Orientação a Objetos 1

Enumerações

Prof. Dr. Vinícius Camargo Andrade

vcandrade@utfpr.edu.br

Departamento Acadêmico de Informática
Universidade Tecnológica Federal do Paraná

Enumerações

Enumerações

Enumeração é um tipo de dado abstrato, cujos valores são atribuídos a exatamente um elemento de um conjunto finito de identificadores escolhidos pelo programador.

Enumerações

Esse tipo é geralmente usado para variáveis categóricas, como por exemplo, meses do ano, dias da semana, naipes do baralho, entre outros.

Enumerações

Como vantagens, se tem uma melhor semântica e código mais legível

Enumerações

A enumerações em java são definidas pela palavra-chave enum.

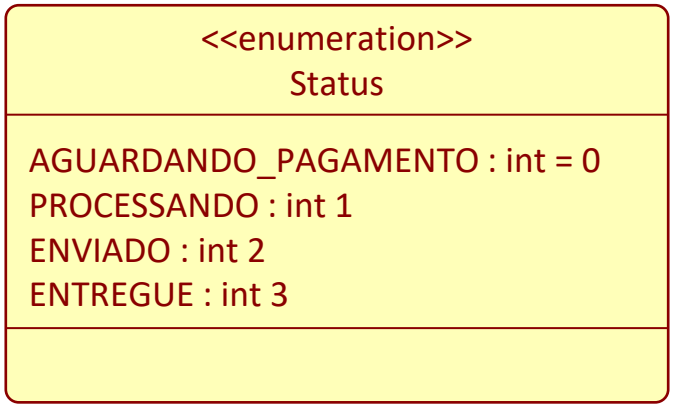
Exemplo 1

Exemplo 1

Desenvolva uma aplicação para gerar e controlar pedidos. Um pedido possui um id, data que foi gerado e um status.

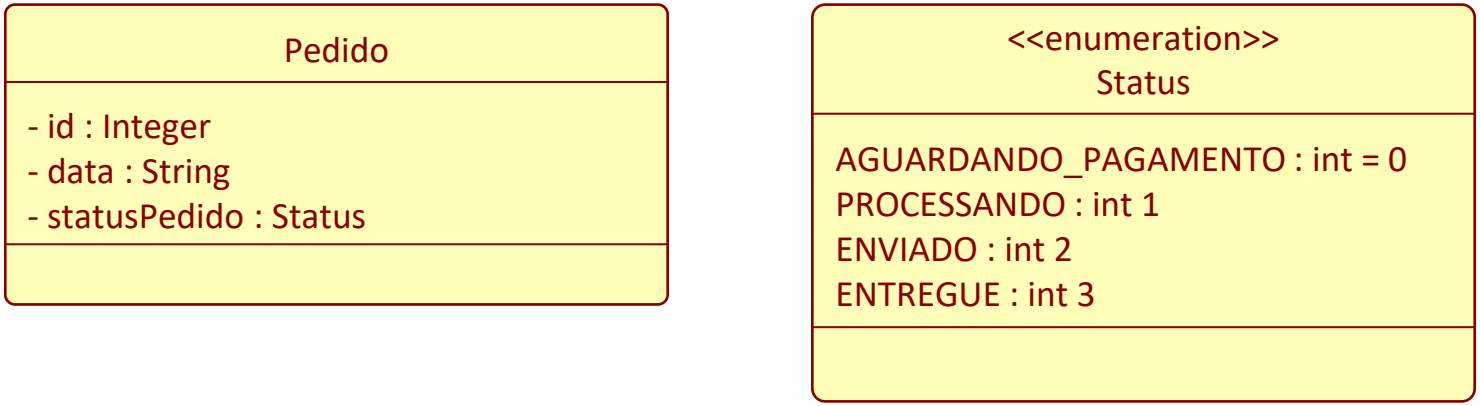
O status pode ser: aguardando pagamento, processando, enviado e entregue.

