dosya;
    Dosya.open("insaatmalzemeleri.txt", ios::out | ios::app);
    cout << "\nUrun No :"; cin >> urunNum;
    //Urunler dosyasinda girilen urun no'nun bulunup bulunmadigi sorgulaniyor. Yok ise diger bilgiler alinarak dosyaya kaydediliyor.
    if (urunNo_Kontrol(urunNum) == 1)
    {
        cout << "Bu urun no mevcut. Lutfen yeniden deneyiniz.\n";
    }
    else
    {
        urun.urunNo = urunNum;
        cin.ignore();
        cout << "\nUrun Adi :"; getline(cin, urun.urunAdi);
        cout << "\nUrun Birimi :"; cin >> urun.urunBirim;
        cout << "\nUrun Birim Fiyat :"; cin >> urun.urunBirimFiyat;
        cin.ignore();
        cout << "\nUrun Marka :"; getline(cin, urun.urunMarka);
        Dosya << urun.urunNo << "," << urun.urunAdi << "," << urun.urunBirim << "," << urun.urunBirimFiyat << "," << urun.urunMarka << "\n";
        cout << "Urun eklenmistir.\n";
    }
    Dosya.close();
}
```

Ürün numarasının insaatmalzemeleri.txt dosyasında bulunduğu durumda konsol çıktısı şöyledir;

```
Urun No :1001
Bu urun no mevcut. Lutfen yeniden deneyiniz.
```

1.

Ürün numarası mevcut değil ise gerekli bilgiler alınarak ürün kaydı yapılmaktadır. Konsol çıktısı şöyledir;

```
Urun No :1002
Urun Adi :Sucuk
Urun Birimi :kg
Urun Birim Fiyat :7
Urun Marka :Sahin
Urun eklenmistir.

Ana menüye donmek için herhangi bir tusuna basiniz.
Press any key to continue . . .
```

1. urunNo_Kontrol fonk.

2) Urun Silme

UrunSil fonksiyonu ile işlem sağlanmaktadır. Fonksiyonda gerekli kontroller yapılarak yorum satırında yazmakta olan işlemler yapılmaktadır.

```
//insaatzemeleri.txt dosyasından urun silinebilmesini saglayan fonksiyon.
void UrunSil(string urunSilNo)
{
    fstream Dosya;
    fstream YeniDosya;
    string satir;
    Dosya.open("insaatzemeleri.txt", ios::in | ios::app);
    YeniDosya.open("insaatzemeleriyeeni.txt", ios::out | ios::app);
    bool kontrol = 0;
    //Urunun olup olmadigi kontrolu yapiliyor. Var ise tum satirlar taranarak urun no'nun dosyadaki urun no'ya esit olmadigi durumdaki satirlari gecici olarak olusturulan yeni dosyaya yazdiriyor.
    if (urunNo_Kontrol(urunSilNo) == 0)
    {
        cout << "Urun no zaten mevcut degil.\n";
    }
    else
    {
        kontrol = 1; //Urun no'nun mevcut olup, islem yapildigini ifade ederek 1 degeri aliyor.
        while (getline(Dosya, satir))
        {
            stringstream ss(satir);
            getline(ss, urun.urunNo, ',');
            getline(ss, urun.urunAdi, ',');
            getline(ss, urun.urunBirim, ',');
            getline(ss, urun.urunBirimFiyat, ',');
            getline(ss, urun.urunMarka, ',');
            if (urun.urunNo != urunSilNo)
            {
                YeniDosya << urun.urunNo << "," << urun.urunAdi << "," << urun.urunBirim << "," << urun.urunBirimFiyat << "," << urun.urunMarka << "\n";
            }
        }
    }
    Dosya.close();
    YeniDosya.close();
    //kayit girildiyse eski dosya siliniyor ve gecici olarak olusturulan dosya eski dosyasinin adini aliyor.
    if (kontrol == 1)
    {
        remove("insaatzemeleri.txt");
        rename("insaatzemeleriyeeni.txt", "insaatzemeleri.txt");
        cout << "Urun silindi.\n";
    }
    else
    {
        remove("insaatzemeleriyeeni.txt");
    }
}
```

Silinmesi istenilen ürün no'nun zaten ürünlerin bulunduğu dosyada zaten bulunmadığı durumda konsol çıktısı şöyledir;

```
Silmek istediginiz urun no yaziniz.
1000
Urun no zaten mevcut degil.
```

1.

Ürün numarasının bulunduğu durumda ise ürünün silindiği konsol çıktısı şöyledir;

```
Silmek istediginiz urun no yaziniz.
1004
Urun silindi.

Ana menüye donmek icin herhangi bir tusuna basiniz.
Press any key to continue . . .
```

1. urunNo_Kontrol fonk.

3) Urun Arama

UrunAra fonksiyonu ile işlem sağlanmaktadır. Fonksiyonda gerekli kontroller yapılarak yorum satırında yazmakta olan işlemler yapılmaktadır.

```
//insaatzemeleri.txt dosyasından urun aratılabilmesini saglayan fonksiyon.
void UrunAra(string urunBulNo)
{
    fstream Dosya;
    string satir;
    Dosya.open("insaatzemeleri.txt", ios::in | ios::app);
    //Urunun olup olmadigi sorgulanıyor. Urun var ise tum satirlar okunarak girilen urun no'nun dosyadaki urun no'ya esit oldugu satir ekrana yazdiriliyor.
    if (urunNo_Kontrol(urunBulNo) == 0)
    {
        cout << "Urun no mevcut degil.\n";
    }
    else
    {
        while (getline(Dosya, satir))
        {
            stringstream ss(satir);
            getline(ss, urun.urunNo, ',');
            getline(ss, urun.urunAdi, ',');
            getline(ss, urun.urunBirim, ',');
            getline(ss, urun.urunBirimFiyat, ',');
            getline(ss, urun.urunMarka, ',');
            if (urun.urunNo == urunBulNo)
            {
                cout << left << setw(12) << "UrunNo" << setw(17) << "UrunAdi" << setw(15) << "UrunBirimi" << setw(20) << "UrunBirimFiyat" << setw(15) << "UrunMarka" << endl;
                cout << left << setw(12) << urun.urunNo << setw(17) << urun.urunAdi << setw(15) << urun.urunBirim << setw(20) << urun.urunBirimFiyat << setw(15) << urun.urunMarka << "\n";
            }
        }
    }
    Dosya.close();
}
```

Aranılması istenilen ürün no'nun ürünlerin bulunduğu dosyada bulunmaması durumunda konsol çıktısı şöyledir;

```
Aramak istediginiz urun no yaziniz.
1000
Urun no mevcut degil.

Ana menüye donmek için herhangi bir tusuna basiniz.
Press any key to continue . . .
```

1.

Ürün no'nun bulunması durumunda konsol çıktısı ise şöyledir;

```
Aramak istediginiz urun no yaziniz.
1005
UrunNo      UrunAdi      UrunBirimi      UrunBirimFiyat      UrunMarka
-----
1005      Dis Macunu      adet      18      Colgate
```

1. urunNo_Kontrol fonk.

4) Ürünleri Listeleme

UrunListele fonksiyonu ile işlem sağlanmaktadır. Fonksiyonda yorum satırlarında yazmakta olan işlemler yapılarak ürünlerin listelenmesi sağlanmaktadır.

```
//insaatzemeleri.txt dosyasından tum urunlerin listelenebilmesini saglayan fonksiyon. Tum satirlar okunarak ekrana yazdiriliyor.
void UrunListele()
{
    fstream Dosya;
    string satir;
    Dosya.open("insaatzemeleri.txt", ios::in | ios::app);
    cout << left << setw(12) << "UrunNo" << setw(17) << "UrunAdi" << setw(15) << "UrunBirimi" << setw(20) << "UrunBirimFiyat" << setw(15) << "UrunMarka" << endl;
    cout << left << setw(12) << "-----" << endl;
    while (getline(Dosya, satir))
    {
        stringstream ss(satir);
        getline(ss, urun.urunNo, ',');
        getline(ss, urun.urunAdi, ',');
        getline(ss, urun.urunBirim, ',');
        getline(ss, urun.urunBirimFiyat, ',');
        getline(ss, urun.urunMarka, ',');
        cout << left << setw(12) << urun.urunNo << setw(17) << urun.urunAdi << setw(15) << urun.urunBirim << setw(20) << urun.urunBirimFiyat << setw(15) << urun.urunMarka << "\n";
    }
    Dosya.close();
}
```

Konsol çıktısı ise şöyledir;

UrunNo	UrunAdi	UrunBirimi	UrunBirimFiyat	UrunMarka
1001	Lastik	adet	4	Lassa
1002	Sucuk	kg	7	Sahin
1003	Televizyon	adet	11	Vestel
1005	Dis Macunu	adet	18	Colgate
1006	Outdoor Bot	adet	21	The North Face
1007	Notebook	adet	25	Lenovo
1008	Araba	adet	28	Tesla
1009	Su	lt	32	Erikli
1010	Fayans	m2	37	Bien Seramik
1004	Cep Telefonu	adet	14	Apple

5) Proje Ekleme

```
//projeler.txt için gerekli olan girdilerin tanımlanması.
struct Proje
{
    string projeNo, projeAdi, projeFirma, projeSorumlu, projeKontrolor;
}proje;
```

Proje dosyasına girişi yapılacak bilgiler için struct yapısı oluşturulmuş durumda. ProjeEkle fonksiyonu ile işlem sağlanmaktadır. Fonksiyonda gerekli kontroller yapılarak yorum satırında yazmakta olan işlemler yapılmaktadır.

