



Görsel Programlama!

- Görsel elementler kullanılarak program oluşturmak
- Suf görsel programlama dilleri

Suf olmayan görsel programlama
- Proje büyütme kolaylığıını sağlar.
(Softağa şımarak)
 - Görsel olarak yazılır.
 - Skratch

- Visual Basic
 - Visual C#
 - Visual J#
 - Metin olarak yazılır.

!!! Visual Studio 2019 download

~~Sonuç~~ / Kapalı türde geliştirme
tekniklerne yararlanan
dogru adımlar
olabiliyor mu?

C# is an object-oriented programming language.

• .NET Framework Class Library

Database	Debugging
Web APPS	Multi-threading
Graphics	File processing
input\output	(GUI)
Networking	Data Structure
Mobile	
String processing	

GUI (Graphical User Interface)

You can write programs that respond to user initiated events such as mouse clicks, key strokes, touches, finger swipes, gestures that are widely used on mobile phone or tablets.

Visual Studio enables you to use C# as a visual programming language. Visual Studio will write much of the GUI code for you.

using System;

It is a using directive that tells the compiler where to look for a class that's used in this app.

System is a namespace and Console is a class in System namespace

using keyword can be used so that the complete name is not required.

System.Console.WriteLine("xxx")

or

using System;
Console.WriteLine("xxx");

class Sintadi {}
It begins a class ~~definition~~ for the ^{class} named Sintadi. Every app consists of the least one class declaration that's defined by you.

static void Main() {}

It is where the app begins execution. This is known as the entry point for each app one off the methods in a class must be called Main; otherwise app will not execute.

Console.WriteLine("Hello");

class console provides standard input/output capabilities that enable apps to read and display text in the console window from which the app execute

WriteLine method displays a line of text in the console window.

WriteLine method is under console class

Sınıflar!!!

• .NET nedir?

- Double ile terminalde yaparten normal de nüfus
Float ve decimal ile değer belirttiğimizde sonuna
hast eklerse 20undayı2.
- Methodlar, constructorlar dışındakı property denilen
sey nedir?

Degiskenler:

Kullanıcıdan veya veritabanından aldığımız bir
veriyi program boyunca kullanmayı istenir. Bu nedenle
ıçın değişkenlere ihtiyac duyuyoruz.

Degisken tipi

Degisken adı

=

Degiskenin
degeri

atama operatörü

string isimSoyisim = "Musa Akyüz";

String büyük = isimSoyisim.ToUpper();

String küçük = isimSoyisim.ToLower();

String kesit = isimSoyisim.Substring(2,4)

→ Tavanı büyük hafte sonunda MUSAAKYÜZ

→ Tavanı küçük hafte sonunda musaakyüz

→ 2. indexten ok 6 tane sterler sər Ak

DEĞİŞKENLER

char degiskeni: tek karakter setleyen degisken tipi.

byte ve int degiskenleri: byte ve int degiskenler

sayısal degerler setleyenler.

byte (0,255)

+ C# ordineler iti kosa tab tusu (kugol)

Sayısal Değer (ondalık) Değerlerini

Double Decimal Float

double adı1 = 10,2;

double adı2 = 10; \rightarrow Son hafı kaynak 20nuk
double adı3 = 10,9d;

decimal adı4 = 10,2M; \rightarrow Son hafı kaynak 20nuk

float adı5 = 10,2F;

bool adı6 = false;

bool adı7 = true;

DateTime adı8 = DateTime.Now; \rightarrow Sırası sıralı.

TÜR DÖNÜŞÜMLERİ

Bilinçli

Bilinçsiz

Boxing / Unboxing

byte sayı1 = 100;

int sayı2 = sayı1;

int karşıdaki değere
baktı, byte göründü
kabul etti

int sayı1 = 100;

byte sayı2 = (byte)sayı1;

byte residde değere baktı
integer içinde kabul etmedi
ancı byte pointer içinde
byte işaret ketili olduktan
da kabul etti.

