



# JOURNAL of SOCIAL and HUMANITIES SCIENCES RESEARCH (JSHSR)

Uluslararası Sosyal ve Beşeri Bilimler Araştırma Dergisi

**Received/Makale Geliş** 11.10.2021  
**Published /Yayınlanma** 31.12.2021  
**Article Type/Makale Türü** Research Article

**Citation/Alıntı:** Of, M. & Kılıçaslan, İ. (2021). Bilgisayar programlama dillerinin işletmelere sağladığı katkılar. *Journal of Social and Humanities Sciences Research*, 8(78), 3284-3292.  
<http://dx.doi.org/10.26450/jshsr.2841>



**Öğr. Gör. Mustafa OF**

<https://orcid.org/0000-0002-7924-9073>

Kocaeli Üniversitesi, Kocaeli Meslek Yüksekokulu, Kocaeli / TÜRKİYE



**Öğr. Gör. İsmail KILIÇASLAN**

<https://orcid.org/0000-0002-8443-9912>

Kocaeli Üniversitesi, Ali Rıza Veziroğlu Meslek Yüksekokulu, Kocaeli / TÜRKİYE

## BİLGİSAYAR PROGRAMLAMA DİLLERİNİN İŞLETMELERE SAĞLADIĞI KATKILAR

## THE CONTRIBUTION OF COMPUTER PROGRAMMING LANGUAGES TO BUSINESSES

### ÖZET

Programlama dili, programcının belirli bir düzende oluşturulmuş bir algoritmayı (İş Akışı) ifade etmek için kullanılan bir komut veya ifadeler topluluğudur. Programlama dilleri, bilgisayara hangi işlemleri yapacağını anlatmayı sağlayan komut veya fonksiyonlara sahiptir. Alt seviye, orta seviye ve yüksek seviye olmak üzere üçe ayrılırlar. Yüksek seviyeli programlama dilleri konuşma diline çok yakın komut veya fonksiyonlara sahiptir. Bu yüzden öğrenilmesi daha kolaydır. Örneğin C#.Net (C Sharp), Python veya Visual Basic.Net. Alt seviye diller bilgisayarın makine diline (İkili sayı sistemi) çok yakın dillerdir ve bu dillerde program yazmak oldukça zordur. Fakat çok hızlı çalışırlar. Orta seviyeli diller hem alt seviye hem de yüksek seviye özelliğini taşırlar. Örneğin C++ Programlama dili. Bir dili öğrenmeye başlarken öncelikle niçin kod yazılması gerektiğinin belirlenmesi gereklidir. Daha sonra geliştirilecek uygulamanın algoritması ya kâğıt üstünde veya zihinde netleşmesi gerekir. Sonraki adımda ise hangi programlama dili seçilecek buna karar verilir. Genel amaçlı bir uygulama geliştirmek için yüksek seviyeli bir dil olan C# seçilebilir. Küçük uygulamaları geliştirerek işe başlanması iyi bir seçim olacaktır. Kod yazımı sırasında hatalar meydana gelebilir. Fakat sabırlı bir şekilde hataların çözülmesi gereklidir. Mümkünse seçilen programlama dili ile ilgili kurslar alınmalıdır. Bu konuda yönlendirici olacak bir eğitmen edinilmesi çok önemlidir. Başkalarının yazdığı örnek kodların incelenmesi, dil öğrenimini oldukça etkileyecektir. Yeni bir şeyleri geliştirebilmenin hazzını programlama dilleri öğrenerek kolayca elde edilebileceği unutulmamalıdır. Bu çalışmada bilgisayar programlama dillerinin işletme çalışanlarına sağladığı katkılar açıklanacaktır. Sürekli yapılan rutin işlemlerin programlama dili ile yazılan uygulamalar sayesinde hatasız ve otomatik olarak nasıl yapılacağı belirtilecektir. Bir programlama dilinin çalışana sağladığı katkılardan biri olan analitik düşünme yeteneği irdelenecektir.

**Anahtar Kelimeler:** Programlama Dilleri, Makine Dilleri, Algoritma, Python, C#.Net, Visual Basic.Net, C++.

### ABSTRACT

A programming language is a set of commands or expressions used to express a programmer's algorithm created in a particular order. Programming languages have commands or functions that tell the computer what operations to do. They are divided into three as low level, intermediate level and high level. High-level programming languages have commands or functions very close to spoken language. So it is easier to learn. For example C#.Net (C Sharp), Python, or Visual Basic.Net. Low-level languages are very close to the machine language of the computer (Binary number system), and it is very difficult to write programs in these languages. But they work very fast. Intermediate languages are both low-level and high-level. For example C++ programming language. When starting to learn a language, it is necessary to determine why code should be written first. The algorithm of the application to be developed later must be clear either on paper or in the mind. In the next step, it is decided which programming language to choose. C#, a high-level language, can be chosen to develop a general-purpose application. It would be a good choice to start by developing small applications. Errors can occur while writing code. But it is necessary to resolve the errors patiently. If possible, courses related to the chosen programming language should be taken. It is very important to get a trainer who will guide you in this regard. Examining sample codes written by others will

Issue/Sayı: 78

Volume/Cilt: 8

[jshsr.org](http://jshsr.org)

ISSN: 2459-1149

greatly affect language learning. It should not be forgotten that the pleasure of developing something new can be easily obtained by learning programming languages. In this study, the contributions of computer programming languages to business employees will be explained. It will be stated how the routine operations that are done continuously will be done error-free and automatically the applications written in the programming language. Analytical thinking ability, which is one of the contributions of a programming language to the employee, will be examined.