Exemplo 1


<<enumeration>>
Status
AGUARDANDO_PAGAMENTO : int = 0
PROCESSANDO : int 1
ENVIADO : int 2
ENTREGUE : int 3

Modelagem do Sistema

Modelagem do Sistema

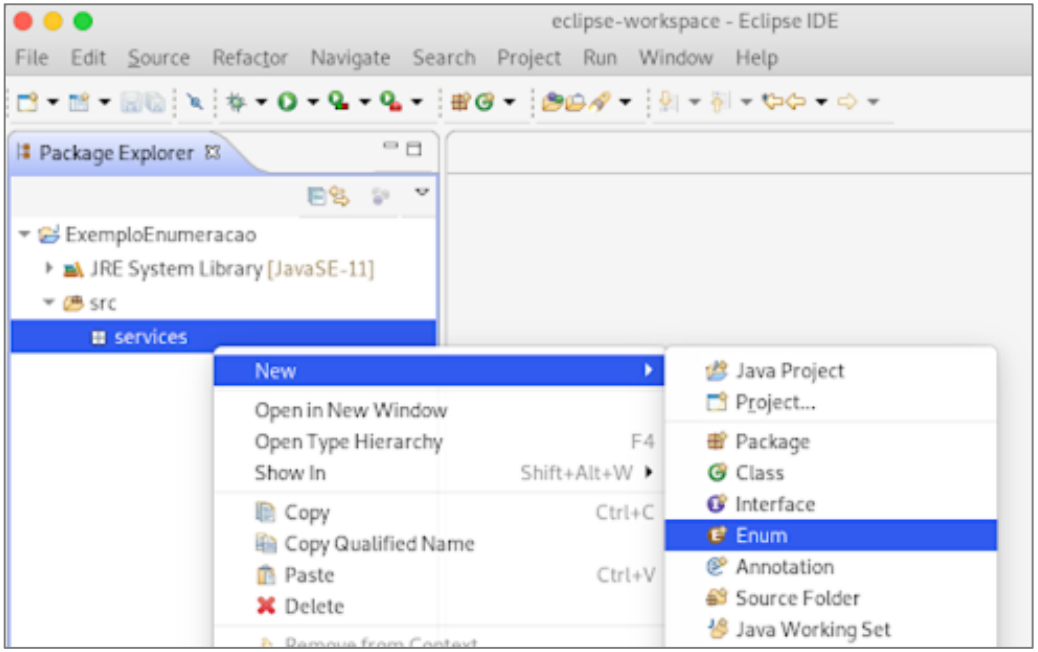


Implementação

Implementação

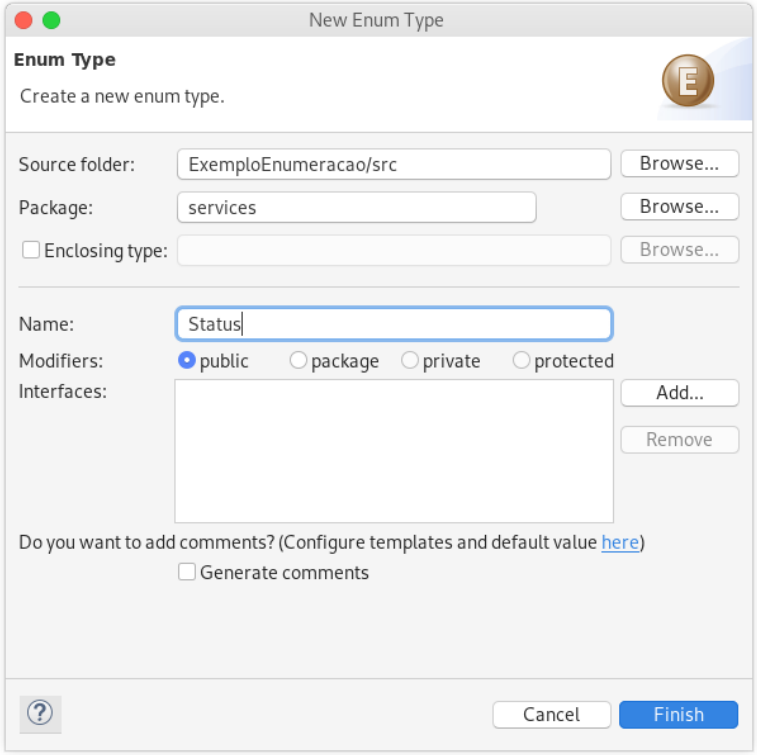
Clique com o botão direito do mouse no pacote criado.