```
//projeler.txt dosyasına projenin eklenmesini saglayan fonksiyon.
void ProjeEkle()
{
    string projeNum;
    fstream Dosya;
    Dosya.open("projeler.txt", ios::out | ios::app);
    cout << "\nProje No :"; cin >> projeNum;
    //Proje numarasının projeler.txt dosyasında olup olmadığını kontrolu yapılıyor. Yok ise diğer bilgiler alınarak dosyaya kaydediliyor.
    if (ProjeNo_kontrol(projeNum) == 1)
    {
        cout << "Proje no zaten mevcut. Lutfen yeniden deneyiniz.\n";
    }
    else
    {
        proje.projeNo = projeNum;
        cin.ignore();
        cout << "\nProje Adi :"; getline(cin, proje.projeAdi);
        cout << "\nProje Firma :"; getline(cin, proje.projeFirma);
        cout << "\nProje Sorumlusu :"; getline(cin, proje.projeSorumlu);
        cout << "\nProje Kontrolör :"; getline(cin, proje.projeKontrolor);
        Dosya << proje.projeNo << "," << proje.projeAdi << "," << proje.projeFirma << "," << proje.projeSorumlu << "," << proje.projeKontrolor << "\n";
        cout << "Proje eklenmistir.\n";
    }
    Dosya.close();
}
```

Proje numarasının projeler.txt dosyasında bulunduğu durumda konsol çıktısı şöyledir;

```
Proje No :2001
Proje no zaten mevcut. Lutfen yeniden deneyiniz.

Ana menüye donmek icin herhangi bir tusuna basiniz.
1. Press any key to continue . . .
```

Proje numarası mevcut değil ise gerekli bilgiler alınarak proje kaydı yapılmaktadır. Konsol çıktısı şöyledir;

```
Proje No :2004
Proje Adi :Metalurji
Proje Firma :Hepsiburada
Proje Sorumlusu :Cem Akca
Proje Kontrollor :Bugra Guner
Proje eklenmistir.
```

1. ProjeNo_Kontrol fonk.

6) Proje Silme

ProjeSil fonksiyonu ile işlem sağlanmaktadır. Fonksiyonda gerekli kontroller yapılarak yorum satırında yazmakta olan işlemler yapılmaktadır.

```
//projeler.txt dosyasından proje silinebilmesini sağlayan fonksiyon.
void ProjeSil(string projeSilNo)
{
    fstream Dosya;
    fstream YeniDosya;
    string satir;
    Dosya.open("projeler.txt", ios::in | ios::app);
    YeniDosya.open("projeleryeni.txt", ios::out | ios::app);
    bool kontrol = 0;
    //Projenin olup olmadigi kontrolu yapiliyor. Var ise tum satirlar taranarak proje no'nun dosyadaki proje no'ya esit olmadigi durumdaki satirlari gecici olarak olusturulan yeni dosyaya yazdiriyor.
    if (ProjeNo_Kontrol(projeSilNo) == 0)
    {
        cout << "Proje no zaten mevcut degil.\n";
    }
    else
    {
        if (HakedisProjeNo_Kontrol(projeSilNo) == 0)
        {
            kontrol = 1; //Proje no'nun mevcut olup, islem yapildigini ifade ederek 1 degeri aliyor.
            while (getline(Dosya, satir))
            {
                stringstream ss(satir);
                getline(ss, proje.projeNo, ',');
                getline(ss, proje.projeAdi, ',');
                getline(ss, proje.projeFirma, ',');
                getline(ss, proje.projeSorumlu, ',');
                getline(ss, proje.projeKontrolor, ',');
                if (proje.projeNo != projeSilNo)
                {
                    YeniDosya << proje.projeNo << "," << proje.projeAdi << "," << proje.projeFirma << "," << proje.projeSorumlu << "," << proje.projeKontrolor << "\n";
                }
            }
        }
        else
        {
            cout << "Proje no hakedis listesinde oldugundan silinemez.\n";
        }
    }
    Dosya.close();
    YeniDosya.close();
    //Kayit girildiyse eski dosya siliniyor ve gecici olarak olusturulan dosya eski dosyasinin adini aliyor.
    if (kontrol == 1)
    {
        remove("projeler.txt");
        rename("projeleryeni.txt", "projeler.txt");
        cout << "Proje silindi.\n";
    }
    else
    {
        remove("projeleryeni.txt");
    }
}
```

Proje numarasının projeler dosyasında bulunmadığı durumda konsol çıktısı şöyledir;

```
Silmek istediginiz proje no yaziniz.
2000
Proje no zaten mevcut degil.
```

1.

Proje numarasının hakediş dosyasında bulunduğu durumda konsol çıktısı şöyledir;

```
Silmek istediginiz proje no yaziniz.
2006
Proje no hakedis listesinde oldugundan silinemez.
```

2.

Projenin silindiği durumda ise konsol çıktısı şöyledir;

```
Silmek istediginiz proje no yaziniz.
2010
Proje silindi.
```

1. ProjeNo_Kontrol fonk.
2. HakedisProjeNo_Kontrol fonk.

7) Proje Arama

ProjeAra fonksiyonu ile işlem sağlanmaktadır. Fonksiyonda gerekli kontroller yapılarak yorum satırında yazmakta olan işlemler yapılmaktadır.

```
//projeler.txt dosyasından proje aratılabilmesini sağlayan fonksiyon.
void ProjeAra(string projeBulNo)
{
    fstream Dosya;
    string satir;
    Dosya.open("projeler.txt", ios::in | ios::app);
    //Projenin olup olmadigi sorgulanıyor. Proje var ise tum satirlar okunarak girilen proje no'nun dosyadaki proje no'ya esit oldugu satir ekrana yazdiriliyor.
    if (ProjeNo_Kontrol(projeBulNo) == 0)
    {
        cout << "Proje no mevcut degil.\n";
    }
    else
    {
        while (getline(Dosya, satir))
        {
            stringstream ss(satir);
            getline(ss, proje.projeNo, ',');
            getline(ss, proje.projeAdi, ',');
            getline(ss, proje.projeFirma, ',');
            getline(ss, proje.projeSorumlu, ',');
            getline(ss, proje.projeKontrolor, ',');
            if (proje.projeNo == projeBulNo)
            {
                cout << left << setw(12) << "ProjeNo" << setw(17) << "ProjeAdi" << setw(25) << "ProjeFirma" << setw(25) << "ProjeSorumlusu" << setw(15) << "ProjeKontroller" << endl;
                cout << "-----" << endl;
                cout << left << setw(12) << proje.projeNo << setw(17) << proje.projeAdi << setw(25) << proje.projeFirma << setw(25) << proje.projeSorumlu << setw(15) << proje.projeKontrolor << "\n";
            }
        }
        Dosya.close();
    }
}
```

Aranılması istenilen proje no'nun projelerin bulunduğu dosyada bulunmaması durumunda konsol çıktısı şöyledir;

```
Aramak istediginiz proje no yaziniz.
2000
Proje no mevcut degil.
```

- 1.

Proje no'nun bulunması durumunda konsol çıktısı ise şöyledir;

```
Aramak istediginiz proje no yaziniz.
2009
ProjeNo      ProjeAdi      ProjeFirma      ProjeSorumlusu      ProjeKontroller
-----
2009      Akilli Ses      Asus      Emek Derun      Gizem Balkan
```

1. ProjeNo_Kontrol fonk.

8) Projeleri Listele

ProjeListele fonksiyonu ile işlem sağlanmaktadır. Fonksiyonda yorum satırlarında yazmakta olan işlemler yapılarak ürünlerin listelenmesi sağlanmaktadır.

