

## Aula 121

# Curso C# Completo Programação Orientada a Objetos + Projetos

**Capítulo: Enumerações, composição**

<http://educandoweb.com.br>

Prof. Dr. Nelio Alves

## Aula 123

# Enumerações (enum)

<http://educandoweb.com.br>

Prof. Dr. Nelio Alves

## Checklist

- Definição / discussão
- Exemplo: estados de um pedido
- Conversão de string para enum
- Representação UML

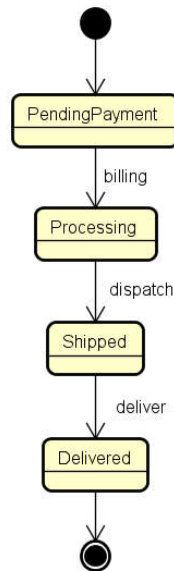
## Enumerações

- É um tipo especial que serve para especificar de forma literal um conjunto de constantes relacionadas
- Palavra chave em C#: `enum`
  - Nota: enum é um tipo valor
- Vantagem: melhor semântica, código mais legível e auxiliado pelo compilador
- Referência: <https://docs.microsoft.com/en-us/dotnet/csharp/language-reference/keywords/enum>

## Exemplo

Ciclo de vida de um pedido.

Diagrama de máquina de estados



1. `OrderStatus os = Enum.Parse<OrderStatus>("Delivered");`
2. `OrderStatus os = (OrderStatus)Enum.Parse(typeof(OrderStatus), "Delivered");`
3. `OrderStatus os;`
4. `Enum.TryParse("Delivered", out os);`

<https://github.com/acenelio/enum1-csharp>

```
enum OrderStatus : int {  
    PendingPayment = 0,  
    Processing = 1,  
    Shipped = 2,  
    Delivered = 3  
}  
  
class Order {  
    public int Id { get; set; }  
    public DateTime Moment { get; set; }  
    public OrderStatus Status { get; set; }  
}
```

Na aula fazemos a conversão de string para enumeração assim:

```
1 | OrderStatus os = Enum.Parse<OrderStatus>("Delivered");
```

Mas, dependendo da versão do compilador C#, isso pode dar problema. Se for o caso, por favor tente deste jeito:

```
1 | OrderStatus os = (OrderStatus)Enum.Parse(typeof(OrderStatus),  
    "Delivered");
```

Se mesmo assim ainda tiver dando erro, há ainda uma terceira forma:

```
1 | OrderStatus os;  
2 | Enum.TryParse("Delivered", out os);
```

## Conversões

```
Order order = new Order
{
    Id = 1080,
    Moment = DateTime.Now,
    Status = OrderStatus.PendingPayment
};
Console.WriteLine(order);
```

### Para o tipo original: use casting

```
OrderStatus os1 = (OrderStatus)2;
int n1 = (int)OrderStatus.Processing;

Console.WriteLine(os1);
Console.WriteLine(n1);
```

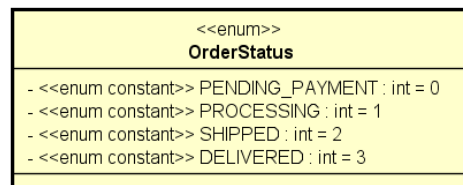
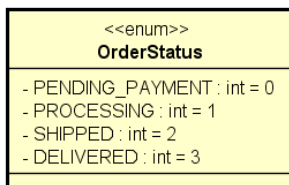
### string - enum:

```
string txt = OrderStatus.PendingPayment.ToString();
OrderStatus os = Enum.Parse<OrderStatus>("Delivered");

Console.WriteLine(os);
Console.WriteLine(txt);
```

Aula 124 @ final

## Notação UML



# Vamos falar um pouco de design

<http://educandoweb.com.br>

Prof. Dr. Nelio Alves

## Categorias de classes

- Em um sistema orientado a objetos, de modo geral "tudo" é objeto.
- Por questões de design tais como organização, flexibilidade, reuso, delegação, etc., há várias categorias de classes:

Views

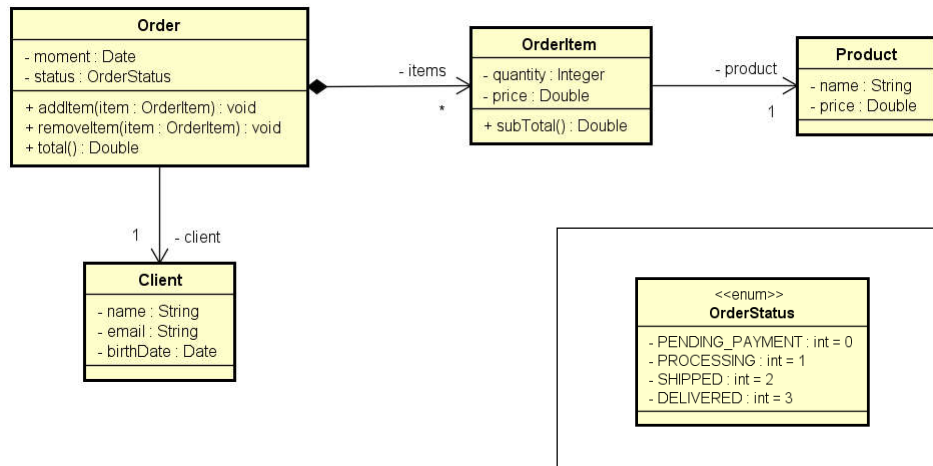
Controllers

Entities

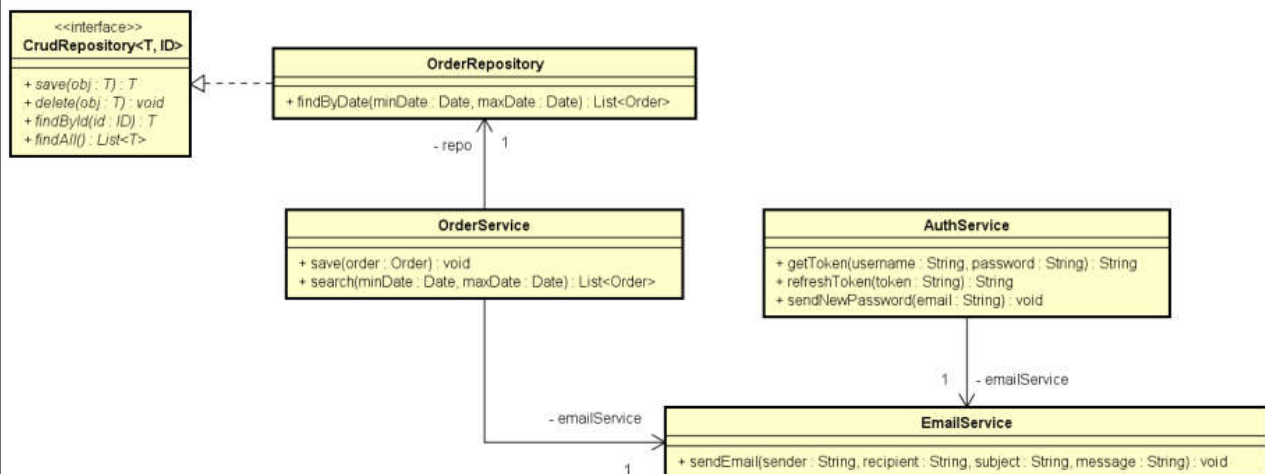
Services

Repositories

# Entities



# Services



# Composição

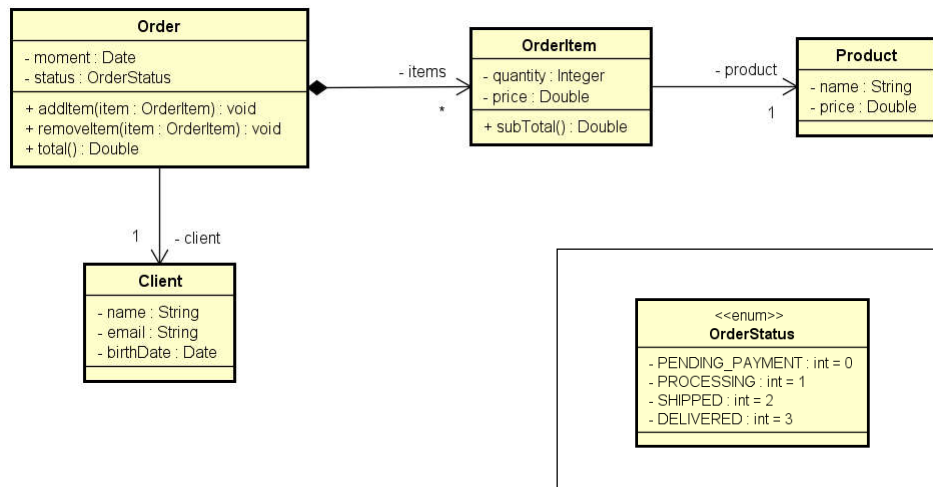
<http://educandoweb.com.br>

Prof. Dr. Nelio Alves

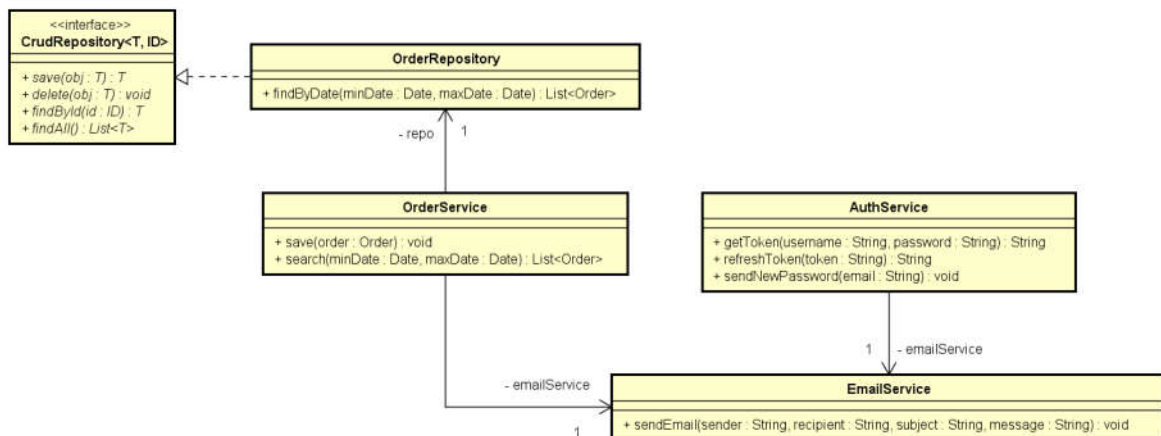
## Composição

- É um tipo de associação que permite que um objeto contenha outro
- Relação "tem-um" ou "tem-vários"
- Vantagens
  - Organização: divisão de responsabilidades
  - Coesão
  - Flexibilidade
  - Reuso
- Nota: embora o símbolo UML para composição (todo-parte) seja o diamante preto, neste contexto estamos chamando de composição qualquer associação tipo "tem-um" e "tem-muitos".