New > Enum



Implementação

Name: Status



Implementação

```
public enum Status {

    AGUARDANDO_PAGAMENTO,
    PROCESSANDO,
    ENVIADO,
    ENTREGUE;
}
```

Implementação

Palavra reservada enum indica que é uma classe do tipo Enumeration.

```
public enum Status {

    AGUARDANDO_PAGAMENTO,
    PROCESSANDO,
    ENVIADO,
    ENTREGUE;
}
```

Implementação

As constantes são programadas no corpo da classe separadas por vírgulas. Ao final da instrução, coloca-se ponto e vírgula (;).

```
public enum Status {

    AGUARDANDO_PAGAMENTO,
    PROCESSANDO,
    ENVIADO,
    ENTREGUE;
}
```

Implementação

```java
public class Pedido {

    private int id;
    private String data;
    private Status statusPedido;

    public Pedido(int id, String data) {

        this.id = id;
        this.data = data;
        this.statusPedido = Status.AGUARDANDO_PAGAMENTO;
    }

    public void imprimirPedido() {

        System.out.println("Código: " + this.id);
        System.out.println("Data: " + this.data);
        System.out.println("Status: " + this.statusPedido);
    }

    public void setStatus(Status statusPedido) {

        this.statusPedido = statusPedido;
    }
}
```

```java
public class PedidoTeste {

    public static void main(String[] args) {

        Pedido pedido1 = new Pedido(1, "10/10/2010");
        pedido1.imprimirPedido();

        System.out.println("----------------------------");
        pedido1.setStatus(Status.PROCESSANDO);
        pedido1.imprimirPedido();

        System.out.println("----------------------------");
        pedido1.setStatus(Status.ENVIADO);
        pedido1.imprimirPedido();

        System.out.println("----------------------------");
        pedido1.setStatus(Status.ENTREGUE);
        pedido1.imprimirPedido();
    }
}
```

```
Código: 1
Data: 10/10/2010
Status: AGUARDANDO_PAGAMENTO
----------------------------
Código: 1
Data: 10/10/2010
Status: PROCESSANDO
----------------------------
Código: 1
Data: 10/10/2010
Status: ENVIADO
----------------------------
Código: 1
Data: 10/10/2010
Status: ENTREGUE
```

Exercício 1

Exercício 1

Desenvolva um sistema para manter cadastro de cliente de um determinado estabelecimento. O cliente é identificado por: código, nome, data de nascimento e estado civil.