```
//projeler.txt dosyasından tüm projelerin listelenebilmesini sağlayan fonksiyon. Tüm satırlar okunarak ekrana yazdırılıyor.
void ProjeListele()
{
    fstream Dosya;
    string satir;
    Dosya.open("projeler.txt", ios::in | ios::app);
    cout << left << setw(12) << "Projeno" << setw(17) << "ProjeAdi" << setw(25) << "ProjeFirma" << setw(25) << "ProjeSorumlusu" << setw(15) << "ProjeKontroller" << endl;
    cout << "-----" << endl;
    while (getline(Dosya, satir))
    {
        stringstream ss(satir);
        getline(ss, proje.projeno, ',');
        getline(ss, proje.projeadi, ',');
        getline(ss, proje.projefirma, ',');
        getline(ss, proje.projesorumlu, ',');
        getline(ss, proje.projekontrolor, ',');
        cout << left << setw(12) << proje.projeno << setw(17) << proje.projeadi << setw(25) << proje.projefirma << setw(25) << proje.projesorumlu << setw(15) << proje.projekontrolor << "\n";
    }
    Dosya.close();
}
```

Konsol çıktısı ise şöyledir;

Projeno	ProjeAdi	ProjeFirma	ProjeSorumlusu	ProjeKontroller
2001	Yazılım	Apple	Nermin Ciftci	Gulcin Celik
2002	Egitim	HP	Didem Ozcimen	Meyrem Vehapi
2003	Kimya	Trendyol	Sevil Yucel	Elif Mete
2004	Metalurji	Hepsiburada	Cem Akca	Bugra Guner
2005	Insaat	Lenovo	Nurcan Tugrul	Ozge Yilmaz
2006	Mimari	Huawei	Murat Sari	Tulay Yildirim
2007	Matematik	Rubenis	Dolunay Sakar	Azize Dizdar
2008	Termik Santral	Boyner	Guray Arslan	Mehdi Ozturk
2009	Akilli Ses	Asus	Emek Derun	Gizem Balkan
2010	Bacakopter	Sony	Haci Ilhan	Hayri Aslan

9) Hakediş Ekleme

```
//hakedisler.txt için gerekli olan girdilerin tanımlanması.
struct Hakedis
{
    string projeNo, donem, urunNo, urunAdet, urunToplamFiyat;
}hakedis;
```

Hakediş dosyasına girişi yapılacak bilgiler için struct yapısı oluşturulmuş durumda. HakedisEkle fonksiyonu ile işlem sağlanmaktadır. Fonksiyonda gerekli kontroller yapılarak yorum satırında yazmakta olan işlemler yapılmaktadır.

```
//hakedisler.txt dosyasına hakedis eklenebilmesini sağlayan fonksiyon.
void HakedisEkle()
{
    string projeNum, urunNum, donem, satir;
    int urunAdet, birimFiyat = 0;
    fstream Dosya;
    vector<string> urunNolar;
    vector<int> urunAdetler;
    vector<int> urunToplamFiyatlar;
    char cevap = 'e';
    Dosya.open("hakedisler.txt", ios::out | ios::app);
    cout << "\nProje No :"; cin >> projeNum;
    //Projeler dosyasında girilen proje no'nun bulunup bulunmadığı sorgulanıyor.
    if (ProjeNo_Kontrol(projeNum) == 0)
    {
        cout << "Proje no proje listesinde mevcut degil.\n";
    }
    else
    {
        cout << "\nHakedis Donemi :"; cin >> donem;
        //Proje no mevcut ise bu proje no'ya ait girilen boyle bir donem mevcut mu onun kontrolu yapiliyor. Mevcutsa bu proje no'ya bu donemin tekrar girilmesi engelleniyor.
        if (HakedisteProjeNoVeDonemVarMi(projeNum, donem) == 0)
        {
            do
            {
                cout << "\nUrun No: "; cin >> urunNum;
                //urunlerin bulunduğu dosyada urun no'nun bulunup bulunmadığı sorgulanıyor. Boyle bir urun var ise adet bilgisi de istenerek urunler dosyasından birim fiyat cekildikten sonra
                //girilen miktar ile carpılarak toplam fiyata yaziliyor. Birden fazla urun girilebilme ihtimaline karsin girilen veriler vectorlere islenerek tutuluyor ve urunno, miktar,
                //toplamfiyat bolunlerinde aralarina "-" isareti bulunuyor.*/
                if (urunNo_Kontrol(urunNum) == 0)
                {
                    cout << "Urun no urun listesinde mevcut degil. Lutfen yeniden deneyiniz.\n";
                }
                else
                {
                    hakedis.projeNo = projeNum;
                    hakedis.donem = donem;
                    urunNolar.push_back(urunNum);
                    cout << "\nurun adet :"; cin >> urunAdet;
                    urunAdetler.push_back(urunAdet);
                    fstream Dosya;
                    Dosya.open("insaatmalzemeleri.txt", ios::in | ios::app);
                    while (getline(Dosya, satir))
                    {
                        stringstream ss(satir);
                        getline(ss, urun.urunNo, ',');
                        getline(ss, urun.urunAdi, ',');
                        getline(ss, urun.urunBirim, ',');
                        ...
                        getline(ss, urun.urunBirimFiyat, ',');
                        getline(ss, urun.urunMarka, ',');
                        if (urun.urunNo == urunNum)
                        {
                            birimFiyat = stoi(urun.urunBirimFiyat);
                        }
                    }
                    Dosya.close();
                    urunToplamFiyatlar.push_back(birimFiyat * urunAdet);
                }
            } while ((cevap != 'h'));
            cout << "\nBu hakedise baska urun kaydi yapacak misiniz?(e/h)\n"; cin >> cevap;
        } while ((cevap != 'h'));
        /*Oncelikle urun listesinde olmayan bir urun girilirse yeniden hakedise baska urun kaydi yapilip yapilmadigi sorulduğundan dolayi boyle bir durumda
        h basip direkt cikilmasi ihtimaline karsin vectore veri girildiginin kontrolu yapiliyor. Girildi ise dosyaya yaziliyor.*/
        if (urunNolar.size() > 0)
        {
            hakedis.urunNo = "";
            hakedis.urunAdet = "";
            hakedis.urunToplamFiyat = "";
            for (int i = 0; i < urunNolar.size(); i++)
            {
                hakedis.urunNo += urunNolar[i] + "-";
                hakedis.urunAdet += to_string(urunAdetler[i]) + "-";
                hakedis.urunToplamFiyat += to_string(urunToplamFiyatlar[i]) + "-";
            }
            Dosya << hakedis.projeNo << "," << hakedis.donem << "," << hakedis.urunNo << "," << hakedis.urunAdet << "," << hakedis.urunToplamFiyat << "\n";
        }
        else
        {
            cout << "Proje no'ya ait boyle bir donem hakediste mevcut.\n";
        }
    }
    urunNolar.clear();
    urunAdetler.clear();
    urunToplamFiyatlar.clear();
    Dosya.close();
}
```

Proje no'nun projelerin bulunduğu dosyada bulunmaması durumunda konsol çıktısı şöyledir;

```
Proje No :2000
Proje no proje listesinde mevcut degil.
```

1.

Proje no'nun projeler dosyasında bulunduğu fakat hakediş listesinde bu proje numarasına ait hakediş döneminin mevcut olduğu durumda konsol çıktısı şöyledir;

```
Proje No :2001
Hakedis Donemi :1
Proje no'ya ait boyle bir donem hakediste mevcut.
```

2.

Proje no'nun projeler dosyasında bulunduğu ve bu proje numarasına ait hakediş döneminin bulunmadığı fakat girilen ürün numarasının ürünler dosyasında bulunmadığı durumda konsol çıktısı şöyledir;

```
Proje No :2001
Hakedis Donemi :1
Urun No:1000
Urun no urun listesinde mevcut degil. Lutfen yeniden deneyiniz.
```

3.

Tüm şartların hakedişe eklenebilir hale geldiği durumda konsol çıktısı şöyledir;

```
Proje No :2001
Hakedis Donemi :1
Urun No:1001
Urun adet :14
Bu hakedise baska urun kaydi yapacak misiniz?(e/h)
e
Urun No:1010
Urun adet :7
Bu hakedise baska urun kaydi yapacak misiniz?(e/h)
h
```

Hakediş sistemlerinde çok kez kullanımı mevcut fonksiyonların kodlaması sayfa sonunda mevcut olacaktır.

1. ProjeNo_Kontrol fonk.
2. HakedisteProjeNoveDonemVarMi fonk.
3. urunNo_Kontrol fonk.

10) Hakediş Ürün Silme

HakedisUrunSil fonksiyonu ile işlem sağlanmaktadır. Fonksiyonda gerekli kontroller yapılarak yorum satırında yazmakta olan işlemler yapılmaktadır.