**Keywords:** Programming Languages, Machine Languages, Algorithm, Python, C#.Net, Visual Basic.Net, C ++.

## 1. GİRİŞ

Gittikçe sanallaşan ve dijital hale dönüşen bir dünyaya doğru hızla gidiyoruz. Bilgisayarlar ve dolayısıyla bilgisayar yazılımları günlük yaşantımızın her anında karşımıza çıkmakta. Firmalar, ürünlerine erişim hizmetlerinin kalitesini arttırmak için geliştirmiş oldukları yazılımların reklamlarını duyurarak daha geniş kitlelere ulaşmak arayışındalar. Örneğin İnternet bankacılığı ile mevduat yönetiminin kolaylaştırılması. İnternet'in hayatın her noktasına girmesi sonucunda hizmetlere hem Web hem de Mobil ortamlardan erişim vazgeçilmez hizmetlerin başında gelmektedir. Bahsedilen bu tür işlerin tamamı yazılımlarla yapılmaktadır. Yazılımları geliştirdiği ortam, programcıların kullandığı bilgisayar programlama dilleri adı verilen kendisi de bir yazılım grubudur. Ticari anlamda yazılımın gücünü, dünyada en çok kullanılan işletim sistemi (Operating Systems) ve çeşitli yazılımların geliştiricisi olan Microsoft'un kurucusu ve sahibi Bill Gates'in kişisel servetinin 132 milyar dolar olması (2021 yılı itibari ile) durumu yeteri kadar açıklamaktadır.

Bu çalışmada bilgisayar programlama dillerinin önemine vurgu yapılarak tanımı yapılacak ve bir dilin öğrenilme noktasındaki incelikleri tartışılacaktır. Belki de en çok bilgisayarı iş yerlerinde kullanan bir işletme çalışanlarının programlama dillerini öğrenmeye nereden başlayacağı izah edilecektir.

## 2. PROGRAMLAMA DİLLERİ

Programlama dili, program geliştiricisinin ortaya çıkmış olan belirli bir algoritmayı veya yolu gerçekleştirmek için kullandığı belli başlı kuralları olan kod bir yazım şeklidir. Bir yazılım geliştiricisi bilgisayar donanımına yaptırmak istediği işlemler için programlama dilleri kodlarını kullanır. Programlama dilleri, yazılımcının bilgisayar donanımına ne tür veriler üzerinde nasıl bir işlem yapacağını, bu verilerin nerelerde depolanıp kullanabileceğini, hangi şartlara uyarak hareket edeceğini tam olarak açıklamasını sağlar. Şu ana kadar irili ufaklı 2000'den fazla programlama dili geliştirilmiştir. Bu dillerden bir kısmı popüleritesi sayesinde beğenilmiş ve kullanılmıştır.

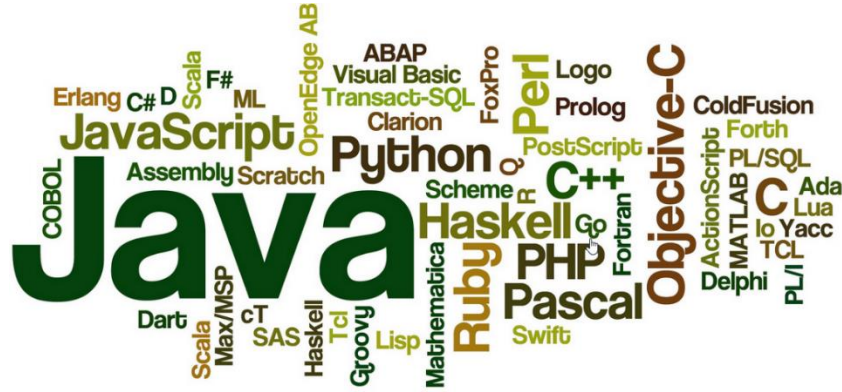
Programlama dilleri insan konuşma diline yakınlığına bağlı olarak 3 gruba bölünebilir.

Alt seviye programlama dilleri: Bilgisayarın makine dili kodu denilen ikili sayı sistemine oldukça yakın programlama dilleridir. Bilgisayara yani donanıma hâkimiyeti yüksektir. Bu programlama dillerini bilen kişilerin mikro işlemciler hakkında bilgi sahibi olması gereklidir. (Makine dili, Assembly programlama dili). Alt seviye programlama dilleri ile program yazmak insan konuşma diline uzak olduğu için biraz daha zordur. Her türlü uygulamayı bu dillerle yazmak mantıklı değildir.