# Entities



# Services





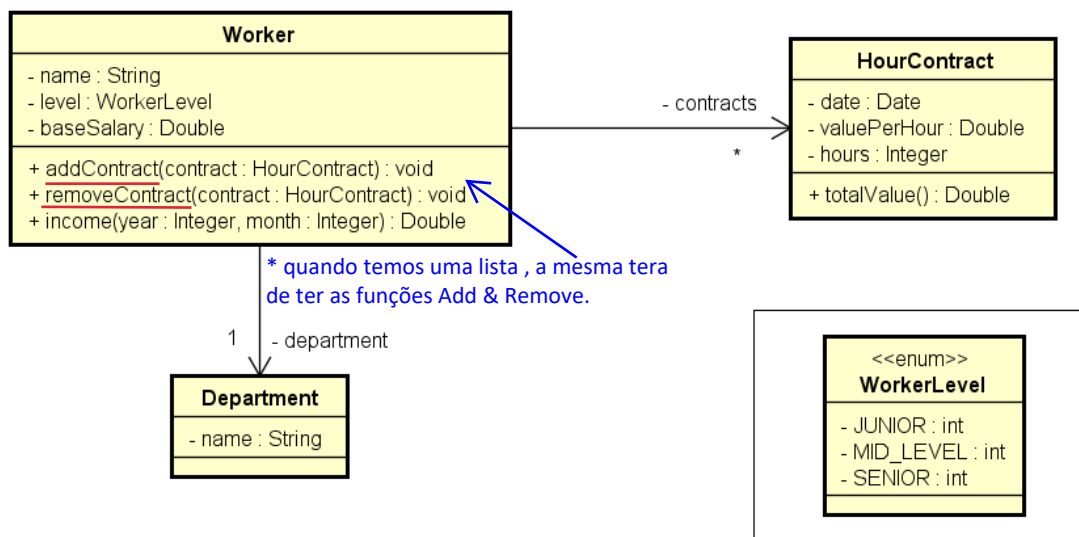
# Exercício resolvido 1

<http://educandoweb.com.br>

Prof. Dr. Nelio Alves

1. `OrderStatus os = Enum.Parse<OrderStatus>("Delivered");`
2. `OrderStatus os = (OrderStatus)Enum.Parse(typeof(OrderStatus), "Delivered");`
3. `OrderStatus os;`
4. `Enum.TryParse("Delivered", out os);`

Ler os dados de um trabalhador com N contratos (N fornecido pelo usuário). Depois, solicitar do usuário um mês e mostrar qual foi o salário do funcionário nesse mês, conforme exemplo (próxima página).

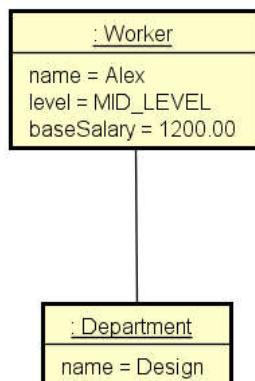


```
Enter department's name: Design
Enter worker data:
Name: Alex
Level (Junior/MidLevel/Senior): MidLevel
Base salary: 1200.00
How many contracts to this worker? 3
Enter #1 contract data:
Date (DD/MM/YYYY): 20/08/2018
Value per hour: 50.00
Duration (hours): 20
Enter #2 contract data:
Date (DD/MM/YYYY): 13/06/2018
Value per hour: 30.00
Duration (hours): 18
Enter #3 contract data:
Date (DD/MM/YYYY): 25/08/2018
Value per hour: 80.00
Duration (hours): 10

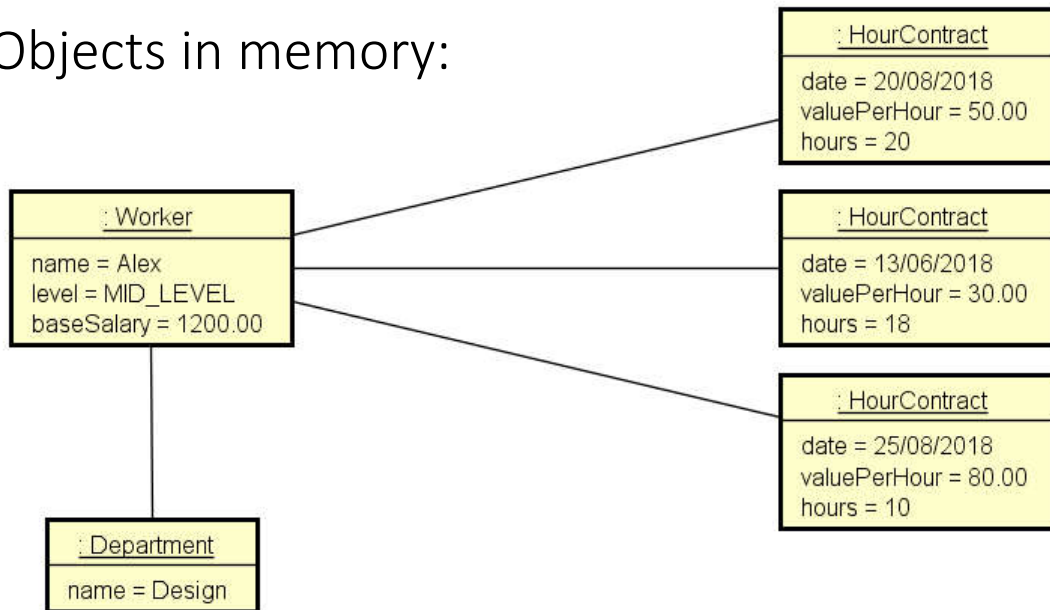
Enter month and year to calculate income (MM/YYYY): 08/2018
Name: Alex
Department: Design
Income for 08/2018: 3000.00
```