```
//hakedisler.txt dosyasında hakedise eklenmiş olan ürünlerin silinebilmesini sağlayan fonksiyon.
void HakedisUrunSil(string projelNo)
{
    string urunNum, donem, satir;
    fstream Dosya;
    fstream YeniDosya;
    vector<string>eskiUrunler;
    vector<string>eskiUrunToplamlari;
    vector<string>eskiUrunAdetleri;
    vector<string>yeniUrunler;
    vector<string>yeniUrunToplamlari;
    vector<string>yeniUrunAdetleri;
    char cevap = 'e';
    Dosya.open("hakedisler.txt", ios::in | ios::app);
    YeniDosya.open("hakedislerenyeni.txt", ios::out | ios::app);
    bool kontrol = 0;
    //Projeler dosyasında girilen proje no'nun bulunup bulunmadığı sorgulanıyor.
    if (ProjelNo_Kontrol(projelNo) == 0)
    {
        cout << "Proje no projeler listesinde mevcut değil.\n";
    }
    else
    {
        //Hakedisler dosyasında girilen proje no'nun olup olmadığı kontrolü yapılıyor.
        if (HakedisProjelNo_Kontrol(projelNo) == 1)
        {
            cout << "Hakedis donemini giriniz.\n"; cin >> donem;
            //Proje no hakedisler dosyasında da mevcut ise bu proje no'ya ait girilen böyle bir donem mevcut mu onun kontrolü yapılıyor. Mevcut ise tüm satırlar taranıyor.
            if (HakedisteProjelNoVeDonemVarMi(projelNo, donem) == 1)
            {
                while (getline(Dosya, satir))
                {
                    stringstream ss(satir);
                    getline(ss, hakedis.projelNo, ',');
                    getline(ss, hakedis.donem, ',');
                    getline(ss, hakedis.urunNo, ',');
                    getline(ss, hakedis.urunAdet, ',');
                    getline(ss, hakedis.urunToplamFiyat, ',');
                    //Hakedisler dosyasından tüm satırlar okunarak proje no ve donemin girdilere eşit olduğu satırlar için kosa giriyor ve true ise silinecek ürün no bilgisi alınıyor.
                    //false ise o satır aynen yeniden yazılıyor.
                    if (hakedis.projelNo == projelNo && hakedis.donem == donem)
                    {
                        cout << left << setw(12) << "ProjelNo" << setw(12) << "Donemi" << setw(35) << "Urun(ler)" << setw(20) << "Adet(ler)" << setw(20) << "Fiyat(lar)" << endl;
                        cout << "-----" << endl;
                        cout << left << setw(12) << hakedis.projelNo << setw(12) << hakedis.donem << setw(35) << hakedis.urunNo << setw(20) << hakedis.urunAdet << setw(20) << hakedis.urunToplamFiyat << "\n";
                        do
                        {
                            cout << "\nUrun No: "; cin >> urunNum;
                            //Silinmesi istenen urunun, urunler dosyasında bulunup bulunmadığı sorgulanıyor.
                            if (urunNo_Kontrol(urunNum) == 0)
                            {
                                cout << "Urun no urun listesinde mevcut değil. Lütfen yeniden deneyiniz.\n";
                            }
                            else
                            {
                                //Urun listesinde olup girilen proje ve doneme ait hakediste urunun bulunmama ihtimaline karşın bu hakediste urun no'da, girilen urun no'nun bulunup bulunmadığı
                                sorgulanıyor. Girilen urun no bu hakediste mevcut ise parcala fonksiyonu ile urunNo, urunAdet, urunToplamFiyat'ta bulunan girdiler eski adı altındaki vektörlere
                                aktarılıyor.
                                if (hakedis.urunNo.find(urunNum + "-") != string::npos)
                                {
                                    const char tire = '-';
                                    parcala(hakedis.urunNo, tire, eskiUrunler);
                                    parcala(hakedis.urunToplamFiyat, tire, eskiUrunToplamlari);
                                    parcala(hakedis.urunAdet, tire, eskiUrunAdetleri);
                                    //Bu hakediste yalnızca bir ürün bulunuyor ise urunun silinebilmesi engelleniyor. Eğer birden fazla ürün bulunuyor ise eski adı altındaki vektörlerden
                                    girilen ürün no'nun ait olduğu indis ve yine aynı indiste bulunan ürün no'ya ait miktar, toplam fiyat bilgileri hariç diğer bilgiler aynen tasınıyor.
                                    if (eskiUrunler.size() > 1)
                                    {
                                        kontrol = 1;
                                        for (int i = 0; i < eskiUrunler.size(); i++)
                                        {
                                            if (eskiUrunler[i] != urunNum)
                                            {
                                                yeniUrunler.push_back(eskiUrunler[i]);
                                                yeniUrunToplamlari.push_back(eskiUrunToplamlari[i]);
                                                yeniUrunAdetleri.push_back(eskiUrunAdetleri[i]);
                                            }
                                        }
                                        hakedis.urunNo = "";
                                        hakedis.urunToplamFiyat = "";
                                        hakedis.urunAdet = "";
                                        for (int j = 0; j < yeniUrunler.size(); j++)
                                        {
                                            hakedis.urunNo += yeniUrunler[j] + "-";
                                            hakedis.urunToplamFiyat += yeniUrunToplamlari[j] + "-";
                                            hakedis.urunAdet += yeniUrunAdetleri[j] + "-";
                                        }
                                        cout << "Urun silindi.\n";
                                    }
                                    else
                                    {
                                        cout << "Yalnızca 1 ürün bulunduğundan silme işlemi yapılamaz.\n";
                                    }
                                }
                            }
                        }
                    }
                }
            }
        }
    }
}
```

```

        eskiurunler.clear();
        eskiurunToplamlari.clear();
        eskiurunAdetleri.clear();
        yeniurunler.clear();
        yeniurunToplamlari.clear();
        yeniurunAdetleri.clear();
    }
    else
    {
        cout << "Bu proje no ve doneme ait boyle bir urun kayitli degil.\n";
    }
    cout << "\nBu proje no ve doneme ait baska bir urun silecek misiniz?(e/h)\n"; cin >> cevap;
    while ((cevap != 'h'));
    YeniDosya << hakedis.projelNo << "," << hakedis.donem << "," << hakedis.urunNo << "," << hakedis.urunAdet << "," << hakedis.urunToplamFiyat << "\n";
    else
    {
        YeniDosya << hakedis.projelNo << "," << hakedis.donem << "," << hakedis.urunNo << "," << hakedis.urunAdet << "," << hakedis.urunToplamFiyat << "\n";
    }
    else
    {
        cout << "Proje no'ya ait boyle bir hakedis donemi mevcut degil.\n";
    }
    else
    {
        cout << "Proje no hakedis listesinde olmadigindan islem yapilabilecek hakedis bulunamadi.\n";
    }
}
Dosya.close();
YeniDosya.close();
//Kayit girildiyse eski dosya siliniyor ve gecici olarak olusturulan dosya eski dosyasinin adini aliyor.
if (kontrol == 1)
{
    remove("hakedisler.txt");
    rename("hakedisleryeni.txt", "hakedisler.txt");
    cout << "Urun(ler) silindi.\n";
}
else
{
    remove("hakedisleryeni.txt");
}
}

```

Proje no'nun projelerin bulunduğu dosyada bulunmaması durumunda konsol çıktısı şöyledir;

```

Proje no giriniz.
2000
1. Proje no projeler listesinde mevcut degil.

```

Proje no'nun projeler dosyasında bulunduğu fakat hakedişler dosyasında bulunmadığı durumda konsol çıktısı şöyledir;

```

Proje no giriniz.
2004
2. Proje no hakedis listesinde olmadigindan islem yapilabilecek hakedis bulunamadi.

```

Proje no'nun projeler dosyasında ve hakedişler dosyasında bulunduğu fakat bu proje numarasına ait girilen donemin bulunmadığı durumda konsol çıktısı şöyledir;

```

Proje no giriniz.
2005
Hakedis donemini giriniz.
10
3. Proje no'ya ait boyle bir hakedis donemi mevcut degil.

```

Proje no'nun projeler, hakedişler dosyasında bulunduğu ve proje numarasına ait girilen donemin bulunduğu fakat girilen ürün numarasının ürünler dosyasında bulunmadığı durumda konsol çıktısı şöyledir;