Object nesnelerin tür, tarihi, klasörleri, özellikleri
ve bir属性dir.

Boxing

object o1 = 100;

object o2 = 10.92;

object o3 = true;

Unboxing

string matint = "100";

int soyi = int.Parse(matint);

int soyi2 = Convert.ToInt32(soyi);

parse
method
ile

ToInt32
method
ile

Sınıflar olusturulup kullanıldığında kuralların
nesneleri sınıflandırma için kuralları
anatır.

• Nesneleri farklılıkla sınıflar.

Anahtar Sınıfı

Dönüşümler

- Yenidenilebilir
- Durdurulabileceğine
- Hizlanabileceğine

"on" Sınıflar

- Direksiyon
- Motor
- Kettek

Anahtar sınıfı tür

İşlemleri ve davranışlarını

Kapsüleme (örnek bir sınıf tanımlamak
arasıdır).

Yöntem dörtlü sınıfın eklenen ek özelliklere ve
eklenen özellikler ve ek özelliklerin eklenen
bir araya getirilmesidir.

- Yöntem ve verilen bir sınıf içerisinde birleştirilir. ve
bu sayede sınıflandırma, destekler.
- Yöntem ve verilenin kullanımını dorettir.

class örnekler sözlüğü

- C# dilinde yeni bir sınıf tanımlanmak için class
örnekler sözlüğü kullanılır.
- Sınıfın var ve gösterilen sınıf tanımının içinde
kullanılsın tanımlanır.

class Circle

{

 double Area()

 {

 return Math.PI * radius * radius;

 }

 int radius;

}

Örnekte görüldüğü gibi Math sınıfından matematiksel
sabitler içeren orter ve ve matematiksel
hesaplamalar içindir.

— / —

Sınıf içinde Area() gibi fonksiyonlar ve nesnesi gibi alanlar var.

Bu sınıfa ait bir nesne yapmak için new anahtar sözcüğü kullanılır.

Circle c; // Bir daire oluşturur.
c = new Circle(); // Basitçe bir
otur.

Sınıfa ait olan bir nesneye ilk değer atayarsınız.

Sınıfın bir özelliğini aynı türden başka bir
birliğe atayabilirsiniz.

Circle c;

c = new Circle();

Circle d;

d = c;

Sınıflar ister istemez bir diğer öremeli konu oluşturuyor.

default olarak yerler nesneler ve alanlara
ve fonksiyonlara onsenir sınıf.

Erisilebilirliği kontrol etmek için public veya
private anahtar kelimelerini kullanabiliyoruz.

Erişebilirlik

- Bir yöntem veya alan sadece sınıfın
içinden erişebilir olması istenirse o yöntem
veya alan private gibi ile tanımlanır.

- Varsayılan ayar olanač herhangi bir - oradan belirle icermeyen alıcı veya topluyluklar private olur; ancak bu durum da bir şekilde belirtmekte yerar ver.
- Bir yöntem veya alıcı hem sınıf içersinden hem de dışarıdan erişilebilirse o yöntemi ve alıcı public olarak tanımlanır.

class Circle

{

public double Area()

{

 return Math.PI * radius * radius;

}

private int radius;

}

ÖZMELİNDİRME KURALLARI

- Sınıf içersinde public tanımlayıcıları büyük harfle belirler
- Sınıf içersinde private tanımlayıcıları küçük harfle belirler

↳ Bu da firsatı olanač türküler (private olma den) büyük harfle belirler

KURUCULAR (CONSTRUCTOR)

- Bir sınıfın örnegini prototipinde otomatik olarak oluşturmak için bir özel bir yöntemdir
- Sınıf ile aynı adlı sahip topluylardır.
- Parancısalabilirler. Herhangi bir değerle

- Oluşturdığımız sınıfların kurucuları (constructors) otomatik olarak bir tane olur. Oluşturulan sınıfların ve kalkımları.
- Kurs. Kurucular (constructor) oluşturmak için sınıf ile aynı adda bir sınıf tanımlanır. Bu sınıf public olur bir fonksiyon olus.