Orta seviye programlama dilleri: Oldukça esnek olan bu diller hem üst hem alt seviye programlama yapabilirler. Alt seviye dillere oranla nazaran daha anlaşılır seviyededirler. İki seviye ile iletişimde bulunması, tercih sebebini arttırır. Daha kolay kod yazma imkânı doğurur. Deyimleri günlük hayattaki kelimelere yakındır. (C programlama dili, C++ programlama dili)

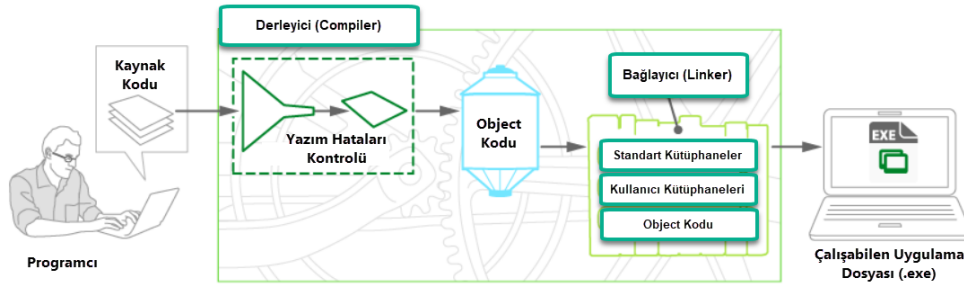
Yüksek seviye programlama dilleri: Olay yönlendirmeli (Events Driven) programlama dilleri olarak da isimlendirilirler. Hızlı ve etkin programlama dilleri bu gruptadır. (C#.Net, Visual Basic.Net, Delphi, Python) Diğer programlama dillerine göre daha basit olduğu ve uygulanabilirlik seviyesinin kolaylığından dolayı yazılım geliştirmeye yeni başlayanlara yönelik en uygun diller üst seviye programlama dilleridir. Bütünleşik yazılım editörü (Integrated Development Editor) sayesinde görsel (Visual) bir ortamı olan programlama dilleri ile daha kolay uygulamalar geliştirilebilir. Örneğin Visual Studio, Netbeans, Eclipse.

Yüksek seviyeli programlama dillerinin bir kısmında uygulamanın çalışabilmesi için bilgisayarın makine dili olan 0 ve 1'lere dönüştürülmesi gerekmektedir. Bu işlem için ilgili programlama dilinin derleyicisi (Compiler) kullanılır. Böylece programlama dili ile yazılmış olan kaynak programın kodları, bilgisayarın çalıştırabileceği makine diline dönüştürülür. Kodların bulunduğu kaynak programın içeriğinin değiştirilebilir. Ancak derlenmiş olan programın içeriğine tam olarak erişim imkânı yoktur. Ters mühendislik metotları ile derlenmiş uygulamaların bir kısmı geri döndürülebilir.



Şekil 1: Düünden Bugüne Kadar Gelmiş Birçok Programlama Dilleri

Aşağıdaki şekilde bir Microsoft Windows işletim sisteminde kendi başına çalışabilen makine diline dönüşmüş bir uygulamanın derleme ve bağlama aşamaları gösterilmektedir. Kaynak kodu bir metin dosyasıdır ve yazılım geliştiricisi kodlarını bu dosyalara yazar. Bir metin dosyasının kendi başına çalışma özelliği yoktur. İşletim sistemine bağlı olarak kendi başına çalışabilme özelliği yani makine dili denilen ikili sayı sistemine dönüştürülmesi gereklidir. Bu işlemleri derleyici (Compiler) ve bağlayıcı (Linker) yazılımlar yapmaktadır. Derleyici, kodların programlama dilinin yazım kurallarına uygun olup olmadığını denetler. Eğer bir hata varsa bunu geliştiriciye bildirir. Hata yoksa orta seviye bir dosya olan Object dosyasını üretir. Object dosyası artık makine diline dönüştürülmeye hazırdır. Bağlayıcı (Linker) yazılımları aracılığı ile Object dosyası kendi başına çalışabilen makine dili dosyasına dönüşür. Örneğin Windows işletim sistemlerinde Exe uzantısı olan bir dosya. Tek başına çalışabilme özelliği işletim sistemlerine göre farklılıklar gösterir. Mobil işletim sistemi olan Android, küçültülmüş Java sanal makinesi olan Dalvik sanal makinesi yazılımı üzerinde uygulamayı çalıştırmaktadır.



Şekil 2: Microsoft Windows İşletim Sisteminde Kendi Başına Çalışabilen (Exe) Bir Uygulamanın Oluşturulma Evresi

## 2.1. Assembly Programlama Dili

Assembly dili, geliştiricilerin kodlarıyla daha fazla verimlilik elde etmek için projelerinde kullanabilecekleri güçlü bir araçtır. Assembly dili, herhangi bir platformda kullanılabilen düşük seviyeli programlama dillerinden biridir. İkili sayı sisteminden (0 ve 1) ibaret olan makine diline nazaran anlaşılması kolay olan bir dildir. Deyimleri, iki veya üç harfli kısaltılmış metinlerden oluşmaktadır. Makine dilinden sonra en hızlı çalışan programlama dilidir. İnsan konuşma diline daha yakın olan yüksek seviyeli programlama dillerinin hız problemi, Assembly kodları ile bütünleştirilerek aşılabılır. Donanımla hızlı ve direkt ilişkisi olan uygulamaları geliştirme aşamasında başvurulan bir dildir. (Lyashko, 2017:27).