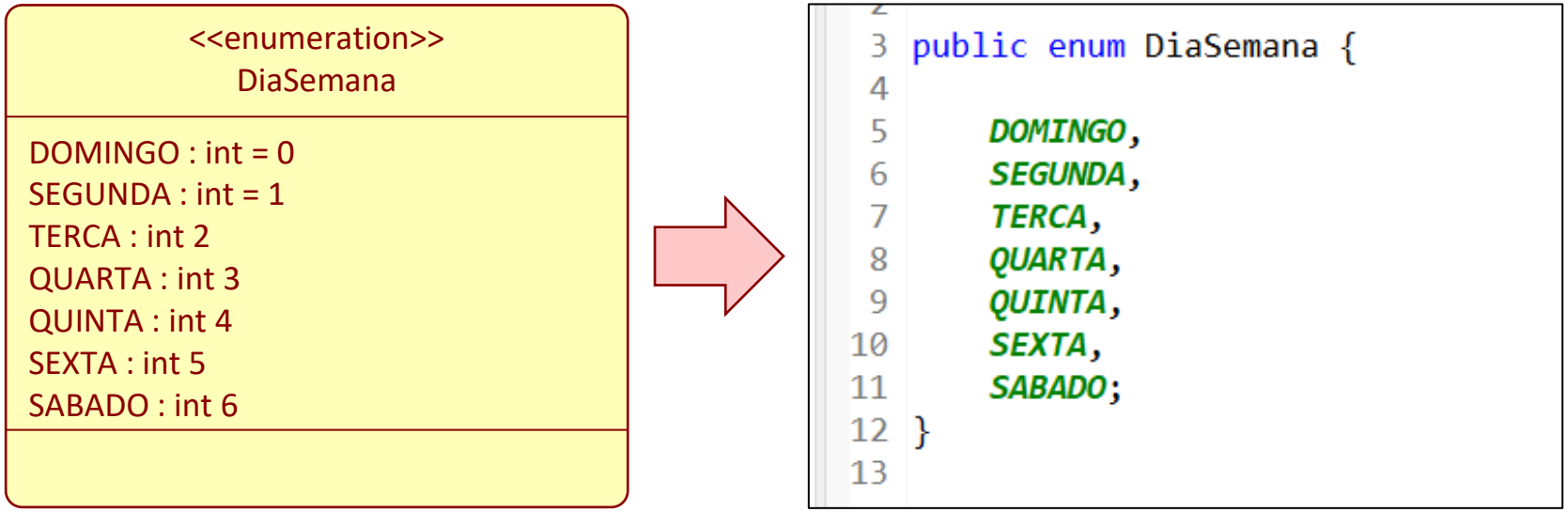
O estado civil pode ser: solteiro, casado, divorciado e viúvo.

Exemplo 2

Exemplo 2

Para um segundo exemplo, pode-se utilizar os dias da semana. Para isso, uma enumeration conterá os valores: DOMINGO, SEGUNDA, TERCA, QUARTA, QUINTA, SEXTA e SABADO.

Exemplo 2


<<enumeration>>
DiaSemana
DOMINGO : int = 0
SEGUNDA : int = 1
TERCA : int 2
QUARTA : int 3
QUINTA : int 4
SEXTA : int 5
SABADO : int 6
public enum DiaSemana {
DOMINGO,
SEGUNDA,
TERCA,
QUARTA,
QUINTA,
SEXTA,
SABADO;
}

Exemplo 2

Porém, neste exemplo incluiremos detalhes a cada indicador, como por exemplo: número do dia da semana e dia da semana por extenso.

Exemplo 2

```
2
3  public enum DiaSemana {
4
5      DOMINGO(1, "domingo"),
6      SEGUNDA(2, "segunda-feira"),
7      TERCA(3, "terça-feira"),
8      QUARTA(4, "quarta-feira"),
9      QUINTA(5, "quinta-feira"),
10     SEXTA(6, "sexta-feira"),
11     SABADO(7, "sábado");
12
13     private int numero;
14     private String dia;
15
16     private DiaSemana(int numero, String dia) {
17
18         this.numero = numero;
19         this.dia = dia;
20     }
21
22     public int getNumero() {
23         return numero;
24     }
25
26     public String getDia() {
27         return dia;
28     }
29 }
```

Exemplo 2

```
2
3  public enum DiaSemana {
4
5      DOMINGO(1, "domingo"),
6      SEGUNDA(2, "segunda-feira"),
7      TERCA(3, "terça-feira"),
8      QUARTA(4, "quarta-feira"),
9      QUINTA(5, "quinta-feira"),
10     SEXTA(6, "sexta-feira"),
11     SABADO(7, "sábado");
12
13     private int numero;
14     private String dia;
15
16     private DiaSemana(int numero, String dia) {
17
18         this.numero = numero;
19         this.dia = dia;
20     }
21
22     public int getNumero() {
23         return numero;
24     }
25
26     public String getDia() {
27         return dia;
28     }
29 }
```