```

Proje no giriniz.
2005
Hakedis donemini giriniz.
9
ProjeNo      Donemi      Urun(ler)      Adet(ler)      Fiyat(lar)
-----
2005         9           1004-1010-1007-  5-2-8-         70-74-200-

Urun No:1000
Urun no urun listesinde mevcut degil. Lutfen yeniden deneyiniz.
4. Bu proje no ve doneme ait baska bir urun silecek misiniz?(e/h)

```

Proje no'nun projeler, hakedişler dosyasında bulunduğu, proje numarasına ait girilen donemin olduğu ve girilen ürün numarasının ürünler dosyasında bulunduğu fakat girilen ürün numarasının bu dönem ve proje numarasına ait hakedişte bulunup bulunmaması durumu kontrolünün kodu şöyledir;

```
if (hakedis.urunNo.find(urunNum + "-") != string::npos)
```

Bulunmaması durumunda ise konsol çıktısı ise şöyledir;

```
Proje no giriniz.
2005
Hakedis donemini giriniz.
9
ProjeNo      Donemi      Urun(ler)      Adet(ler)      Fiyat(lar)
-----
2005         9         1004-1010-1007-  5-2-8-        70-74-200-

Urun No:1005
Bu proje no ve doneme ait boyle bir urun kayitli degil.
Bu proje no ve doneme ait baska bir urun silecek misiniz?(e/h)
```

Tüm şartların sağlandığı fakat hakedişte tek ürün bulunduğu veya kaldığı durumda konsol çıktısı şöyledir;

```
Proje no giriniz.
2002
Hakedis donemini giriniz.
9
ProjeNo      Donemi      Urun(ler)      Adet(ler)      Fiyat(lar)
-----
2002         9         1005-          19-           342-

Urun No:1005
Yalnizca 1 urun bulundugundan silme islemi yapamazsiniz.
```

Tüm şartların hakedişten silinebilir hale geldiği durumda konsol çıktısı şöyledir;

```
10
Proje no giriniz.
2009
Hakedis donemini giriniz.
10
ProjeNo      Donemi      Urun(ler)      Adet(ler)      Fiyat(lar)
-----
2009         10        1001-1002-1005-1007-  10-7-9-16-    40-49-162-400-

Urun No:1005
Urun silindi.

Bu proje no ve doneme ait baska bir urun silecek misiniz?(e/h)
e

Urun No:1001
Urun silindi.

Bu proje no ve doneme ait baska bir urun silecek misiniz?(e/h)
```

Hakediş sistemlerinde çok kez kullanımı mevcut fonksiyonların kodlaması sayfa sonunda mevcut olacaktır.

1. ProjeNo_Kontrol fonk.
2. HakedisProjeNo_Kontrol fonk.
3. HakedisteProjeNoveDonemVarMi fonk.
4. urunNo_Kontrol fonk

11) Hakediş Ürün Güncelleme

HakedisUrunGuncelle fonksiyonu ile işlem sağlanmaktadır. Fonksiyonda gerekli kontroller yapılarak yorum satırında yazmakta olan işlemler yapılmaktadır.

```
//hakedisler.txt dosyasında hakedise eklenmiş olan ürünlerin yeniden girilebilmesini sağlayan fonksiyon.
void HakedisUrunGuncelle(string projeNum)
{
    string urunNum, donem, satir;
    int urunAdet, birimFiyat = 0;
    fstream Dosya;
    fstream YeniDosya;
    vector<string> urunNolar;
    vector<int> urunAdetler;
    vector<int> urunToplamFiyatlar;
    char cevap = 'e';
    Dosya.open("hakedisler.txt", ios::in | ios::app);
    YeniDosya.open("hakedisler yeni.txt", ios::out | ios::app);
    bool kontrol = 0;
    //Projeler dosyasında girilen proje no'nun bulunup bulunmadığı sorgulanıyor.
    if (ProjeNo_Kontrol(projeNum) == 0)
    {
        cout << "Proje no projeler listesinde mevcut degil.\n";
    }
    else
    {
        //Hakedisler dosyasında girilen proje no'nun olup olmadığ kontrolu yapılıyor.
        if (HakedisProjeNo_Kontrol(projeNum) == 1)
        {
            cout << "Hakedis donemini giriniz.\n"; cin >> donem;
            //Proje no hakedisler dosyasında da mevcut ise bu proje no'ya ait girilen boyle bir donem mevcut mu onun kontrolu yapılıyor. Mevcutsa bu hakedis için yeniden urun no ve miktar bilgileri aliniyor.
            if (HakedisProjeNoVeDonemVarMi(projeNum, donem) == 1)
            {
                while (getline(Dosya, satir))
                {
                    stringstream ss(satir);
                    getline(ss, hakedis.projeNo, ',');
                    getline(ss, hakedis.donem, ',');
                    getline(ss, hakedis.urunNo, ',');
                    getline(ss, hakedis.urunAdet, ',');
                    getline(ss, hakedis.urunToplamFiyat, ',');
                    //Hakedisler dosyasından tüm satırlar okunarak proje no ve donemin girildire eşit olduğu satırlar için kosula giriyor ve true ise yeniden urun no ve miktar bilgileri aliniyor
                    //ve urunler dosyasından birin miktar cekilerek verilerek tekrar hakedis eklemede olduğu gibi yaziliyor. false ise o satir aynen yeniden yaziliyor.*/
                    if (hakedis.projeNo == projeNum && hakedis.donem == donem)
                    {
                        do
                        {
                            cout << "\nUrun No: "; cin >> urunNum;
                            if (UrunNo_Kontrol(urunNum) == 0)
                            {
                                cout << "Urun no urun listesinde mevcut degil. Lutfen yeniden deneyiniz.\n";
                            }
                            else
                            {
                                urunNolar.push_back(urunNum);
                                cout << "\nUrun adet : "; cin >> urunAdet;
                                urunAdetler.push_back(urunAdet);
                                fstream Dosya;
                                Dosya.open("insaatmalzemeleri.txt", ios::in | ios::app);
                                while (getline(Dosya, satir))
                                {
                                    stringstream ss(satir);
                                    getline(ss, urun.urunNo, ',');
                                    getline(ss, urun.urunAdi, ',');
                                    getline(ss, urun.urunBirim, ',');
                                    getline(ss, urun.urunBirimFiyat, ',');
                                    getline(ss, urun.urunMarka, ',');
                                    if (urun.urunNo == urunNum)
                                    {
                                        birimFiyat = stoi(urun.urunBirimFiyat);
                                    }
                                }
                                Dosya.close();
                                urunToplamFiyatlar.push_back(birimFiyat * urunAdet);
                            }
                        } while ((cevap != 'h'));
                        cout << "\nBaska urun kayit yapacak misiniz?(e/h)\n"; cin >> cevap;
                    }
                    //Öncelikle urun listesinde olmayan bir urun girilirse yeniden hakedise baska urun kaydi yapilip yapilmadigi sorulduğundan dolayi boyle bir durumda
                    //h basip direkt cikilmasi ihtimaline karsin vectore veri girildiginin kontrolu yapiliyor. Girildi ise dosyaya yaziliyor.*/
                    if (urunNolar.size() > 0)
                    {
                        kontrol = 1;
                        hakedis.urunNo = "";
                        hakedis.urunAdet = "";
                        hakedis.urunToplamFiyat = "";
                        for (int i = 0; i < urunNolar.size(); i++)
                        {
                            hakedis.urunNo += urunNolar[i] + "-";
                            hakedis.urunAdet += to_string(urunAdetler[i]) + "-";
                            hakedis.urunToplamFiyat += to_string(urunToplamFiyatlar[i]) + "-";
                        }
                        YeniDosya << hakedis.projeNo << "," << hakedis.donem << "," << hakedis.urunNo << "," << hakedis.urunAdet << "," << hakedis.urunToplamFiyat << "\n";
                    }
                }
            }
            else
            {
                YeniDosya << hakedis.projeNo << "," << hakedis.donem << "," << hakedis.urunNo << "," << hakedis.urunAdet << "," << hakedis.urunToplamFiyat << "\n";
            }
        }
    }
}
```