Assembly programlama dilinin genel olarak kullanıldığı yerleri şöyle sıralayabiliriz;

- Donanıma direkt erişmek istenildiği yerlerde,
- Bir donanımı işletim sistemine tanıtan sürücü (Device Driver) dosyalarını oluşturma işleminde,
- Elektronik yongalara (Chip) güncelleme vb. işlemleri yapmak için program yükleme işlemlerinde,
- İşletim sistemlerinin geliştirilme aşamalarında (Operating System)
- Sistemlerin zayıf noktalarını tespit etme işlemlerinde, şifre kırma vb. işlemlerde,
- Başka sistemlere bulaşabilen virüs adı verilen uygulamaları geliştirme işlemlerinde.

```

Title Yazı Programı
dosseg
.model small
.stack 100h
.data
my_message db 'Selam!',0dh,0ah, '$'
.code
main proc
    mov ax,@data
    mov ds,ax
    mov ah,9
    mov dx,offset my_message
    int 21h
    mov ax,4C00h
    int 21h
main endp
end main

```

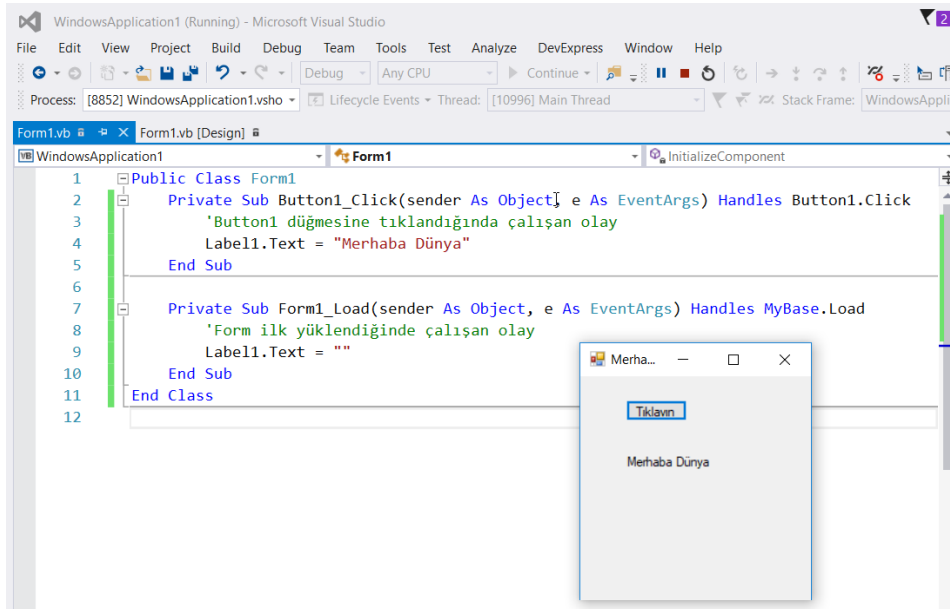
Şekil 3: Ekrana Bir Yazı Yazan Assembly Kod Örneği

Yukarıdaki şekilde gösterildiği gibi ekrana bir yazı yazmak için birçok satıra ihtiyaç vardır. Bunun sebebi yapılacak işlemi aracı kullanmadan direk donanıma bildirmek içindir. Çalışma hızı buradan gelmektedir. Düşük seviyeli dil olmasından dolayı işletmedeki bir çalışanın rutin olarak yapılan işlemler için başvuracağı bir dil değildir. Yerine yüksek seviyeli programlama dilleri tercih edilir. Örneğin yüksek seviyeli dillerden biri olan Python ile bir mesaj yazmak için sadece print("Merhaba") deyimini yeterlidir.

## 2.2. Visual Basic.Net Programlama Dili

Görsel programlama dillerinden biri olan Visual Basic 'in en son sürümüdür. Görsel programlama dilleri, programcıya büyük kolaylıklar sağlamıştır. Bu özelliğe ilave olarak Microsoft'un ortak çalışma çatısı olan .Net Framework'a üye olması sayesinde .Net kütüphanesinin tüm imkânlarından faydalanmıştır. Kütüphaneler, içerisinde birçok fonksiyonu barındıran yapılardır. Halen API (Application Program Interface) olarak da anılırlar. Programlama dili dünyasına 1980'li yıllarda adım atanlar genelde Basic (Beginners All Purpose Instruction Code) veya GWBasic programlama dili ile tanışmışlardır. VB.NET ise bu alışkanlığı güçlü kütüphaneleri ile devam ettirmiştir.

VB.NET ile program geliştirmek için iyi bir geliştirme ortamı olan Visual Studio'nun ücretsiz sürümünü yani Community sürümünü şu adresten indirebilirsiniz. <https://www.visualstudio.com/tr/>



Şekil 4: Visual Studio geliştirme ortamında VB.NET ile yazılmış bir Windows form uygulaması için örnek kod

## 2.3. C Sharp (C#) Programlama Dili

C# Microsoft tarafından geliştirilmiş olan bir programlama dilidir. C++ ve Java dillerine benzer yönleri oldukça fazladır. Ancak C# dilinin bu benzerliği yanında farklı yönleri çoktur. Örneğin C#, C++'dan farklı olarak tamamı ile nesne yönelimli programlama yaklaşımına sahiptir. Java'dan farklı olarak ise

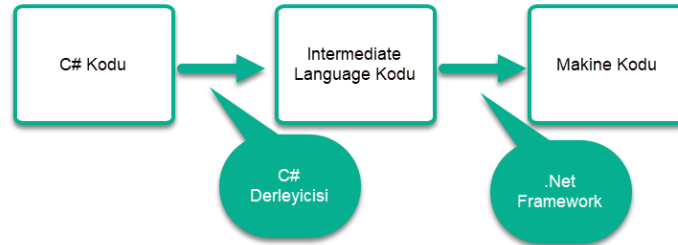
C#'ta gösterici (pointer) kullanılabilir. Böylelikle eski yazılım bileşenleriyle uyumlu bir şekilde çalışılabilir.