Class Circle

{

public Circle() // Varsayılan kurucu

{

radius = 0;

{

public double Area()

{

return Math.PI * radius * radius

{

private int radius;

{

Yukarıdaki kodda bir sınıfın var. Oluşturularak tüm örneklerin radius değeri sıfır (0) olur ve bunu değiştirmek yolu yoktur.

- Bu sınımda örnek neden kurucuya örneklere gidecek (overloading) yapısı farklı olursa önceki yerde bir tane oluşturuyor

class Circle

{

public Circle() //norseyilar kurver

radius = 0;

}

public Circle(int initRadius);

radius = initRadius;

}

public double Area();

{

return Math.PI * radius * radius;

}

private int radius;

}

KİŞİSİ SINIFLAR

Bir sınıf çok sayıda yöntem, alan, türkçe ve diğer yöntemlere sahiptir. Bu sebeple sınıf olursa genisleyebilir. Geniş bir sınıf teker teker prototiplenmiş form olarak parçalar oluşturmak mümkündür.

- Bir sınıfın birçok değişkeni aynı zamanda prototip adıton sözülebilir. Bu

örneğin Circle sınıfını 'ki parçaya, bir sonraki sayfada gönderecekti şekilde

'ki parçaya eratabiliriz.

circle.cs

partial class Circle

public Circle()

{
 radius = 0;

public Circle(int initRadius)

{
 radius = initRadius;

circle2.cs

partial class Circle

public double Area()

{
 return Math.PI * radius * radius;

private int radius;

Static Yöntemler

Daha önce Meth sınıfını kullanır

Meth.PI

Meth.Sqrt(25);

Burada Meth sınıfına ait bir nesne
yaratmak yerine direkt olarak metodu
çalışmaya yarar.

Meth sınıfına ait bir nesne yaratmak

{Meth m = new Meth();

{double d = m.Sqrt(25);

→ Bu şekilde de yapabilebilir ancak hala
nesne değil static bir yöntem çok
daha iyi bir şekilde kullanılabılır.

C# dilinde tüm yöntemler bir sınıf
 içinde birden fazla olur.

- Ancak bir yöntem verya also static
olarak tanımlanırsa sınıf adını kullanarak
yöntemi çağrılın yada olağan kullanılarak

class Meth

{

public static double Sqrt(double d)

}

}

Static bir yöntem temsil eder.
bu yöntem sadece static olarak kullanılabılır.

Static Member

Paylaşılmış olabilir.

Aşırı toplam sayı Static memberler S_2

İkinci bir. B. Sayede sınıfa ait tüm
nesneler tarafından paylaşılan bir
olur yani toplam Static olmayan olab-
ker bir nesne oluşturmak için gerekli

Circle sınıfına oluşturularak saye-
bir paylaşılmış Static olur yani olur.

class Circle

{

 public Circle() //mesajları kuvvet

{

 radius = 0;

}

 NumCircles++;

 Constuk

//operator

 public Circle(int initRadius) {

{

 radius = initRadius;

}

 NumCircles++;

--

 private int radius;

 public static int NumCircles = 0;

3

const Andıtor Sözçü

- Static bir class deponindr hıca depon-yecegin bilgisinek kullanısimız.
- const de classın oluşturulan static kelimesi bulunmasa da bu class static olarakdır.
- Sadece int ve double türündeki class const kullanıbılır.

class Meth

{

public const double PI = 3.14592---;

Static Sınıflar

C# dilinde sınıfları da static olabilir.
Static kılın sınıf sadece static yöntem
ve alanlar içindir.

Bu tür sınıfları oluşturmakta ona
standart da bir yöntemi ve alanları
toplarmasıdır.

Static sınıfın oluşturduğu yerde olsa
veya new anahtar kelimesiyle bu sınıf
artı neye düşerse.