```

    else
    {
        cout << "Proje no'ya ait boyle bir hakedis donemi mevcut degil.\n";
    }
}
else
{
    cout << "Proje no proje listesinde olup, hakedis listesinde olmadigindan silinecek hakedis bulunamadi.\n";
}
}
Dosya.close();
YeniDosya.close();
//Kayit girildiyse eski dosya siliniyor ve gecici olarak olusturulan dosya eski dosyasinin adini aliyor.
if (kontrol == 1)
{
    remove("hakedisler.txt");
    rename("hakedisler yeni.txt", "hakedisler.txt");
    cout << "Urun(ler) guncellendi.\n";
}
else
{
    remove("hakedisler yeni.txt");
}
}
urunNolar.clear();
urunAdetler.clear();
urunToplamFiyatlar.clear();
}

```

Proje no'nun projelerin bulunduğu dosyada bulunmaması durumunda konsol çıktısı şöyledir;

```

Proje no giriniz.
2000
1. Proje no projeler listesinde mevcut degil.

```

Proje no'nun projeler dosyasında bulunduğu fakat hakedişler dosyasında bulunmadığı durumda konsol çıktısı şöyledir;

```

Proje no giriniz.
2004
2. Proje no hakedis listesinde olmadigindan islem yapilabilecek hakedis bulunamadi.

```

Proje no'nun projeler dosyasında ve hakedişler dosyasında bulunduğu fakat bu proje numarasına ait girilen donemin bulunmadığı durumda konsol çıktısı şöyledir;

```

Proje no giriniz.
2005
Hakedis donemini giriniz.
10
3. Proje no'ya ait boyle bir hakedis donemi mevcut degil.

```

Proje no'nun projeler, hakedişler dosyasında bulunduğu ve proje numarasına ait girilen donemin bulunduğu fakat girilen ürün numarasının ürünler dosyasında bulunmadığı durumda konsol çıktısı şöyledir;

```

11
Proje no giriniz.
2007
Hakedis donemini giriniz.
3

Urun No:1000
4. Urun no urun listesinde mevcut degil. Lutfen yeniden deneyiniz.

```

Tüm şartların hakedişten ürünlerin güncellenebilir hale geldiği durumda konsol çıktısı şöyledir;

```
Proje no giriniz.
2007
Hakedis donemini giriniz.
3

Urun No:1007

Urun adet :19

Baska urun kayit yapacak misiniz?(e/h)
```

1. ProjeNo_Kontrol fonk.
2. HakedisProjeNo_Kontrol fonk.
3. HakedisteProjeNoveDonemVarMi fonk.
4. urunNo_Kontrol fonk

12) Hakediş Kullanılan Ürün Arama

HakedisKullanilanUrunListele fonksiyonu ile işlem sağlanmaktadır. Fonksiyonda gerekli kontroller yapılarak yorum satırında yazmakta olan işlemler yapılmaktadır.

```
//hakedisler.txt dosyasında girilen proje no'ya ait tum donemlerin cekilerek kullanilan urunlerle beraber ekrana yazilmasini saglayan fonksiyon.
void HakediskullanilanUrunListele(string projenUm)
{
    ifstream Dosya;
    string satir;
    Dosya.open("hakedisler.txt", ios::in | ios::app);
    //Projeler dosyasında girilen proje no'nun bulunup bulunmadigi sorgulanıyor.
    if (ProjeNo_Kontrol(projenUm) == 1)
    {
        //Hakedisler dosyasında girilen proje no'nun olup olmadigi kontrolu yapiliyor. Ardından tum satirlar taranarak girilen proje no'ya ait hakedislerde kullanilan urunler ekrana yazdiriliyor.
        if (HakedisProjeNo_Kontrol(projenUm) == 1)
        {
            cout << left << setw(12) << "ProjeNo" << setw(12) << "Donemi" << setw(35) << "Urun(lar)" << setw(20) << "Adet(lar)" << setw(20) << "Fiyat(lar)" << endl;
            cout << "-----" << endl;
            while (getline(Dosya, satir))
            {
                stringstream ss(satir);
                getline(ss, hakedis.projeNo, ',');
                getline(ss, hakedis.donem, ',');
                getline(ss, hakedis.urunNo, ',');
                getline(ss, hakedis.urunAdet, ',');
                getline(ss, hakedis.urunToplamFiyat, ',');
                if (projenUm == hakedis.projeNo)
                {
                    cout << left << setw(12) << hakedis.projeNo << setw(12) << hakedis.donem << setw(35) << hakedis.urunNo << setw(20) << hakedis.urunAdet << setw(20) << hakedis.urunToplamFiyat << "\n";
                }
            }
        }
        else
        {
            cout << "Hakedis listesinde bu proje no'ya ait hakedis mevcut degil.\n";
        }
    }
    else
    {
        cout << "Proje listesinde bu proje no mevcut degil.\n";
    }
    Dosya.close();
}
```

Proje no'nun projelerin bulunduğu dosyada bulunmaması durumunda konsol çıktısı şöyledir;

```
Kullanilan urunleri gormek istediginiz proje no giriniz. (Proje no'ya ait eger birden fazla donem var ise tumu gorunt
ulenecektir.)
2000
1. Proje listesinde bu proje no mevcut degil.
```

Proje no'nun projeler dosyasında bulunduğu fakat hakedişler dosyasında bulunmadığı durumda konsol çıktısı şöyledir;

```
Kullanilan urunleri gormek istediginiz proje no giriniz. (Proje no'ya ait eger birden fazla donem var ise tumu gorunt
ulenecektir.)
2010
2. Hakedis listesinde bu proje no'ya ait hakedis mevcut degil.
```


Tüm şartların sağlandığı durumda bu projede hangi ürünlerin kullanıldığının görüntüsü konsol çıktısı şöyledir;

```
Kullanılan ürünleri görmek istediğiniz proje no giriniz. (Proje no'ya ait eger birden fazla donem var ise tumu gorunt
ulenecektir.)
2008
```

ProjeNo	Donemi	Urun(ler)	Adet(ler)	Fiyat(lar)
2008	8	1006-1003-	11-2-	231-22-
2008	5	1007-1008-	17-6-	425-168-
2008	12	1010-1009-1005-	10-5-13-	370-160-234-

1. ProjeNo_Kontrol fonk.

2. HakedisProjeNo_Kontrol fonk.

13) Hakediş Ürünlerde Arama

HakedisUrunlerdeAra fonksiyonu ile işlem sağlanmaktadır. Fonksiyonda gerekli kontroller yapılarak yorum satırında yazmakta olan işlemler yapılmaktadır.

```
//hakedisler.txt dosyasında girilen urun no'nun bulunduğu tum hakedislerin ekrana yazdırılmasını sağlayan fonksiyon.
void HakedisUrunlerdeAra(string urunNum)
{
    fstream Dosya;
    string satir;
    Dosya.open("hakedisler.txt", ios::in | ios::app);
    //Urunler dosyasında girilen urun no'nun bulunup bulunmadığı sorgulanıyor.
    if (urunNo_Kontrol(urunNum) == 1)
    {
        /*Hakedisler dosyasında girilen urun no'nun herhangi bir hakediste bulunup bulunmadığı sorgulanıyor. Bulunuyor ise tum satirlar taraniyor,
        girilen urun no'nun bulunduğu satirlar ekrana yazdırılıyor.*/
        if (HakedisUrunNo_Kontrol(urunNum) == 1)
        {
            cout << left << setw(12) << "ProjeNo" << setw(12) << "Donemi" << setw(35) << "Urun(ler)" << setw(20) << "Adet(ler)" << setw(20) << "Fiyat(lar)" << endl;
            cout << "-----" << endl;
            while (getline(Dosya, satir))
            {
                stringstream ss(satir);
                getline(ss, hakedis.projNo, ',');
                getline(ss, hakedis.donem, ',');
                getline(ss, hakedis.urunNo, ',');
                getline(ss, hakedis.urunAdet, ',');
                getline(ss, hakedis.urunToplamFiyat, ',');
                if (hakedis.urunNo.find(urunNum + "-") != string::npos)
                {
                    cout << left << setw(12) << hakedis.projNo << setw(12) << hakedis.donem << setw(35) << hakedis.urunNo << setw(20) << hakedis.urunAdet << setw(20) << hakedis.urunToplamFiyat << "\n";
                }
            }
        }
        else
        {
            cout << "Hakedis listesinde bu urun no'nun bulunduğu hakedis mevcut degil.\n";
        }
    }
    else
    {
        cout << "Urunler listesinde bu urun no mevcut degil.\n";
    }
    Dosya.close();
}
```

Ürün no'nun ürünlerin bulunduğu dosyada bulunmaması durumunda konsol çıktısı şöyledir;

