

# Yazılım Mühendisliğine (YM) Giriş

Ders Notu IV

2023 GÜZ

# 1. Yapısal (Procedural) Programlama Dilleri

- ❑ Yapısal bir dil, istenen çıktıyı elde etmek için bir dizi ifadeyi (expression) veya komutu (command) takip eder.
- ❑ Her adım dizisine prosedür (modül) adı verilir.
- ❑ Bu dillerden birinde yazılan bir programın içinde bir veya daha fazla prosedür (fonksiyon) bulunur.

C and C++

Java

Pascal

BASIC

## 2. Fonksiyonel (İşlevsel) Programlama Dilleri

- ❑ İşlevsel diller, ifadelerin yürütülmesine odaklanmak yerine, matematiksel işlevlerin ve değerlendirmelerin çıktısına odaklanır.
- ❑ Yeniden kullanılabilen bir kod modülü olan her işlev, belirli bir görevi gerçekleştirir ve bir sonuç döndürür.
- ❑ Sonuç, işleve hangi verileri girdiğinize bağlı olarak değişecektir.

**Scala (2004)** : Yüksek düzeyli genel amaçlı bir programlama dili. Hep fonksiyonel hem de nesneye yönelik.

**Erlang (1986)** Ericson tarafından yazılan genel amaçlı, eşzamanlı (concurrent) programlama dili . Telekomünikasyon sistemlerinde kullanılıyor.

**Haskell (1990)**

**Elixir (2012)** a dynamic, functional language for building scalable and maintainable applications.

**F# (2005)** **JavaScript** and **.NET** language for **web, cloud, data-science, apps** and more.

## 3. Nesneye Yönelik (Object Oriented) Programlama Dilleri

- ❑ Nesneye yönelik bir dil, bir programı, nitelikler (attributes) ve yöntemler (methods) olarak bilinen, veri ve program öğelerinden oluşan bir nesne grubu olarak incelenir.
- ❑ Nesneler bir program içinde veya başka programlarda yeniden kullanılabilir.
- ❑ Kodun yeniden kullanılması ve ölçeklendirilmesi daha kolaydır.
- ❑ Karmaşık programlar için popüler bir dil tür haline getirmiştir. .

Java (1995) James Gosling

Python (1991)

PHP (1995)

C++ (1985)

Ruby (1995)

## 4. Betik Diller (Scripting languages)

□ Programcılar, tekrarlanan görevleri (işleri) otomatikleştirmek, dinamik web içeriğini yönetmek veya daha büyük uygulamalardaki süreçleri desteklemek için komut dosyası dillerini, yani betik dillerini kullanır.

PHP

Ruby

Python

bash

Perl

Node.js (2009) cross-platform, open-source server environment. Run on Windows, Linux, Unix, macOS, ..... back-end JavaScript runtime environment. runs on the V8 JavaScript engine, and executes JavaScript code outside a web browser

## 5. Mantıksal (Logic) Programlama Dilleri

- ❑ Mantıksal programlama dili, bilgisayara nasıl karar vereceği konusunda talimat vermek için bir dizi olguyu ve kuralı ifade eder.
- ❑ Böylece bir bilgisayara neler yapması gerektiğini bildirmek gerekmez.

Prolog (1972)

Absys (1957) Declarative

Datalog (1977 ) Declarative

Lisp (1958) `((A B) (HELLO WORLD) 94)`

# Prolog Örneği

```
cat(tom).                % tom is a cat
mouse(jerry).           % jerry is a mouse

animal(X) :- cat(X).    % each cat is an animal
animal(X) :- mouse(X). % each mouse is an animal

big(X) :- cat(X).       % each cat is big
small(X) :- mouse(X).   % each mouse is small

eat(X,Y) :- mouse(X), cheese(Y). % each mouse eats each cheese
eat(X,Y) :- big(X), small(Y).    % each big being eats each small being
```

```
?- animal(X)
X= tom
```

```
?- cat(tom)
Yes
```

# Programlama Dillerinin Farklı Sınıflandırması -A

## Ön uç (front-end) programlama dilleri

- ❑ Ön uç diller öncelikle yazılımın 'kullanıcı' yönüyle ilgilidir.
- ❑ Ön uç, kullanıcının web sitenizde veya uygulamanızda gezinirken karşılaşacağı tüm metinler, renkler, düğmeler, resimler ve gezinmeyle ilgilidir.
- ❑ Grafik tasarım veya sanat alanında geçmişi olan herkes, ön uç dillerden birini öğrenmeye başlamak için daha fazla ilham alabilir.

HTML

CSS

JavaScript

React



# Programlama Dillerinin Farklı Sınıflandırması -B

## Arka uç (back-end) programlama dilleri

- Arka uç dilleri, yazılımın sunucu tarafının depolanması ve işlenmesiyle ilgilenir.
- Bu, yazılımın kullanıcının doğrudan temas etmediği ancak perde arkası deneyimini destekleyen kısımdır.
- Buna veri mimarisi, komut dosyası oluşturma ve uygulamalar ile temel veritabanları arasındaki iletişim dahildir
- Mühendislik deneyimi olan herkes arka uç geliştirmeye daha fazla uygun olacaktır.