C# Programlama dili ile şunları yapabilirsiniz:

- Konsol uygulaması oluşturma
- Windows sistemlerinde çalışan uygulama geliştirme
- Web ortamında çalışan ASP.NET uygulaması geliştirme
- Web servisleri yazma
- DLL (Dynamic Linked Libraries) yazma

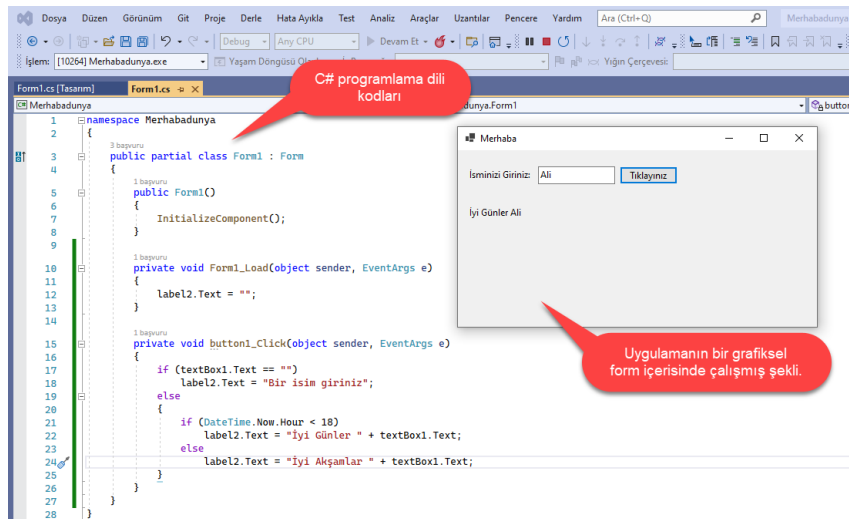
ve daha birçok işlem...

Başlangıçta Microsoft, yalnızca Windows için bir CLR (Common Language Runtime) uygulaması geliştirdi ve .NET uygulamalarını Linux veya OSX'te kullanmak isteyenler, Mono Project (<http://www.mono-project>) gibi bir üçüncü taraf .NET Framework uygulamasına başvurmak zorunda kaldılar. Veya DotGNU Portable.NET (<http://dotgnu.org/pnet.html>) gibi çözümler kullanarak farklı işletim sistemlerinde çalıştırıldılar. Ancak 2015'te Microsoft, Windows, Linux ve Mac OSX üzerinde çalışabilen .NET Framework'ün devamı olan .NET Core'u piyasaya sürdü. .NET Core açık kaynaklı (Open Source) bir yazılım çerçevesidir. Bu makalenin yazıldığı tarih olan 2021 yılı sonunda .NET Core 6.0 olarak kullanılabilir haldedir (Jayden, 2016:10)



Şekil 5: C# Kodunun Makine Diline Dönüştürülmesi

C# programlama dilinin çekici hale gelmesinin altında yatan önemli bir sebep ise geliştirici editörü olan Visual Studio uygulamasıdır. Geliştiriciye büyük kolaylıklar sunan bu uygulama ile bir yazılımı ortaya çıkarmak çok kolay hale gelmektedir.



Şekil 6: C# Programlama Dili ile Geliştirilmiş Bir Form Uygulaması

## 2.4. Python Programlama Dili

Python öğrenilmeye değer bir programlama dilidir. Python'u öğrenmek, sizi yalnızca son derece karmaşık görevleri en basit şekilde yapmaya motive etmekle kalmayacak, aynı zamanda geleneksel programlama paradigmalarının mitlerini de yıkacaktır. Programlama şeklinizi değiştirecek ve dolayısıyla bir sorunu çözecek bir dildir. Python, 1980'lerin sonlarında Guido Van Rossum tarafından

hazırlanmış güçlü, prosedürel, nesne yönelimli, işlevsel bir dildir. Python, Britanyalı bir komedi grubu olan Monty Python'dan adını almıştır. Dil şu anda çeşitli uygulama alanlarında kullanılmaktadır. Bunlara yazılım geliştirme, web geliştirme, Masaüstü GUI (Graphical User Interface) geliştirme, eğitim ve bilimsel uygulamalar dâhildir. Bu nedenle, ihtiyaçların neredeyse bütününe kapsamaktadır. Çok tercih edilmesinin altındaki en temel özelliği basit olmasından kaynaklanmaktadır.