<https://github.com/acenelio/composition1-csharp>

## Objects in memory:



Objects in memory:



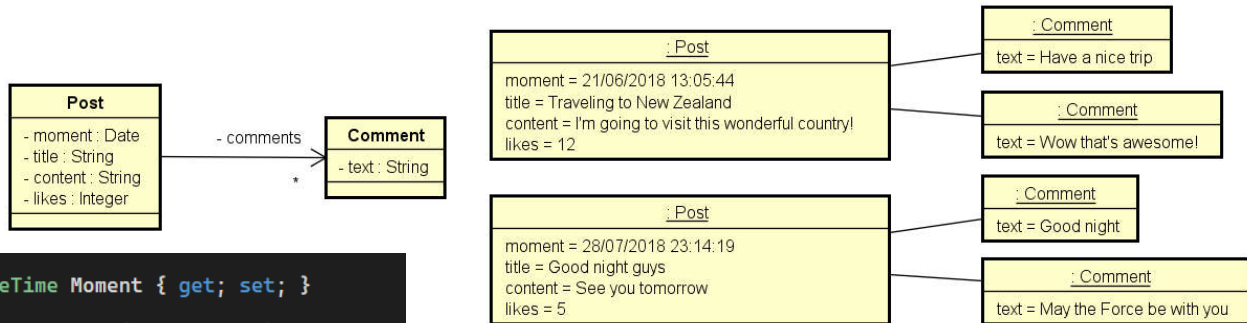
## Aula 131

# Exercício resolvido 2 (demo StringBuilder)

<http://educandoweb.com.br>

Prof. Dr. Nelio Alves

Instancie manualmente os objetos mostrados abaixo e mostre-os na tela do terminal, conforme exemplo.



1. As listas são declaradas e instanciadas nas properties para que não apresente o retorno de null

```
2 references
public DateTime Moment { get; set; }
2 references
public string Title { get; set; }
2 references
public string Content { get; set; }
2 references
public int Likes { get; set; }
3 references
public List<Comment> Comments { get; set; }
    = new List<Comment>();
2 references
public Post(DateTime moment, string title, string content, int likes)
{
    Moment = moment;
    Title = title;
    Content = content;
    Likes = likes;
}
```

2. As listas nunca são carregadas pelo construtor

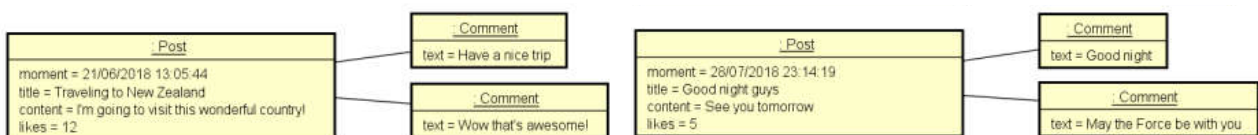
### StringBuilder:

```
0 references
public override string ToString()
{
    StringBuilder sb = new StringBuilder();
    sb.AppendLine(Title);
    sb.AppendLine(Likes);
    sb.AppendLine(" Likes - ");
    sb.AppendLine(Moment.ToString("dd/MM/yyyy HH:mm:ss"));
    sb.AppendLine(Content);
    sb.AppendLine("Comments:");
    foreach (Comment c in Comments)
    {
        sb.AppendLine(c.Text);
    }
    return sb.ToString();
}
```

### Console output:

```
Traveling to New Zealand
12 Likes - 21/06/2018 13:05:44
I'm going to visit this wonderful country!
Comments:
Have a nice trip
Wow that's awesome!

Good night guys
5 Likes - 28/07/2018 23:14:19
See you tomorrow
Comments:
Good night
May the Force be with you
```