Bir long da bir alanın getirilese
static sınıf versayıla bir tane veya sahip
olabilir.

Ancak kurucular da static olmak zorunda
gerekli.

Eğer sınıfı sınıflarla static metodları almak için sınıfın oluşturulan set.

public static class Math

{
 public static double Sin(double x)
 { - } }

public static double Cos(double x)
{ - }

}

3

ANONYM SINIF

- Adı olmayan sınıflardır.
New ornek Sınıfı oluşturmak için
Sınıf içinde olmasının steadığınız
olarak ve değerler toplanabilir.

burada Anonym Nınam = new { } = "Ali"

Bu durumda "Sınıf" adında public olur
başlığı.

Bu sınıfları sınıfları içinde döngüde
sınıf içi kodları adını verebiliriz.

Ayrıca sınıfın sınıf içi bölgeleri
olusturabilmiz

Anonym Nınam = new { } = "Ayşe"

Diziler - Koleksiyonlar

Diziler sıralı veya ögesler serisi dir

Bu dizi de türün ögesler aynı turudur.

Bir dizinin ögesleri bilgilerin hafızasında birlesik olarak tutulur.

`int [] numaralar;`

`numaralar = new int[10];`

Bir dizi new anahtar sözcüğü ile oluşturulur.
Bu da dizinin elemanlarına verilecek değerler atanır.

`int → default(0)`

`String → default(null)`

`bool → default(false)`

`Random r = new Random();`

`int [] number = new int[4] { r.Next() % 10,
 r.Next() % 10, r.Next() % 10,
 r.Next() % 10 }`

Dizideki her nesnenin türü aynı olmalıdır.

`var isimler = new [] { "Ali", "Veli", "Ayşe" }`

Dizi elemanlarına index numaralarıyla erişim
index "0" sıfırından başlar.

`int [] pins = { 9, 3, 7, 5 }`

`foreach (int pin in pins)`

`{`

`}`

• Next()

• Length

• Clone()

Amay, Copy(---)

Diziler de içinde koleksiyonlar kullanarak ögelerini bir dizeye getirebiliriz.

System.Collections

Sınıf ögelerini nasıl oluşturabiliriz. (Ögelerin türleri, objecttipi)

Queue First In First Out

Yeni oluşturuları sonra sıraya katılır ve sırayı on ekledeğiştirir.

Queue numbers = new Queue();

numbers.Enqueue(number); // sayıyı ekler

numbers.Dequeue(); // sayıyı siler

Stack Last In Last Out

Son eklenen ilk olur.

Yeni oluşturuları ekler ve sırayı ters çevirir.

Stack numbers = new Stack();

numbers.Push(number); // elemanı ekler

numbers.Pop(); // elemanı siler

Hash tablosu koleksiyon sınıfı

Bir dizi içinde tam sayı tipindeki değerler

ile string, double ve Time türünde

değerlerin başlangıç değerinin

HashTables掌贮器 / deger \rightarrow hali ikenir
Gör eleman iken haliye boyutunu artabılır.
HashTablesinde foreach kullanırsak Dictionary
Entry
olduğunu

Hashtable ages = new Hashtable();

ages["John"] = 42;

ages["Diana"] = 43;

foreach (DictionaryEntry element in ages)
{

}

SortedList

Dizi vs Koleksiyon

Sıralanır	Bırıncı
olarak elemanı	Genetik
türeng	yaz.
belirtilmeli	

Dizinin boyutu	Bütçeli
Sabitdir	Otomatik
Boş olabilir	yer almaz.
Boş olamaz	
manzıf değil	

Biri dizi bir boyutlu deðil.	(de boyutlu yapılabilir eve yoktur.)
---------------------------------	--

— / —

Kalitim

- programlamada farklılıkla sınıflandırılmış olabilecek farklı sınıfların ortak özellikleri temin etmek için oluşturulan sınıfları organize edebilmek için kalitim sistemini kullanır.

class DerivedClass : BaseClass