

Adım 1- classları oluşturup içine gerekli değişkenleri metotları

Ve katar sınıfının içine karakterler[max] isminde nenseler dizisi oluşturuyoruz.

Adım 2-sonrasında kurucu fonksiyon yardımı ile karakter sayısını 0 a eşitliyoruz

```
Katar()//kurucu fonksiyon olaraktan karakter saiyisin
{
    karaktersayisi = 0;
}
```

Adım 3-calssa nesne tanımlıyoruz katar sınıfı için ktr1 ve karakter sınıfı için de karakter1 olarak.

Adım 4-main fonksiyonu içinde do while döngüsü içinde 1,2,3 gibi seçenekler sunarak

Karakter ekleme ve çıkarma metotlarına yönlendiriyoruz.

```
string secim = "0";
do//menü işlemleri....
{
    SetConsoleTextAttribute(GetStdHandle(STD_OUTPUT_HANDLE), 7);
    cout << endl << "Ekleme için 1'e basınız:" << endl;
    << "Cikarmak için 2 'ye basınız: " << endl;
    << "Cikiss yapmak için 3 'e basınız:";
    cin >> secim;
    if (secim == "1" && ktr1.karaktersayisi == 50)
    {
        SetConsoleTextAttribute(GetStdHandle(STD_OUTPUT_HANDLE), 4);

        cout << "EKLEME YAPILAMAZ..." << endl;
        SetConsoleTextAttribute(GetStdHandle(STD_OUTPUT_HANDLE), 7);
    }
    else if (secim == "1")
    {
        cout << endl << "ONCEKI DURUM eleman sayisi:" << ktr1.karaktersayisi << endl;

        ktr1.yazdir();
        ktr1.karakterEkle();
        cout << endl << "SONRAKI DURUM eleman sayisi:" << ktr1.karaktersayisi << endl;
        ktr1.yazdir();
    }
    else if (secim == "2")
    {
        if (ktr1.karaktersayisi > 0)
        {
            cout << endl << "ONCEKI DURUM eleman sayisi:" << ktr1.karaktersayisi << endl;
            ktr1.yazdir();
            ktr1.karakterCikar();
            cout << endl << "SONRAKI DURUM eleman sayisi:" << ktr1.karaktersayisi << endl;
            ktr1.yazdir();
        }
        else
        {
            SetConsoleTextAttribute(GetStdHandle(STD_OUTPUT_HANDLE), 4);
            cout << endl << "CIKARMA YAPILAMAZ..." << endl;
        }
    }
} while (secim != "3");
```

Adım 5-karakter ekleme metodunun içine `karakterler[karaktersayisi].harf = (char)(rand() % 26 + 65);` ifadesini tanımlıyoruz aynı şekilde buna benzer bir renk eşitlemesi yaparakta her karakter indisinin kendisine ait bir harfi ve bir renk değerine sahip.

```
64 }
65 void karakterEkle()//kullanıcı 1 bastığı zaman karakter ekle metodunu devre
66 {
67
68
69
70
71
72 if (karaktersayisi < 50)//max karakter sayısı 50 olduğundan 50 üzeri eklenemez
73 {
74     karakterler[karaktersayisi].harf = (char)(rand() % 26 + 65); // ascii değeri
75     karakterler[karaktersayisi].renk = rand() % 6 + 9; //her harfe belirli bir renk
76
77     karaktersayisi = karaktersayisi + 1;
78
79 }
80
81
82
83 }
```

Adım 6-karakter çıkar metodunda ise dizinin son elemanı olan elemanın harfini `'\0'` yapıp son elemanı boş değere eşitliyoruz yani karakteri o diziden çıkarıyoruz.

```
void karakterCikar()
{
    if (karaktersayisi == 0)
    {
        //cout << "CIKARMA ISLEMI YAPILAMAZ...";
    }
    else
    {
        karakterler[karaktersayisi].harf = '\0';
        karakterler[karaktersayisi].renk = rand() % 6 + 9;
        karaktersayisi--;
    }
}
```

ADIM 7- İlk karakterin kutucuğunu ve son kutucuğun tasarımını diğer kutulardan ayrı bir blokta tasarlayıp ekrana çıktı veriyorum.

İlk kutu için....

```
if (karaktersayisi <= 50)
{
    if (karaktersayisi == 1)
    {
        cout << SOLUSTKOSE << DUZCIZGI << DUZCIZGI << SAGUSTKOSE << endl;
        cout << DIKEYCIZGI;
        SetConsoleTextAttribute(GetStdHandle(STD_OUTPUT_HANDLE), karakterler[0].renk);
        cout << karakterler[0].harf;
        SetConsoleTextAttribute(GetStdHandle(STD_OUTPUT_HANDLE), 7);
        cout << " " << DIKEYCIZGI << endl;
        cout << SOLALTKOSE << DUZCIZGI << DUZCIZGI << SAGALTKOSE;
    }
}
```

Son kutu için...

```
    cout << DIKEYCIZGI << endl;
    cout << SOLALTKOSE << DUZCIZGI << DUZCIZGI;

    for (int q = 0; q < karaktersayisi - 1; q++)
    {
        cout << YUKARIAYRAC << DUZCIZGI << DUZCIZGI;
    }

    cout << SAGALTKOSE;
}
```

Adım 7-yazdır metodunda ise o anki karakter sayısına denk gelen sayı miktarınca kutu for yardımı ile her kutuya uygun düşen harf yazdırılıyor...

```
cout << SAGUSTKOSE << endl;

if (karaktersayisi != 1)
{
    for (int k = 0; k < karaktersayisi; k++)
    {
        cout << DIKEYCIZGI;
        SetConsoleTextAttribute(GetStdHandle(STD_OUTPUT_HANDLE), karakterler[k].renk);

        cout << karakterler[k].harf;
        SetConsoleTextAttribute(GetStdHandle(STD_OUTPUT_HANDLE), 7);
        cout << " ";
    }
}
```

Adım 8- Renk kullanımı içinde GetStdHandle'ı kullandım ve her harfi yazdırmadan önce ki yerde renk ayarı yapıyoruz sonrasında ise harfi yazdırdıktan sonra rengi beyaza tekrar döndürüyoruz.

Örnek olarak...

```
cout << SOLUSTKOSE << DUZCIZGI << DUZCIZGI << SAGUSTKOSE << endl;
cout << DIKEYCIZGI;
SetConsoleTextAttribute(GetStdHandle(STD_OUTPUT_HANDLE), karakterler[0].renk);
cout << karakterler[0].harf;
SetConsoleTextAttribute(GetStdHandle(STD_OUTPUT_HANDLE), 7);
cout << " " << DIKEYCIZGI << endl;
cout << SOLALTKOSE << DUZCIZGI << DUZCIZGI << SAGALTKOSE;
```

```
for (if (karakterlerin[i].l == 1)
```