

# Java - Orientação a objetos

Carga horária: 22 horas

TreinaWeb Tecnologia LTDA  
CNPJ: 06.156.637/0001-58  
Av. Paulista, 1765 - Conj 71 e 72  
São Paulo - SP

## CONTEÚDO PROGRAMÁTICO

### 1 - Introdução

▶ Apresentação

Duração: 00:01:11

📄 Introdução

### 2 - Relembrando o que é a orientação a objetos

▶ Apresentação

Duração: 00:00:29

📄 Relembrando o que é a orientação a objetos

📄 Como surgiu o conceito da orientação a objetos?

📄 Mas ainda não estou convencido. Por que a orientação a objetos?

📄 Mais detalhes do paradigma orientado a objetos

📄 Diagramas UML e o diagrama de classes

📄 O código-fonte deste curso está no GitHub!

### 3 - Classes e objetos

▶ Apresentação	Duração: 00:00:19
📄 Classes e objetos	
📄 Mas o que seriam classes?	
📄 E o que seriam os objetos?	
📄 Representação de classes na UML	
▶ Relembrando o conceito de classe e sua representação na UML	Duração: 00:21:18
▶ Criando nosso primeiro projeto e definindo a primeira classe	Duração: 00:08:32
▶ Definindo atributos para uma classe	Duração: 00:05:00
▶ Criando os primeiros métodos para uma classe	Duração: 00:05:34
▶ Entendendo e utilizando objetos	Duração: 00:11:41
📦 Questionário	3 questões
📦 Desafio de Código	
📦 Desafio de Código	

## 4 - Encapsulamento










▶ Apresentação	Duração: 00:00:29
📖 Encapsulamento	
📖 Por que devo me preocupar com encapsulamento?	
📖 Atributos de visibilidade	
📖 Métodos acessores	
📖 Encapsulamento, atributos de visibilidade e UML	
▶ Visibilidade private, métodos acessores e o padrão POJO	Duração: 00:17:51
▶ Representação dos métodos acessores na UML	Duração: 00:02:24
▶ Utilizando o encapsulamento de maneira correta e prática	Duração: 00:15:01
▶ Representações que fogem ao padrão da UML	Duração: 00:04:32
▶ Utilizando estereótipos para representar os métodos acessores na UML	Duração: 00:05:36
▶ Entendendo o significado do this	Duração: 00:08:29
▶ Criando setters mais inteligentes	Duração: 00:18:36
📦 Questionário	4 questões
📦 Desafio de Código	
📦 Desafio de Código	

## 5 - Herança

▶ Apresentação	Duração: 00:00:30
📄 Herança	
📄 Superclasses e subclasses	
📄 Quando devo, de fato, utilizar a herança?	
📄 Por que a herança constitui um assunto tão polêmico?	
📄 O atributo de visibilidade “protected”	
📄 Herança e UML	
📄 Herança simples ou herança múltipla?	
▶ Aplicando a herança em exceções	Duração: 00:07:05
▶ Java e a visibilidade a nível de pacote	Duração: 00:07:36
▶ Criando construtores personalizados	Duração: 00:13:07
▶ Entendendo o processo de serialização e deserialização e o SERIAL VERSION ID	Duração: 00:09:32
▶ Um pouco mais de herança	Duração: 00:04:46
▶ Entendendo a visibilidade protected	Duração: 00:09:22
▶ Entendendo os conceitos de sobrecarga e assinatura	Duração: 00:11:01
▶ Herança múltipla vs herança simples	Duração: 00:03:52
▶ Encadeando construtores	Duração: 00:10:57
▶ Definindo atributos específicos para as subclasses	Duração: 00:04:22
▶ Liskov Substitution Principle e o polimorfismo	Duração: 00:28:12
▶ Sobrescrevendo métodos	Duração: 00:22:15
▶ A palavra-chave final	Duração: 00:09:27
▶ Definindo classes abstratas	Duração: 00:04:46
▶ Definindo métodos abstratos	Duração: 00:11:45
▶ Herança e as representações na UML	Duração: 00:21:27
📦 Questionário	7 questões
📦 Desafio de Código	
📦 Desafio de Código	
📦 Desafio de Código	
📦 Desafio de Código	

 Desafio de Código

## 6 - Interfaces

 Apresentação	Duração: 00:00:26
 Interfaces	
 O que são interfaces?	
 Interfaces e UML	
 Entendendo e utilizando interfaces - Parte 1	Duração: 00:13:34
 Entendendo e utilizando interfaces - Parte 2	Duração: 00:23:13
 Questionário	2 questões
 Desafio de Código	
 Desafio de Código	

## 7 - Conclusão

 Conclusão

Ficou alguma dúvida em relação ao conteúdo programático?  
Envie-nos um e-mail [clikando aqui](#) .