Bahsi geçen görevleri gerçekleştirmek için birçok üçüncü taraf modülü bulunmaktadır. Örneğin, temiz ve hızlı geliştirmeye adanmış son derece popüler bir Web çerçevesi olan Django, Python üzerinde geliştirilmiştir. Gelişmeye açık herkesin katkıda bulunabileceği birçok deposu mevcuttur. Bunlardan en çok kullanılanı pypi.org projesidir (<https://pypi.org>). İhtiyaca bağlı olarak bir kitaplık arayışı içerisinde olan bir yazılım geliştiricisi, başka bir geliştiricinin paylaşımı ile bu depoda yüksek bir ihtimalle aradığını bulacaktır. Ayrıca günümüzün en önemli kavramlarından biri olan Yapay zeka uygulamaları için birçok kitaplığı depoda bulmak oldukça kolaydır. Python ile uygulama geliştirme süreleri oldukça kısadır. Geliştiricinin bir Python kitaplığını sistemine eklemesi için yapması gereken çok basit bir komuttur. Örneğin matematiksel çözümlene işlemleri için SciPy kitaplığını sisteme indirmek için şu komutu, Windows komut satırından (Komut İstemi) yazmak yeterlidir.

```

Seç Komut İstemi - pip install scipy
no such option: --update
C:\Users\Mustafa>pip --version
pip 21.2.3 from C:\Users\Mustafa\AppData\Local\Programs\Python\Python310\lib\site-packages\pip (python 3.10)
C:\Users\Mustafa>pip install scipy
Collecting scipy
  Downloading scipy-1.7.3-cp310-cp310-win_amd64.whl (34.3 MB)
    | 23.8 MB 6.4 MB/s eta 0:00:02

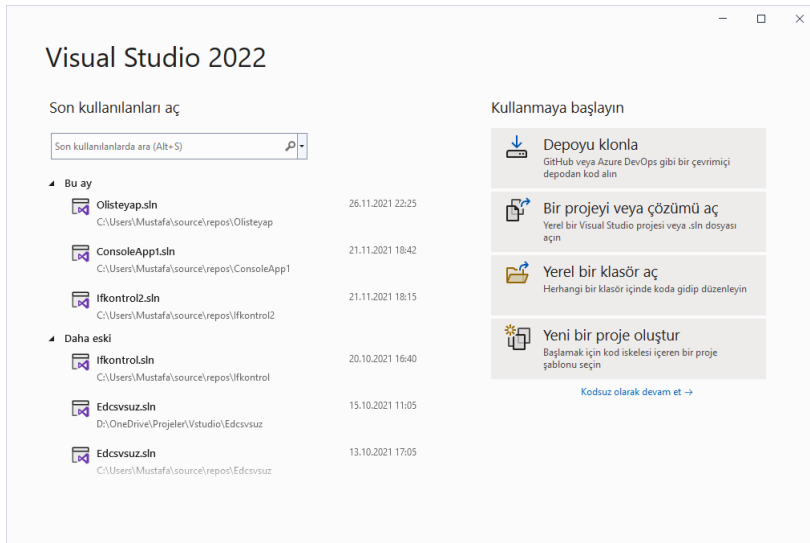
```

Python, uygulamaları bilimsel analizlerde de kullanılabilir. Mühendislik ve Matematik için SciPy gibi kitaplıklar kullanılır. İstatistik ve makine öğrenimi alanında çalışanlarınız, bu kitaplıklardan bazılarını son derece yararlı ve kullanımı kolay bulacaktır. SciPy, MATLAB benzeri özellikler sağlar ve çok boyutlu dizileri işlemek için kullanılabilir (Bashin, 2019:18).

Komutlarının basitliği ve kolay öğrenilmesi sebebiyle çok tercih edilen bir dil olmuştur. İlk başlayanlar için önerilen bir programlama dilidir. Birçok farklı geliştirme editörleri vardır (IDE: Integrated Development Editor). PyCharm, Thonny, Visual Studio Code bunlardan sadece bir kaçıdır.

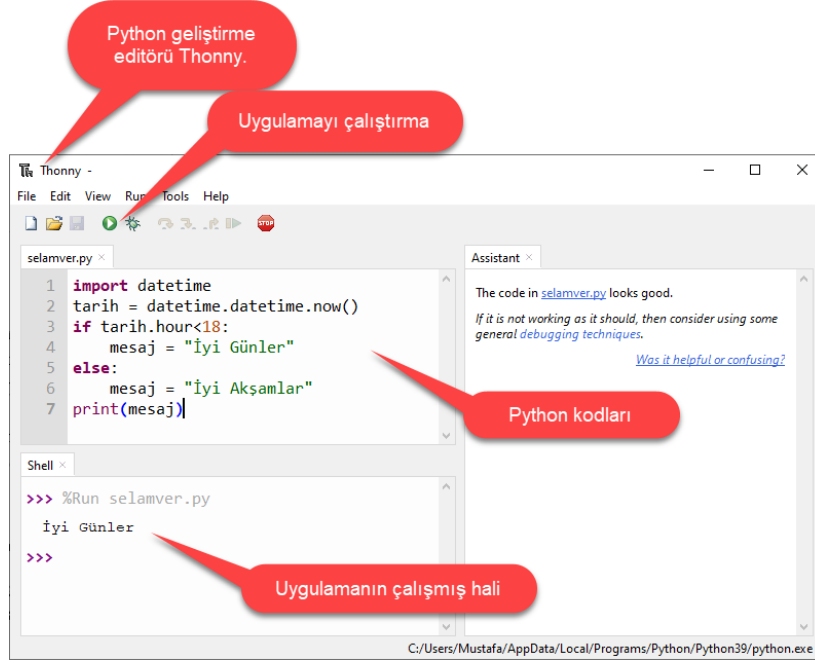
### 3. YAZILIM GELİŞTİRMEYE NEREDEN BAŞLANMALI

Analitik düşünme yeteneği olan herkes bir yazılım geliştirme adaydır. İşe öncelikle programlama dili seçimi ile başlanmalıdır. Yüksek seviyeli dillerden VB.NET, C#.NET veya Python programlama dillerinden biri seçilerek işe başlanabilir. Öncelikle iyi bir Programlama dili IDE (Integrated Development Environment) ortamı yani geliştirme ortamı edinerek başlanabilir. Visual Studio bu anlamda iyi bir seçim olacaktır. Ücretsiz sürümü olan Visual Studio Community, internetten ücretsiz olarak indirilebilir. Bu ortam sayesinde hem masaüstü, web ve bazı ilavelerle mobil uygulama geliştirilebilir.