JavaScript

PHP

Java

Python

Ruby

C#

## Bir başka Dil Sınıflandırması:

### Yüksek Düzeyli (High-level) ve Düşük Düzeyli (Low-level) Diller

- ❑ Yüksek ve düşük seviyeli programlama dillerinin birbirinden en önemli farkı, dilin bir programcı tarafından mı yoksa bir bilgisayar tarafından mı kolaylıkla anlaşılabiliridir.
- ❑ Düşük seviyeli diller makine dostudur
  - ❖ Bu diller bellek kullanımı açısından oldukça verimlidir, ancak bir çevirici (translator) yardımı olmadan anlaşılması zordur.
  - ❖ Konuşma dilini kullanmadıkları için kullanıcı dostu değildir ve kodlama amaçlı kullanılmaz.
- ❑ Düşük düzeyli dillere örnek olarak makine kodu, ikili kod ve assembly dilleri verilebilir.
- ❑ Yüksek seviyeli diller bellek açısından daha az verimlidir, ancak kullanıcı dostudur.
- ❑ Yüksek düzeyli programlama dillerinin yazılması, anlaşılması, bakımı (sürekliliği) ve hata ayıklaması daha kolaydır.
  - ❖ Günümüzde kullanılan en popüler programlama dilleri üst düzey dillerdir.

# Farklı bir Sınıflandırma: Yorumlanabilen (Interpreted) ve Derlenebilen (Compiled) Diller

- ❑ Yorumlanan ve derlenen diller arasındaki fark, üst düzey kodu nasıl dönüştürdükleri ve bilgisayar tarafından okunabilir hale getirdikleri ile ilgilidir.
- ❑ **Yorumlanmış dillerde** kod, yorumlayıcı adı verilen ve kodu satır satır okuyan ve çalıştıran bir programdan geçer.
  - ❖ Bu özellik ile yorumlanabilir dilleri daha esnek ve platformdan bağımsız hale getirme eğilimindedir.

Örnekleri:

Python

JavaScript

PHP

Ruby

- ❑ **Derlenmiş diller**, tüm programın makine koduna dönüştürüldüğü bir adımından geçer.

- ❑ Bu yapı, programın yürütülmesini hızlandırır, ancak aynı zamanda değişiklik yapıldığında gerektiğinde programın yeniden derlenmesi gerekir. Örnekleri:

C, C++, and C#

Rust

Erlang

# Markup language – İşaretleme Dili

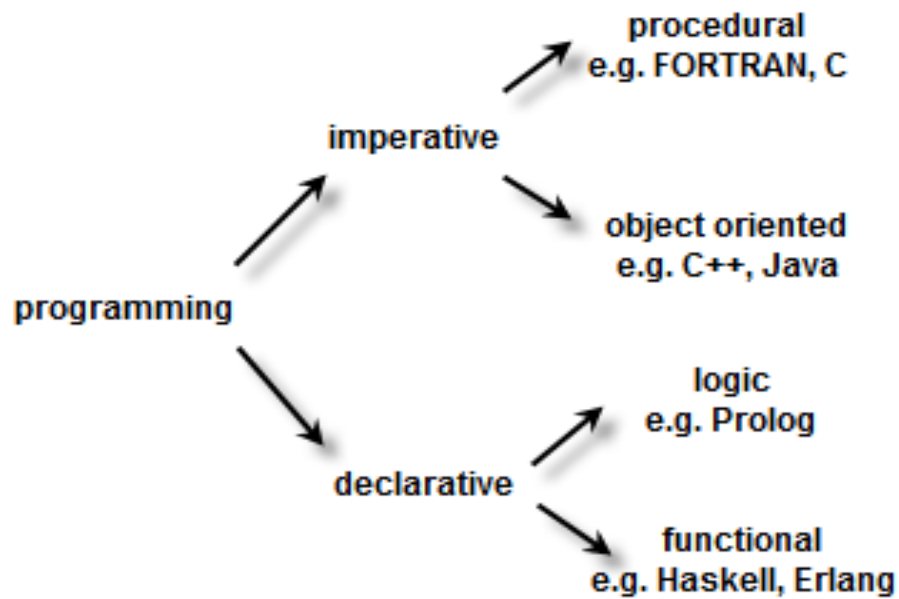
- ❑ Hypertext Markup Language (HTML) gibi biçimlendirme (işaretleme) dillerinin programlama dilleri olarak sınıflandırıldığı yaygındır.
- ❑ Ama işaretleme dilleri programlama dilleriyle aynı kabul edilemez.
  - ❖ Bu diller, bir sayfa veya belgedeki içeriğin formatını ve yapısını kontrol eden sembollerden oluşan metin kodlama sistemleridir.

```
webpage - Notepad
File Edit Format View Help

<!doctype html>
<html>
  <head>
    <title>Document title</title>
  </head>
  <body style="background-color:black;">
    <center>
      
      <br>
      <a href="https://www.mywebsite.com/home">
      <a href="https://www.mywebsite.com/page2">
    </center>
    <br>
    <h1 style="color:white;">About Us</ht>
    <br>
    <p style='color:white;'>A little about us...</p>
    <hr>
  </body>
</html>
```

Windows (CRLF) Ln 3, Col 1 100%

# Programlama Paradigmaları



**Imperative Programming:** **how** you do something

Statements yani koda ait her kod satırı hesaplanan durumu (veri alanını) değiştirir.  
C, C++, Java, PHP, Python, Ruby

**Declarative Programming:** **what** you do.

Programın mantığı tanımlanır, fakat ayrıntılı kontrol akışı içermez.

Prolog (Programming in Logic)

Pek çok «markup languages» , yani HTML, MXML, XAML, XSLT ya da diğer kullanıcı arayüzü işaretleme dilleri

# Imperative Paradigm & Declarative Paradigm

Procedural ve Object-Oriented Programming → **Imperative Paradigm**

Örneğin; C, C++, C#, PHP, Java , Assembly.

Logical Programming, Functional Programming, regular expressions, Database Queries → **Declarative Paradigm**

Örneğin; Prolog ,Javascript (multiparadigmatic) ,Scala, Lisp, SQL ,XQuery , Clojure

```
SELECT * FROM Books
  WHERE Category='Programming'
  AND language='english';
```

bir SQL sorgusudur.