

## Conteúdo Programático

# Programação avançada em C#

*.NET framework e IDE Microsoft Visual Studio*



IN-CLASS



35 HORAS

### Destinatários

Todas as pessoas com interesse em adquirir um nível de competências avançadas para programar em C#.

### Nº mínimo de inscritos

6

### Pré-requisitos

Conhecimentos básicos de sintaxe e semântica em C# e de programação orientada a objetos. Ou frequência da formação “Introdução à Programação em C#”.

### Certificação de técnicos:

[Clique aqui](#) para consultar os critérios de atribuição de créditos

### Contactos

Angola/ (+244) 222 440 447  
academy\_ao@primaverabss.com

Cabo Verde/(+238) 356 37 73  
academy@primaverabss.com

Moçambique/(+258) 21 303 388  
academy\_mz@primaverabss.com

Portugal/(+351) 253 309 241  
academy@primaverabss.com

### Enquadramento

O mercado empresarial necessita de profissionais que possuam competências técnicas nas mais variadas tecnologias. A programação orientada a objetos é, nos dias de hoje, uma competência essencial para um programador.

C# é uma linguagem de programação Orientada a Objetos e foi desenvolvida pela Microsoft como uma forte aposta face a outras linguagens existentes no mercado, fazendo atualmente parte da sua plataforma .NET *Framework*.

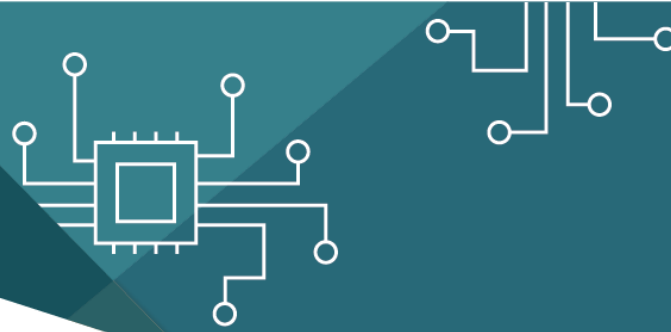
A formação intensiva de “Programação avançada em C#”, pretende formar programadores de excelência com todas as competências necessárias para o exercício de funções numa situação real de trabalho.

Acreditamos que um bom programador não se pode limitar a aprender a sintaxe da linguagem e a desenvolver o raciocínio lógico, necessita ainda de escrever código limpo e utilizar princípios de boas práticas de programação (*SOLID Principles*), melhorando a legibilidade e qualidade do código desenvolvido.

### Objetivos

No final da formação os formandos deverão ser capazes de:

- Utilizar a .NET *framework* e o ambiente de desenvolvimento Microsoft *Visual Studio*;
- Usar a sintaxe, semântica e características da linguagem C#, na produção de código;
- Programar código limpo e utilizar os princípios SOLID;
- Criar testes unitários com Xunit. Introdução ao TDD (*Test Driven Design*);
- Tratar exceções da linguagem C#;
- Usar programação assíncrona com *async* e *await*;
- Utilizar LINQ (*Language integrated Query*) para consulta e manipulação de dados;
- Aprender conteúdos avançados, como a utilização de expressões *Lambda*, *events* e *delegates*, *Tuples*, *Pattern matching*, funções anónimas, funções locais, genéricos ou *Extension Methods*.



## Conteúdo Programático

# Programação avançada em C#

*.NET framework e IDE Microsoft Visual Studio*



IN-CLASS



35 HORAS

### Plano de Formação (resumido)

Módulo 1: POO

Módulo 2: Coleções

Módulo 3: Princípios SOLID e código limpo

Módulo 4: Introdução ao desenvolvimento orientado a testes

Módulo 5: LINQ (*Language integrated Query*)

Módulo 6: Conteúdos avançados

Módulo 7: Funcionalidades recentes

### Plano de Formação (detalhado)

#### Módulo 1: POO

Neste módulo são revistos os conceitos fundamentais e os 4 pilares da Programação Orientada a Objetos – Abstração, Encapsulamento, Herança e Polimorfismo. Serão ainda abordados os Interfaces e a sua reimplementação em subclasses, fazendo-se um paralelo entre estes e a herança de classes de modo a entender as diferenças e tirar o melhor partido de ambas as abordagens.