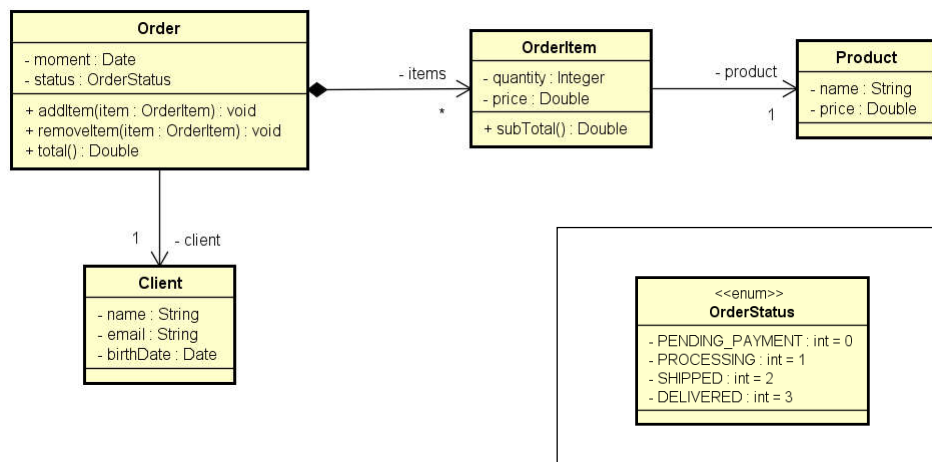
<https://github.com/acenelio/composition2-csharp>

# Exercício de fixação

<http://educandoweb.com.br>

Prof. Dr. Nelio Alves

Ler os dados de um pedido com N itens (N fornecido pelo usuário). Depois, mostrar um sumário do pedido conforme exemplo (próxima página). Nota: o instante do pedido deve ser o instante do sistema: `DateTime.Now`



**Enter cliente data:**

Name: **Alex Green**

Email: **alex@gmail.com**

Birth date (DD/MM/YYYY): **15/03/1985**

**Enter order data:**

Status: **Processing**

How many items to this order? **2**

**Enter #1 item data:**

Product name: **TV**

Product price: **1000.00**

Quantity: **1**

**Enter #2 item data:**

Product name: **Mouse**

Product price: **40.00**

Quantity: **2**

**ORDER SUMMARY:**

Order moment: 20/04/2018 11:25:09

Order status: Processing

Client: Alex Green (15/03/1985) - alex@gmail.com

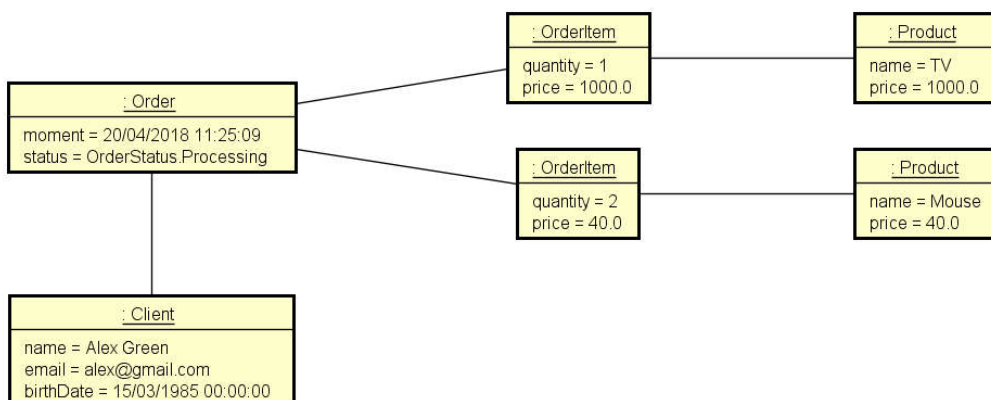
**Order items:**

TV, \$1000.00, Quantity: 1, Subtotal: \$1000.00

Mouse, \$40.00, Quantity: 2, Subtotal: \$80.00

Total price: \$1080.00

Você deverá instanciar os objetos em memória da seguinte forma:



<https://github.com/acenelio/composition3-csharp>