```
Hakedislerde bulunan aramak istediğiniz urun no giriniz.
1000
1. Urunler listesinde bu urun no mevcut degil.
```

Ürün no'nun projeler dosyasında bulunduğu fakat hakedişler dosyasında böyle bir ürünün bulunduğu hakediş bulunmadığı durumda konsol çıktısı şöyledir;

```
Hakedislerde bulunan aramak istediğiniz urun no giriniz.
1006
2. Hakedis listesinde bu urun no'nun bulunduğu hakedis mevcut degil.
```

Tüm şartların sağlandığı durumda bu ürünün hangi hakedişlerde kullanıldığının görüntüsü konsol çıktısı şöyledir;

```
Hakedislerde bulunan aramak istediginiz urun no giriniz.
```

1010				
ProjeNo	Donemi	Urun(ler)	Adet(ler)	Fiyat(lar)
2001	1	1001-1010-	14-7-	56-259-
2005	9	1004-1010-1007-	5-2-8-	70-74-200-
2008	8	1004-1010-	14-6-	196-222-
2008	12	1010-1009-1005-	10-5-13-	370-160-234-

1. urunNo_Kontrol fonk.

2. HakedisUrunNo_Kontrol fonk.

14) Proje numarasına göre hakedişlerin total tutarını görüntüleme

HakedisProjeNodanTotalTutarListele fonksiyonu ile işlem sağlanmaktadır. Fonksiyonda gerekli kontroller yapılarak yorum satırında yazmakta olan işlemler yapılmaktadır.

```
//hakedisler.txt dosyasında girilen proje no'nun bulunduğu tüm hakedislerin ve bu hakedislerin toplam tutarlarının ekrana yazdırılmasını sağlayan fonksiyon.
void HakedisProjeNodanTotalTutarListele(string projeNum)
{
    vector<string> projeToplamFiyat;
    string satir;
    int genelToplam = 0;
    ifstream Dosya;
    Dosya.open("hakedisler.txt", ios::in | ios::app);
    //Projeler dosyasında girilen proje no'nun bulunup bulunmadığı sorgulanıyor.
    if (ProjeNo_Kontrol(projeNum) == 1)
    {
        //Hakedisler dosyasında girilen proje no'nun olup olmadığı kontrolü yapılıyor. Var ise tüm satırlar taranıyor.
        if (HakedisProjeNo_Kontrol(projeNum) == 1)
        {
            cout << left << setw(12) << "ProjeNo" << setw(12) << "Donemi" << setw(35) << "Urun(ler)" << setw(20) << "Adet(ler)" << setw(20) << "Fiyat(lar)" << endl;
            cout << "-----" << endl;
            while (getline(Dosya, satir))
            {
                stringstream ss(satir);
                getline(ss, hakedis.projeNo, ',');
                getline(ss, hakedis.donem, ',');
                getline(ss, hakedis.urunNo, ',');
                getline(ss, hakedis.urunAdet, ',');
                getline(ss, hakedis.urunToplamFiyat, ',');
                /*Satırda proje no bulunun ise birden fazla veri girişi ihtimaline karşı "-" ile ayrılmış olan toplam fiyat kısmı '-'lerden parçala fonksiyonu ile parçalanarak
                projeToplamFiyat vektörüne kaydediliyor. Ardından genelToplam'a vektördeki girdiler ekleniyor. Son olarak hakedisler ve genel toplam ekrana yazdırılıyor.*/
                if (hakedis.projeNo == projeNum)
                {
                    cout << left << setw(12) << hakedis.projeNo << setw(12) << hakedis.donem << setw(35) << hakedis.urunNo << setw(20) << hakedis.urunAdet << setw(20) << hakedis.urunToplamFiyat << "\n";
                    const char tire = '-';
                    parçala(hakedis.urunToplamFiyat, tire, projeToplamFiyat);
                    for (int i = 0; i < projeToplamFiyat.size(); i++)
                    {
                        genelToplam += stoi(projeToplamFiyat[i]);
                    }
                    projeToplamFiyat.clear();
                }
            }
            cout << "\nGirmiş olduğunuz proje no'ya ait tüm dönem hakedislerin toplam tutarı:\t" << genelToplam << "\n";
        }
        else
        {
            cout << "Hakedis listesinde bu proje no'nun bulunduğu hakedis mevcut değil.\n";
        }
    }
    else
    {
        cout << "Projeler listesinde bu proje no mevcut değil.\n";
    }
    Dosya.close();
}
```

Proje no'nun projelerin bulunduğu dosyada bulunmaması durumunda konsol çıktısı şöyledir;

```
Total tutarı gormek istediginiz proje no giriniz.
2000
1. Projeler listesinde bu proje no mevcut degil.
```

Proje no'nun projeler dosyasında bulunduğu fakat hakedişler dosyasında bulunmadığı durumda konsol çıktısı şöyledir;

```
Total tutarı gormek istediginiz proje no giriniz.
2010
2. Hakedis listesinde bu proje no'nun bulunduğu hakedis mevcut degil.
```

Tüm şartların sağlandığı durumda bu projeye ait tüm dönem hakedişlerin listelenmesi ve toplam tutarının yazıldığı konsol çıktısı şöyledir;

```
Total tutarı görmek istediğiniz proje no giriniz.
2008
ProjeNo      Donemi      Urun(ler)      Adet(ler)      Fiyat(lar)
-----
2008         8          1004-1010-      14-6-          196-222-
2008         5          1007-1008-      17-6-          425-168-
2008        12          1010-1009-1005- 10-5-13-        370-160-234-

Girmis oldugunuz proje no'ya ait tum donem hakedislerin toplam tutari: 1775
```

1. ProjeNo_Kontrol fonk.

2. HakedisProjeNo_Kontrol fonk.

15) Döneme göre hakediş total tutarı görüntüleme

HakedisDonemdenTotalTutarListele fonksiyonu ile işlem sağlanmaktadır. Fonksiyonda gerekli kontroller yapılarak yorum satırında yazmakta olan işlemler yapılmaktadır.

```
//hakedisler.txt dosyasında girilen donemin bulunduğu tüm hakedislerin ve bu hakedislerin toplam tutarlarının ekrana yazdırılmasını sağlayan fonksiyon.
void HakedisDonemdenTotalTutarListele(string hakedisDonem)
{
    vector<string> donemToplamFiyat;
    string satir;
    int genelToplam = 0;
    fstream Dosya;
    Dosya.open("hakedisler.txt", ios::in | ios::app);
    //Hakedisler dosyasında girilen donemin olup olmadığı kontrolü yapılıyor. Var ise tüm satırlar taranıyor.
    if (HakedisDonem_Kontrol(hakedisDonem) == 1)
    {
        cout << left << setw(12) << "ProjeNo" << setw(12) << "Donemi" << setw(35) << "Urun(ler)" << setw(20) << "Adet(ler)" << setw(20) << "Fiyat(lar)" << endl;
        cout << "-----" << endl;
        while (getline(Dosya, satir))
        {
            stringstream ss(satir);
            getline(ss, hakedis.projeNo, ',');
            getline(ss, hakedis.donem, ',');
            getline(ss, hakedis.urunNo, ',');
            getline(ss, hakedis.urunAdet, ',');
            getline(ss, hakedis.urunToplamFiyat, ',');
            /*Satırda dönem bulunur ise birden fazla veri girişi ihtimaline karşı "-" ile ayrılmış olan toplam fiyat kısmı '-'lerden parçala fonksiyonu ile parçalanarak
            donemToplamFiyat vektörüne kaydediliyor. Ardından genelToplam'a vektördeki girdiler ekleniyor. Son olarak hakedisler ve genel toplam ekrana yazdırılıyor.*/
            if (hakedis.donem == hakedisDonem)
            {
                cout << left << setw(12) << hakedis.projeNo << setw(12) << hakedis.donem << setw(35) << hakedis.urunNo << setw(20) << hakedis.urunAdet << setw(20) << hakedis.urunToplamFiyat << "\n";
                const char tire = '-';
                parcala(hakedis.urunToplamFiyat, tire, donemToplamFiyat);
                for (int i = 0; i < donemToplamFiyat.size(); i++)
                {
                    genelToplam += stoi(donemToplamFiyat[i]);
                }
                donemToplamFiyat.clear();
            }
        }
        cout << "\ngirmis oldugunuz doneme ait hakedislerin toplam tutari:\t" << genelToplam << "\n";
    }
    else
    {
        cout << "Hakedis listesinde girmis oldugunuz donemin bulunduğu hakedis mevcut degil.\n";
    }
    donemToplamFiyat.clear();
    Dosya.close();
}
```

Girilen dönemin hakedişler dosyasında bulunup bulunmadığının kontrolünü yapan HakedisDonem_Kontrol fonksiyonun kodları şöyledir;