Şekil 7: Visual Studio Community Yeni Proje Geliştirme Ekranı

Python programlama dilinin yazım kurallarının daha hafif olması ve kolay öğrenilmesi nedeniyle ilk yazılım geliştirmeye yeni başlayanlara yönelik olarak tercih edilen dillerin başında gelmektedir. Yazılım geliştirme aşamasında bir harfin bile yanlış yazılması uygulamanın çalışmasını engelleyecektir. İlk başlayanların hevesi daha başlangıçta kaçmış olacaktır. Python'ın asıl amacı kolay olması ile programcılığı herkesin öğrenmesini sağlamaktır. Aşağıdaki kodlar ile gün içindeki saatin durumuna göre selam veren bir uygulama yapılabilir. Satır sayısına dikkat edilirse çok kısa sürede yapılabilen bir uygulama olduğu net bir şekilde ortadadır.



Şekil 8: Thonny Editörü ile Örnek Bir Python Uygulaması

### 3.1. Yazılım Geliştirirken Nelere Dikkat Edilmeli

Bir programlama dilini kullanarak uygulama geliştirirken aşağıdaki kurallara dikkat edilmesinde büyük yarar vardır (Of, 2006):

- Temel programlama dili kavramları öğrenilmeli. Değişkenler, döngüler, şart deyimleri, hata yakalama.
- IDE (Integrated Development Environment) adı verilen uygulama geliştirme ortamları kullanılmalı. Birçok özelliği öğrenilmeli. Örneğin Visual Studio, çok güçlü bir uygulama geliştirme ortamıdır.
- Değişken, metod vb. yapıların isimlendirmelerinde genel kurallara uyulmalı. İngiliz alfabesinin dışında bir karakter kullanılmamalı, noktalama işaretleri kullanılmamalı, boşluk karakteri kullanılmamalı. Sayı ile başlamamalı. Çok uzun ve çok kısa olmamalı. Niçin kullanıldığını belirten bir isim olmalı. Örnek: birimfiyati, hizsabit
- Kodların başlangıç ve bitiş blokları, birbiri ile uyumlu olmalı. Dikey olarak aynı hizada olmalarına özen gösterilmeli. Açıklama satırları ile başlangıç ve bitiş blokları birbirleri ile bağlanmalı.

Örnek C# :

```
ab = 5;
while(ab<10)
{
    ab++;
    Console.WriteLine("Sayaç :"+ ab + "\n");
} //while bitışı
```

- Yazılan birçok kodun başlangıcı ve sonuna yorum yani derlemeye tabi olmayan açıklama satırları konulmalı. Bu açıklamalar daha sonra kodlar değiştirirken çok işe yarayacaktır.

Örnek C# : tutar = adet \* birimfiyat; //Tutar, ürün adeti ile birim fiyatının çarpımına eşittir.

- Dikey olarak bakıldığında içe doğru ve dışa doğru girintiler kullanılmalı. Bu sayede bir kod bloğunun nerede başlayıp nerede bittiğini görmüş oluruz.

Örnek C# :

```
for(int i=0; i<5; i<5)
{
    Console.WriteLine(i);
    Console.WriteLine("\n");
}
```

- Değişken tipleri seçiminde uygun aralıklara uyulmalı. Örneğin müşterinin yaş bilgisini tutmak için bir değişken tipi seçerken bir insan yaşının alabileceği bir sayısal aralıkta değişken tipi seçilmeli.

Örnek C# :

```
int musteriyasi=45;//Burada int tipi -2147483648 – 2147483647 aralığında bir sayı
alabilir. Yaş için çok büyük. Bunun yerine
sbyte musteriyasi = 45; //sbyte tipi -128 ile 127 arasında bir değer olabilir.
```

- Büyük ölçekli projeler geliştirildiğinde ve bir kod bloğunun birden fazla yerde kullanılması gerektiğinde mutlaka NYP (Nesne Yönelimli Programlama) teknikleri kullanılmalı. Bu sayede kod satırları daha da azalacak ve bakım işlemleri çok daha kolaylaşacaktır.

Örnek C# :

```
public class Selam{
    public void SelamVer(string kime)
    {
        Console.WriteLine("Merhaba" + kime);
    }
}
```

Class ve metodun kullanımı

```
Selam selam1=new Selam();
```

```
Selam1.SelamVer("Mustafa");//Konsol ekranına "Merhaba Mustafa" yazacaktır.
```

```
Selam1.SelamVer("Ahmet Eren");//Konsol ekranına "Merhaba Ahmet Eren"
yazacaktır.
```

- Hata yakalama deyimleri kullanılarak çalışma zamanı hataları (Run Time Errors) kontrol altına alınmalı. Örneğin C# programlama dilinde try-catch blokları kullanılarak hata kontrol altına alınabilir.

