Introduction to Programming Eğitimi

Süre: **2 Gün**

BİLGİNÇ
IT Academy
info@bilginc.com
bilginc.com

Introduction to Programming eğitimi, programlamaya yeni başlayanlar için tasarlanmış kapsamlı bir başlangıç seviyesindeki **Programlama Eğitimi**dir.

Bu eğitimde katılımcılar, modern yazılım geliştirmenin temelini oluşturan **Programming Fundamentals**, **Programming Logic**, **Variables**, **Data Types**, **Operators**, **Functions**, **Loops**, **Conditionals** ve **Debugging Techniques** konularını öğrenirler.

İster yazılım geliştirme kariyerine başlamak, ister günlük görevleri otomatikleştirmek ya da teknik becerilerini geliştirmek isteyenler için güçlü bir temel sunar.

Önkoşullar

Bu eğitime katılmak için herhangi bir programlama bilgisi gerekmemektedir.

Aşağıdaki yetkinlikler faydalı olabilir:

- Temel matematik bilgisi
- Mantıksal düşünme becerileri
- Problem çözme yaklaşımı

Kimler Katılmalı

Bu eğitim aşağıdaki kişiler için uygundur:

- Programlama öğrenmek isteyenler
- Yazılım geliştirmeye yeni başlayanlar
- Teknik kariyer hedefleyen öğrenciler
- İş süreçlerini otomatikleştirmek isteyen profesyoneller
- **Coding for Beginners** seviyesinde eğitim arayanlar
- Bilgisayar programlamanın temel prensiplerini öğrenmek isteyen herkes

Introduction to Programming Eğitimi

Süre: **2 Gün**

BİLGİNÇ IT Academy
info@bilginc.com
bilginc.com

Neler Öğreneceksiniz

Bu eğitim sonunda katılımcılar:

- **Programming Fundamentals** kavramlarını anlayabilecek,
- Programlamanın gerçek hayattaki kullanım alanlarını açıklayabilecek,
- **Compiled** ve **Interpreted Programming Languages** arasındaki farkları bilecek,
- Basit programlar geliştirebilecek,
- **Variables**, **Data Types** ve **Operators** kullanabilecek,
- Karar yapıları ve döngüler oluşturabilecek,
- Fonksiyonlar ile kod organizasyonu sağlayabilecek,
- Temel **Debugging Techniques** uygulayabilecek,
- Daha ileri seviyedeki yazılım geliştirme eğitimlerine hazır hale gelecektir.

Eğitim İçeriği

1. Programlamaya Giriş

Introduction to Programming

- Programlama nedir?
- Programlama neden önemlidir?
- Gerçek dünya problemlerinin çözümü
- Otomasyon kavramı
- Programlamanın tarihçesi
- Yazılım geliştirme süreçleri

Öğrenilecek Konular

- Problem çözme yaklaşımı
- Algoritmik düşünme
- Programlama mantığı

2. Programlama Dilleri ve Geliştirme Ortamı

Programming Languages & Development Environment

- Programlama dilleri nedir?
- Programlama dillerinin amacı
- Programlama paradigmleri

Programlama Dili Türleri

- Procedural Programming
- Object-Oriented Programming
- Functional Programming
- Scripting Languages

Popüler Programlama Dilleri

- **Python**
- **JavaScript**
- **Java**
- **C++**
- **C#**

Compiled vs Interpreted Languages

- Derlenen diller
- Yorumlanan diller
- Kullanım senaryoları

Geliştirme Ortamı

- Visual Studio Code (VS Code)
- Online Code Editors
- Runtime ve Interpreter kavramları
- Kod dosyalarının oluşturulması
- Terminal kullanımı

Uygulama

- İlk "Hello World" programı

3. Kodun Yapısı

Structure of Code

- Syntax ve Semantics
- Kod kuralları
- Girintileme (Indentation)
- Kod yorumları
- Pseudocode kavramı

Öğrenilecek Kazanımlar

- Temiz kod yazımı
- Kod okunabilirliği
- Program organizasyonu

4. Veri Tipleri ve Değişkenler

Data Types & Variables

Veri Tipleri

- Numbers
- Strings
- Booleans

Veri Yapıları

- Arrays

BİLGİNÇ IT Academy

info@bilginc.com

bilginc.com

- Lists
- Objects

Değişkenler

- Variable Declaration
- Variable Initialization
- Variable Scope

Best Practices

- İsimlendirme kuralları
- Kod standartları
- Veri yönetimi

5. Operatörler ve İfadeler

Operators & Expressions

Arithmetic Operators

- Toplama
- Çıkarma
- Çarpma
- Bölme
- Modulus

Comparison Operators

- Eşitlik
- Büyük/Küçük karşılaştırmaları

Logical Operators

- AND
- OR
- NOT

Öğrenilecek Kazanımlar

- Hesaplama işlemleri
- Mantıksal değerlendirmeler
- Program akış kontrolü

6. Karar Yapıları

Control Flow (Conditionals)

If / Else Yapıları

- if
- else if
- else

Switch / Case

- Çoklu durum yönetimi
- Karar mekanizmaları

Uygulamalar

- Kullanıcı girişlerinin değerlendirilmesi
- İş kurallarının uygulanması

7. Fonksiyonlar

Functions

Fonksiyonlar kodun tekrar kullanılabilir ve yönetilebilir hale gelmesini sağlar.

Konular

- Function Declaration
- Parameters
- Arguments
- Return Values
- Scope
- Local ve Global Variables

Kazanımlar

- Modüler programlama
- Kod organizasyonu
- Yeniden kullanılabilir kod geliştirme

8. Döngüler ve Tekrarlama

Loops & Iteration

Loop Türleri

- While Loop
- Do-While Loop
- For Loop
- Reverse Loop
- Nested Loop

Loop Control

- Break
- Continue
- Infinite Loops

Kullanım Alanları

- Tekrarlayan işlemler

BİLGİNÇ IT Academy

info@bilginc.com

bilginc.com

- Veri işleme
- Otomasyon süreçleri

9. Temel Veri Yapıları

Basic Data Structures

Arrays ve Lists

- Veri saklama
- İndeksleme
- Veri erişimi

Kullanım Amaçları

- Veri organizasyonu
- Koleksiyon yönetimi
- Performans optimizasyonu

10. Hata Ayıklama

Debugging Techniques

Hata Türleri

- Syntax Errors
- Runtime Errors
- Logic Errors

Hata Çözme Teknikleri

- Log kullanımı
- Print Statements
- Hata mesajlarının okunması
- Adım adım analiz

Kazanımlar

- Daha güvenilir kod yazımı
- Problemleri hızlı çözebilme
- Kod kalitesini artırma