



FIRAT ÜNİVERSİTESİ TEKNOLOJİ FAKÜLTESİ
YAZILIM MÜHENDİSLİĞİ BÖLÜMÜ
YMH 319 YAZILIM MÜHENDİSLİĞİ
PROGRAMLAMA DİLLERİ YIL SONU PROJESİ

Lan-ForKids

Hazırlayan

Rüveyha RÜZGAR

Ders Sorumlusu

Doç Dr. Fatih ÖZKAYNAK

Tanım:

Çocuklar İçin Türkçe Programlama Dili oluşturarak 4 İşlemlili Hesap Makinesi yapma.

Amaç ve Kapsamlar:

Gereksinimleri ve amacımızı düşündüğümüz de, burda karşıya geçirilmek istenen temel düşüncemi açıklamak istiyorum.Çocukların ya da yazılıma yeni başlamış bireylerin programlama dillerini kavrayabilmesi ve karmaşık gelen öğrenme paradoksundan çıkartıp daha basite indirgeyebileceğimizi göstermekteyim.Genel de Türkçe kaynağın az olması bir sorun teşkil ederken, bu projede de bunun önüne geçebilmeyi hedefliyorum.Nasıl çalıştığına gelecek olursak;

Proje Yapısı:

Main()

Öncelikle main bloğumuza bakalım, burda Java diline ait bir yazdır satırı yok.Herhangi bir class referansı alınmıyor.Çağırdığımız fonksiyonlara baktığımız da hepsinin türkçe kelimeler olduğunu görüyoruz. Bi bakıma aslında bu dışardan bakılınca çok anlaşılır.Ne yapıyor sorusunun cevabını rahatlıkla alabiliriz.İşte burda ki tüm amacımız buydu.İstenilen iş= o işin class+ işlem adımı ;

```
7 public static void main(String[] args)
8 {
9     SistemAracları.EkranaYazdir(Matematikselİslemler.Toplama(2,1));
10
11     SistemAracları.Hafıza1= DegiskenOlustur.KesirliSayiOlustur( sayi2: 11);
12
13     SistemAracları.EkranaYazdir(SistemAracları.Hafıza1);
14
15     SistemAracları.Hafıza1Temizle();
16
17     //SistemAracları.EkranaYazdir(Kullaniciİslemleri.DegerAl());
18 }
```

DegiskenOlustur()

Main bloğumuzda da çağırdığımız DeğişkenOlustur classını çağırdığımız da anlaşılabilirliği kaybetmemek adına içerisine tanımladığımız methodlarında bağımlı olması gerekiyordu. Çocuklar için olduğunu unutmayalım. Class + method mantığında çalışırken nesneleri/referansları almamayı tercih etmiştik. Bunu da her method ve classlar da göreceğiniz üzere "static" anahtar sözcüğü ile sağlıyoruz. İşlemimizin değer tipini türkçe olarak alıp çağırabilmek için oluşturulan bu adımları örnekleriyle de yukarda gördük.

```
19      public static class DegiskenOlustur
20      {
21
22          public static int TamSayiOlustur(int sayi1)
23          {
24              float floatsayi = sayi1;
25              return sayi1;
26          }
27          public static float KesirliSayiOlustur(float sayi2)
28          {
29              return sayi2;
30          }
31      }
```

Matematikselİslemler()

İşlemleri yapacağımız class temizlik açısından önemliydi. Burda Matematikselİslemler classını çağırdığımızda yapabileceğimiz 4 işlemi kodluyoruz. Oluşturduğumuz bu işlem methodlarını overload edip 2 veya 3 elemanla girişlerde sorun yaratmaması için garantiliyoruz. Daha sonra işlemleri yazıp sonucu return ettiriyoruz.

```
32      public static class Matematikselİslemler
33      {
34          public static float toplam = 0;
35
36          public static float Toplama(float sayi1, float sayi2, float sayi3)
37          {
38              toplam = sayi1 + sayi2 + sayi3;
39
40              return toplam;
41          }
42          public static float Toplama(float sayi1, float sayi2)
43          {
44              toplam = sayi1 + sayi2;
45              return toplam;
46          }
47      }
48  }
```

SistemAraclari()

Bu classımızda da temel görev, yazdırma ve istenilirse yapılan işlemleri bir depoda (.txt/.xlsx/.csv) "geçmiş" olarak kaydedebilmek adına oluşturuldu. Küçük bir update ile mevcut hale getirilebilir, düzeninde yazıldı. Her programcının ilk öğrendiği ve yazdığı ("Hello world!") çıktısını, Java da System.out.print(); satırı ile yazabiliyorduk. Bir çocuk programcı için ise bunu hiç İngilizce kelime kullanmayarak bir method ile yazdırmak istedim. Sadece her class ve method ile yaptığımız adımları burda da yazdırmak için yapabilir! Bu methodu çağırdığında her şeyi yazdırabilsin diye her değer tipimiz için aynı isim de bir method oluşturup hatanın önüne geçtim. Artık her şey çok daha basit.

```
CalculatorForKidsLanguage.java x
84      public static class SistemAraclari
85      {
86
87          public static void EkranaYazdir()
88          {
89              System.out.println("Henüz bir şey girmediniz");
90          }
91          public static void EkranaYazdir(String S)
92          {
93              System.out.println(S);
94          }
95          public static void EkranaYazdir(int S)
96          {
97              System.out.println(S);
98          }
99          public static void EkranaYazdir(float S)
100         {
101             System.out.println(S);
102         }
103         static public float Hafiza1 = 0;
104         static public float Hafiza2 = 0;
105
106         public static float HafizadaTut(float tut)
107         {
108             Hafiza1 = tut;
109             return tut;
110         }
111         public static float HafizadaTut()
112         {
113             return 0;
114         }
115
116         public static float HafizadaTut2(float tut)
117         {
118             Hafiza2 = tut;
119             return tut;
120         }
121         public static float HafizadaTut2()
122         {
123             return 0;
124         }
125
126         public static void Hafiza1Temizle()
127         {
```