C# Kodu

```
try {
    //Hata oluşturabilecek kod veya kodlar
}
catch {
    //Hata oluştuğunda çalıştırılacak kodlar
}
```



- Uygulamanın çalışabildiği donanım sayısı arttı. Şimdilerde İnternet ortamında ve Mobil ortamda çalışmasını istediğimiz birçok yazılım bulunmakta. Bu yüzden uygulamaların taşınabilir olma durumuna göre yazılım geliştirme yapılmalıdır.
- Bol bol örnek uygulamalar geliştirilmeli. Örnek alınan hazır bir kodu çözümlenerek yeni özellikler eklenebilmelidir.

#### 4. SONUÇLAR

Program geliştirmek sanıldığı kadar zor bir olay değildir. Analiz yeteneği olan ve bilgisayar dünyasına yabancı olmayan herkes bir uygulama geliştirebilir. Gözle görünen yeni bir şeyi geliştirmenin vermiş olduğu haz duygusunu yaşayabilir. Yüksek seviyeli programlama dilleri ile işe başlama çok daha mantıklı olacaktır. İyi bir geliştirme ortamını (IDE) kullanarak bir uygulama (Application) ortaya çıkarmak çok basittir.

İşletmelerde birçok çalışan, zamanının büyük bir kısmını rutin işlemlerle geçirmektedir. İşlemlerin büyük bir kısmı da bilgisayar ortamında yapılmaktadır. Çalışanların ait olduğu iş sahasına göre geliştirilmiş hazır paket yazılımlar mevcuttur. Bu yazılımlar geliştirilmeye pek açık değildir. Böyle bir durumda çalışanın ihtiyaç olarak gördüğü eksiklikleri veya yeni ilaveleri, programlama dillerinden biri ile yazılım geliştirmeye başlayarak rahatlıkla çözmek mümkündür. Yazılım geliştirme aşamasında birçok durum göz önüne alınır. Eğer şu durum oluşmuşsa şunu yap gibi. Yapılacak adımların bilgisayara izah edilmesi gereklidir. Bu izahatı programlama dili gerçekleştirecektir. Yazılımı geliştiren bir çalışan hem zamanını çok daha iyi bir şekilde değerlendirmiş olacak hem de geliştirdiği yazılım sayesinde hatadan arındırılmış sonuçlara sahip bir sistemi kullanıyor olacaktır. Günümüz dünyasında bir yabancı dil öğrenme çılgınlığı halen devam etmektedir. Bireysel veya kurumsal mercilerin yabancı dil için harcadıkları meblağların programlama dillerini öğrenmeye ayrılması sonucunda ortaya çok gelişmiş yazılımlar çıkacaktır. İki de dil olarak anılan bu ihtiyaçların yer değişmesi artık zorunlu hale gelmiştir. Yabancı dil çok gerekmedikçe yerine programlama dili öğrenilmelidir. 2021 yılı en zenginler listesine bir göz atıldığında ilk onuncu sıradaki kişinin yedisinin yazılım teknoloji şirketlerinden olduğu görülmektedir.

Günümüz dünyasında yazılım geliştirici ihtiyacı gün geçtikçe artmaktadır. Yazılım geliştiriciliği, gelir getirisi yüksek olan bir meslek haline gelmiştir. Bu makalede yazılım geliştiriciliğe nasıl ve nereden başlanacağını ve hangi noktalara dikkat edilmesi gerektiği anlatılmıştır. İşletmelerde çalışanların da programlama dillerini öğrenmesine sevk ettirecek sebepler ortaya çıkartılmıştır.

#### KAYNAKÇA

- BASHIN, H. (2019). *Python Basics*. Mercury Learning and Information: U.S.A., Modern Language Assoc. 8. Sürüm
- LYASHKO, A. (2017). *Mastering Assembly Programming*. Packt Publishing: United Kingdom.
- OF, M. (2006). *Programlama Dilleri Ders Notları*. Kocaeli Meslek Yüksekokulu, Kocaeli.
- JAYDEN, K. (2016). *C# A Beginner's Tutorial*, <https://ebookcentral.proquest.com>
- URL 1: <http://www.goalkicker.com>, Stack Overflow Documentation, 2019
- URL 2: [http://www.robotiksistem.com/programlama\\_dilleri\\_ozellikleri.html](http://www.robotiksistem.com/programlama_dilleri_ozellikleri.html), E. Tarihi 08.10.2021
- URL 3: <http://www.csharpnedir.com/articles/read/?id=436>, E. Tarihi 08.10.2021
- URL 4: <http://etralibela-61.blogspot.com/2013/09/programlama-temelleri.html#.WZ6MaSILdPY>, E. Tarihi 04.10.2021
- URL 5: <https://docs.microsoft.com/tr-tr/dotnet/visual-basic/programming-guide/language-features/>, E. Tarihi 09.10.2021
- URL 6: <https://www.visualstudio.com/tr/>, E. Tarihi 09.10.2021
- URL 7: [https://tr.wikibooks.org/wiki/C\\_Sharp\\_Programlama\\_Dili/C\\_Sharp\\_hakk%C4%B1nda\\_temel\\_bilgiler](https://tr.wikibooks.org/wiki/C_Sharp_Programlama_Dili/C_Sharp_hakk%C4%B1nda_temel_bilgiler), E. Tarihi 08.10.2021