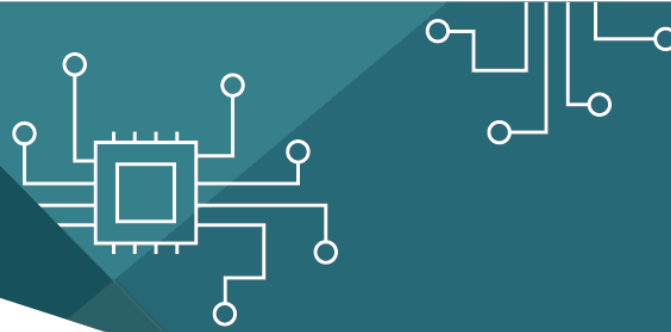
#### Sessão

Programação Orientada a Objetos

- Classes
- Classes abstratas
- Classes e métodos estáticos
- Classes e métodos parciais
- Construtores de instância e estáticos
- Herança
- Encapsulamento
- Polimorfismo
- Interfaces

Interfaces

- Extensão de interfaces
- Implementação explícita de interfaces
- Reimplementação de interfaces numa sub-classe



## Conteúdo Programático

# Programação avançada em C#

*.NET framework e IDE Microsoft Visual Studio*



IN-CLASS



35 HORAS

### Módulo 2: Coleções

Neste módulo são abordadas as coleções para armazenamento de objetos e as funcionalidades que permitem a gestão e manipulação desses mesmos objetos. Veremos ainda como a implementação de determinados interfaces possibilitam e facilitam o uso padronizado de métodos e estruturas de iteração (e.g. *foreach*).

#### Sessão

Interfaces IEnumerable e IEnumerator

Interfaces ICollection e IList

Arrays

Listas, Pilhas, Filas

Dicionários

### Módulo 3: Princípios SOLID e código limpo

Neste módulo são apresentados os princípios SOLID, permitindo a aquisição de bons hábitos no desenvolvimento de código com qualidade, limpo e extensível.

#### Sessão

Princípios SOLID

- Single Responsibility Principle
- Open Close Principle
- Liskov Principle
- Interface Segregation Principle
- Dependency Inversion Principle (IoC containers)

Código limpo

- Organização de código
- Comentários

### Módulo 4: Introdução ao desenvolvimento orientado a testes

Neste módulo são abordados os conceitos de desenvolvimento orientado a testes, possibilitando a criação de código de alta qualidade. Dar-se-á ainda destaque à importância dos testes no *refactoring* de código e extensão de funcionalidades usando-se a package xUnit como ferramenta de validação de testes.

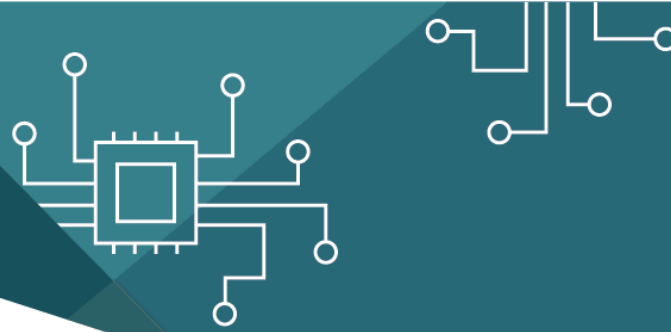
#### Sessão

Introdução aos testes unitários

Padrão de escrita de testes AAA (Arrange, Act, Assert)

Conceitos de Desenvolvimento Orientado a Testes (TDD – Test Driven Development)

Validação de testes com XUnit



## Conteúdo Programático

# Programação avançada em C#

*.NET framework e IDE Microsoft Visual Studio*



IN-CLASS



35 HORAS

### **Módulo 5: LINQ (*Language integrated Query*)**

Neste módulo vai aprender a escrever consultas “*type safe*” estruturadas sobre coleções de objetos.

#### Sessão

Sintaxe  
Operadores  
Expressões de consulta  
Execução diferida  
Subconsultas  
Estratégias de composição e projeção

### **Módulo 6: Conteúdos avançados**

Neste módulo são apresentados os conceitos avançados mais importantes da programação em C#.

#### Sessão

Delegates  
Events  
Expressões *Lambda*  
Tratamento de exceções  
Extension Methods  
Tipos nulos  
Programação assíncrona  
Genéricos

### **Módulo 7: Funcionalidades recentes**

C# é uma linguagem que está em permanente mudança através de funcionalidades que vão sendo incorporadas de forma ágil pela Microsoft. Neste módulo, são apresentadas as funcionalidades mais recentes da linguagem e que ainda não foram abordadas nos módulos anteriores.

#### Sessão

*Tuples*  
*Pattern matching*  
Funções anónimas  
Funções locais  
Discards