```
//hakedisler.txt dosyasında tüm satırları okuyarak girilen donemin dosyada olup olmadığının kontrolünü sağlayan fonksiyon. Var ise kontrol 1, yok ise 0 döndürüyor.
bool HakedisDonem_Kontrol(string hakedisDonem)
{
    bool kontrol = 0; //hakedisProjeNum kayıtlı değil ise 0 alıyor.
    string satir;
    fstream Dosya;
    Dosya.open("hakedisler.txt", ios::in | ios::app);
    while (getline(Dosya, satir))
    {
        stringstream ss(satir);
        getline(ss, hakedis.projeNo, ',');
        getline(ss, hakedis.donem, ',');
        getline(ss, hakedis.urunNo, ',');
        getline(ss, hakedis.urunAdet, ',');
        getline(ss, hakedis.urunToplamFiyat, ',');
        if (hakedis.donem == hakedisDonem)
        {
            kontrol = 1;
        }
    }
    return kontrol;
    Dosya.close();
}
```

Konsol çıktısı şöyledir;

```
Total tutarı görmek istediginiz donemi giriniz.
4
Hakedis listesinde girmis oldugunuz donemin bulundugu hakedis mevcut degil.
```

Şartın sağlandığı durumda bu döneme ait tüm proje hakedişlerin listelenmesi ve toplam tutarının yazıldığı konsol çıktısı şöyledir;

```
Total tutarı görmek istediginiz donemi giriniz.
9
ProjeNo      Donemi      Urun(ler)      Adet(ler)      Fiyat(lar)
-----
2002         9           1005-          19-           342-
2005         9           1004-1010-1007- 5-2-8-        70-74-200-
Girmis oldugunuz doneme ait hakedislerin toplam tutari: 686
```

ORTAK KULLANILAN FONKSİYONLAR

```
//insaatzemeleri.txt dosyasında tüm satırları okuyarak girilen urun no'nun dosyada olup olmadığının kontrolünü sağlayan fonksiyon. Var ise kontrol 1, yok ise 0 donduruyor.
bool urunNo_Kontrol(string urunNum)
{
    bool kontrol = 0;
    fstream Dosya;
    string satir;
    Dosya.open("insaatzemeleri.txt", ios::in | ios::app);
    while (getline(Dosya, satir))
    {
        stringstream ss(satir);
        getline(ss, urun.urunNo, ',');
        getline(ss, urun.urunAdi, ',');
        getline(ss, urun.urunBirim, ',');
        getline(ss, urun.urunBirimFiyat, ',');
        getline(ss, urun.urunMarka, ',');
        if (urun.urunNo == urunNum)
        {
            kontrol = 1;
        }
    }
    return kontrol;
    Dosya.close();
}
```

```
//projeler.txt dosyasında tüm satırları okuyarak girilen proje no'nun dosyada olup olmadığının kontrolünü sağlayan fonksiyon. Var ise kontrol 1, yok ise 0 donduruyor.
bool ProjeNo_Kontrol(string projeNum)
{
    bool kontrol = 0;
    string satir;
    fstream Dosya;
    Dosya.open("projeler.txt", ios::in | ios::app);
    while (getline(Dosya, satir))
    {
        stringstream ss(satir);
        getline(ss, proje.projeNo, ',');
        getline(ss, proje.projeAdi, ',');
        getline(ss, proje.projeFirma, ',');
        getline(ss, proje.projeSorumlu, ',');
        getline(ss, proje.projeKontrolor, ',');
        if (proje.projeNo == projeNum)
        {
            kontrol = 1;
        }
    }
    return kontrol;
    Dosya.close();
}
```

G191210303 BARIŞ YILMAZ 1/A PROGRAMLAMAYA GİRİŞ DERSİ YAZ DONEMI PROJE RAPORU

```
//hakedisler.txt dosyasında tum satirlari okuyarak girilen urun no'nun dosyada olup olmadiginin kontrolunu saglayan fonksiyon. Var ise kontrol 1, yok ise 0 donduruyor.
bool HakedisUrunNo_Kontrol(string hakedisUrunNum)
{
    bool kontrol = 0;
    string satir;
    fstream Dosya;
    Dosya.open("hakedisler.txt", ios::in | ios::app);
    while (getline(Dosya, satir))
    {
        stringstream ss(satir);
        getline(ss, hakedis.projeNo, ',');
        getline(ss, hakedis.donem, ',');
        getline(ss, hakedis.urunNo, ',');
        getline(ss, hakedis.urunAdet, ',');
        getline(ss, hakedis.urunToplamFiyat, ',');
        if (hakedis.urunNo.find(hakedisUrunNum + "-") != string::npos)
        {
            kontrol = 1;
        }
    }
    return kontrol;
    Dosya.close();
}
```

```
//hakedisler.txt dosyasında tum satirlari okuyarak girilen proje no'nun dosyada olup olmadiginin kontrolunu saglayan fonksiyon. Var ise kontrol 1, yok ise 0 donduruyor.
bool HakedisProjeNo_Kontrol(string hakedisProjeNum)
{
    bool kontrol = 0;
    string satir;
    fstream Dosya;
    Dosya.open("hakedisler.txt", ios::in | ios::app);
    while (getline(Dosya, satir))
    {
        stringstream ss(satir);
        getline(ss, hakedis.projeNo, ',');
        getline(ss, hakedis.donem, ',');
        getline(ss, hakedis.urunNo, ',');
        getline(ss, hakedis.urunAdet, ',');
        getline(ss, hakedis.urunToplamFiyat, ',');
        if (hakedis.projeNo == hakedisProjeNum)
        {
            kontrol = 1;
        }
    }
    return kontrol;
    Dosya.close();
}
```

```
//hakedisler.txt dosyasında tum satirlari okuyarak girilen donemin dosyada olup olmadiginin kontrolunu saglayan fonksiyon. Var ise kontrol 1, yok ise 0 donduruyor.
bool HakedisDonem_Kontrol(string hakedisDonem)
{
    bool kontrol = 0; //hakedisProjeNum kayıtlı değil ise 0 alıyor.
    string satir;
    fstream Dosya;
    Dosya.open("hakedisler.txt", ios::in | ios::app);
    while (getline(Dosya, satir))
    {
        stringstream ss(satir);
        getline(ss, hakedis.projeNo, ',');
        getline(ss, hakedis.donem, ',');
        getline(ss, hakedis.urunNo, ',');
        getline(ss, hakedis.urunAdet, ',');
        getline(ss, hakedis.urunToplamFiyat, ',');
        if (hakedis.donem == hakedisDonem)
        {
            kontrol = 1;
        }
    }
    return kontrol;
    Dosya.close();
}
```

```

/*hakedisler.txt dosyasında tum satırları okuyarak girilen proje no ve donemin dosyada olup olmadığının kontrolunu saglayan fonksiyon. Fonksiyonun
amaci bazı hakedis islemlerinde proje no var ise o proje no'ya girilen donemde de veri var mıdır. Var ise kontrol 1, yok ise 0 donduruyor.*/
bool HakedisteProjeNoveDonemVarMi(string projeNum, string donem)
{
    bool kontrol = 0;
    string satir;
    fstream Dosya;
    Dosya.open("hakedisler.txt", ios::in | ios::app);
    while (getline(Dosya, satir))
    {
        stringstream ss(satir);
        getline(ss, hakedis.projeNo, ',');
        getline(ss, hakedis.donem, ',');
        getline(ss, hakedis.urunNo, ',');
        getline(ss, hakedis.urunAdet, ',');
        getline(ss, hakedis.urunToplamFiyat, ',');
        if (hakedis.projeNo == projeNum && hakedis.donem == donem)
        {
            kontrol = 1;
        }
    }
    return kontrol;
    Dosya.close();
}

```

```

/*Hakedislerde 1'den fazla urun kaydı olabilecegi durumlarda silme, guncelleme ve total hakedis hesaplama islemlerinde urun no, urun adeti, toplam fiyat ifadelerini ayirarak
islemlerin yapilabilmesini saglayan fonksiyon. Aralarında "-" olan ifadeler her - arası alınarak vektöre kaydediliyor, ardından vektörden çekilerek gerekli işlemler yapılıyor.*/
void parcala(string const& str, const char tire,
vector<string>& out)
{
    size_t basla;
    size_t bitir = 0;
    while ((basla = str.find_first_not_of(tire, bitir)) != string::npos)
    {
        bitir = str.find(tire, basla);
        out.push_back(str.substr(basla, bitir - basla));
    }
}

```