Exemplo 2

```
2
3  public enum DiaSemana {
4
5      DOMINGO(1, "domingo"),
6      SEGUNDA(2, "segunda-feira"),
7      TERCA(3, "terça-feira"),
8      QUARTA(4, "quarta-feira"),
9      QUINTA(5, "quinta-feira"),
10     SEXTA(6, "sexta-feira"),
11     SABADO(7, "sábado");
12
13     private int numero;
14     private String dia;
15
16     private DiaSemana(int numero, String dia) {
17
18         this.numero = numero;
19         this.dia = dia;
20     }
21
22     public int getNumero() {
23         return numero;
24     }
25
26     public String getDia() {
27         return dia;
28     }
29 }
```

Exemplo 2

```
2
3  public enum DiaSemana {
4
5      DOMINGO(1, "domingo"),
6      SEGUNDA(2, "segunda-feira"),
7      TERCA(3, "terça-feira"),
8      QUARTA(4, "quarta-feira"),
9      QUINTA(5, "quinta-feira"),
10     SEXTA(6, "sexta-feira"),
11     SABADO(7, "sábado");
12
13     private int numero;
14     private String dia;
15
16     private DiaSemana(int numero, String dia) {
17
18         this.numero = numero;
19         this.dia = dia;
20     }
21
22     public int getNumero() {
23         return numero;
24     }
25
26     public String getDia() {
27         return dia;
28     }
29 }
```

Exemplo 2

```
2
3  public enum DiaSemana {
4
5      DOMINGO(1, "domingo"),
6      SEGUNDA(2, "segunda-feira"),
7      TERCA(3, "terça-feira"),
8      QUARTA(4, "quarta-feira"),
9      QUINTA(5, "quinta-feira"),
10     SEXTA(6, "sexta-feira"),
11     SABADO(7, "sábado");
12
13     private int numero;
14     private String dia;
15
16     private DiaSemana(int numero, String dia) {
17
18         this.numero = numero;
19         this.dia = dia;
20     }
21
22     public int getNumero() {
23         return numero;
24     }
25
26     public String getDia() {
27         return dia;
28     }
29 }
```

Exemplo 2

O Funcionário de uma empresa tem direito em folgar um dia da semana. Então, além dos próprios atributos de funcionário, haverá um objeto do tipo enumeration DiaSemana.

Exemplo 2

```java
public class Funcionario {

    private int registro;
    private String nome;
    private String cpf;
    private DiaSemana folga;

    public Funcionario(int registro, String nome, String cpf) {

        this.registro = registro;
        this.nome = nome;
        this.cpf = cpf;
    }

    public void setFolga(DiaSemana folga) {

        this.folga = folga;
    }

    public void imprimirRelatorio() {

        System.out.println("Registro: " + this.registro);
        System.out.println("Nome: " + this.nome);
        System.out.println("CPF: " + this.cpf);
        System.out.println("Número do dia da Semana: " + this.folga.getNumero());
        System.out.println("Dia da Semana de Folga: " + this.folga.getDia());
    }
}
```

Exemplo 2

```java

public class DiaSemanaTeste {

    public static void main(String[] args) {

        Funcionario funcionario1 = new Funcionario(111, "João da Silva", "123.456.789-11");

        funcionario1.setFolga(DiaSemana.QUARTA);

        funcionario1.imprimirRelatorio();
    }
}
```

Properties Console

<terminated> DiaSemanaTeste [Java Application] C:\Users\vinic\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_14.0.2.v20200815-0932\jre\bin\javaw.exe (13 de abr.

```
Registro: 111
Nome: João da Silva
CPF: 123.456.789-11
Número do dia da Semana: 4
Dia da Semana de Folga: Quarta-Feira
```

Exercício 2

Exercício 2

Implemente um sistema para manter os dados de uma empresa. A empresa é identificada pela sua Razão Social, cnpj e unidade federativa que encontra-se sua matriz. A unidade federativa será a classe enumeration e em cada indicador deverá armazenar os valores: nome da unidade federativa, sigla e nome da capital. Cadastre apenas as unidades federativas da região sul e sudeste.

Implemente uma classe de teste para instanciar as empresas e atribuir os valores a elas, bem como imprimir seus dados em console.