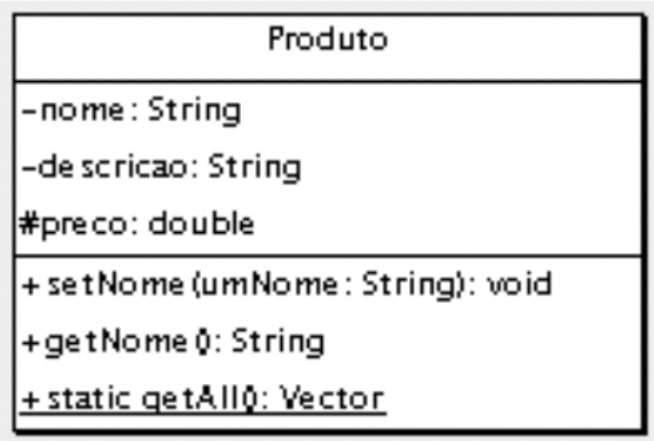
Mapeando diagrama de classe para código

Mapeando diagrama de classe para código

- Implementar um sistema baseado em um modelo
- Depende primeiro do paradigma usado na construção do modelo
- Depois do paradigma da linguagem escolhida para o desenvolvimento
- É possível construir um sistema a partir de um modelo orientado a objetos usando uma linguagem estruturada e vice-versa?
- Mas, para isso teremos que fazer diversas adaptações.

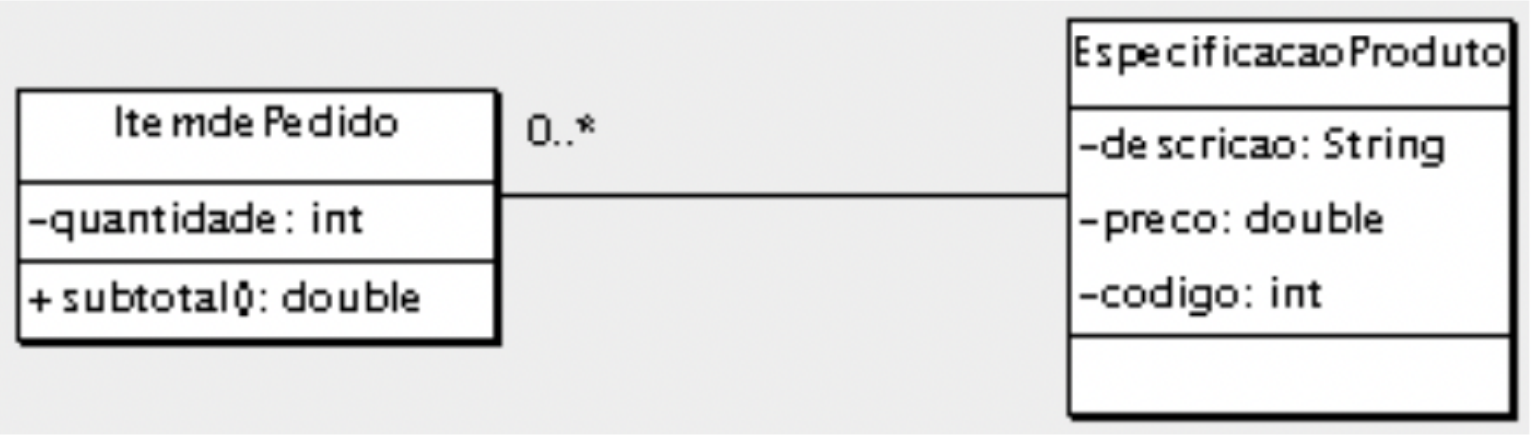
Mapeando diagrama de classe para código




```
class Produto {
        private String nome;
        private String descricao;
        protected double preco;
        public void setNome(String umNome)  {}
        public String getNome()   {return null;}
        static public Vector getAll() {return null;}
}
```

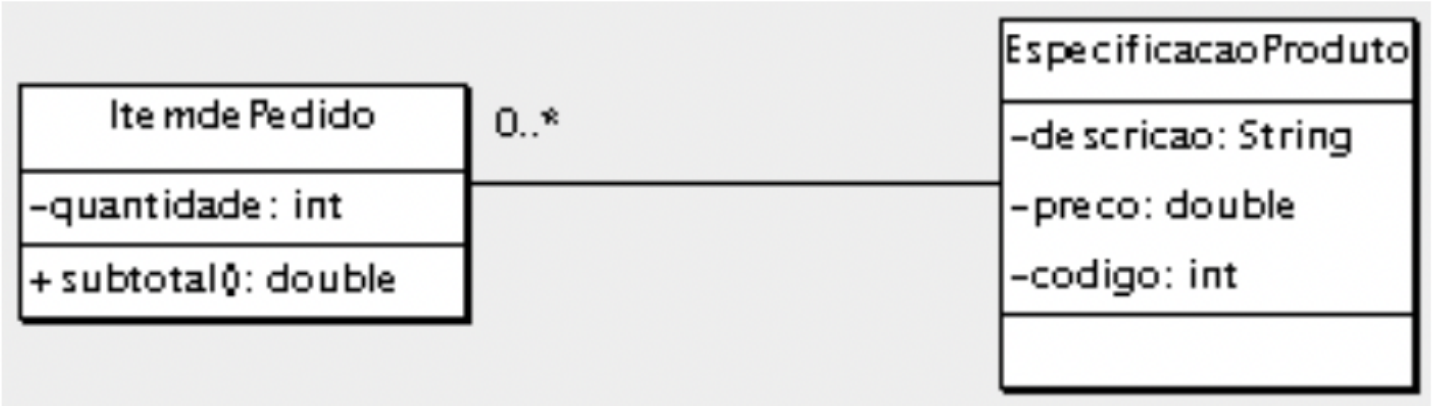
Mapeando diagrama de classe para código

- Implementando associação
  - Precisamos colocar um atributo de referência
  - Em qual classe será colocado este atributo?
  - Depende da multiplicidade da associação



Mapeando diagrama de classe para código

- Implementando associação
  - um para muitos

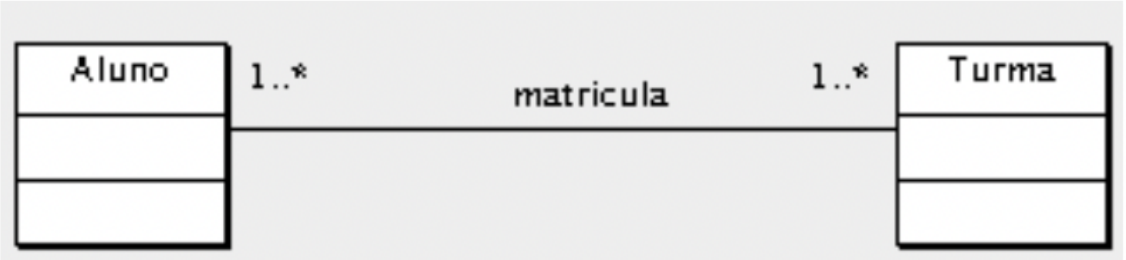



```
public class ItemdePedido
{
      private int quantidade;
      private EspecificacaoProduto produto;
}
```

```
public class ItemDePedido
{
      private int quantidade;
      private int codProduto;
}
```

Mapeando diagrama de classe para código

- Implementando associação
  - muitos para muitos

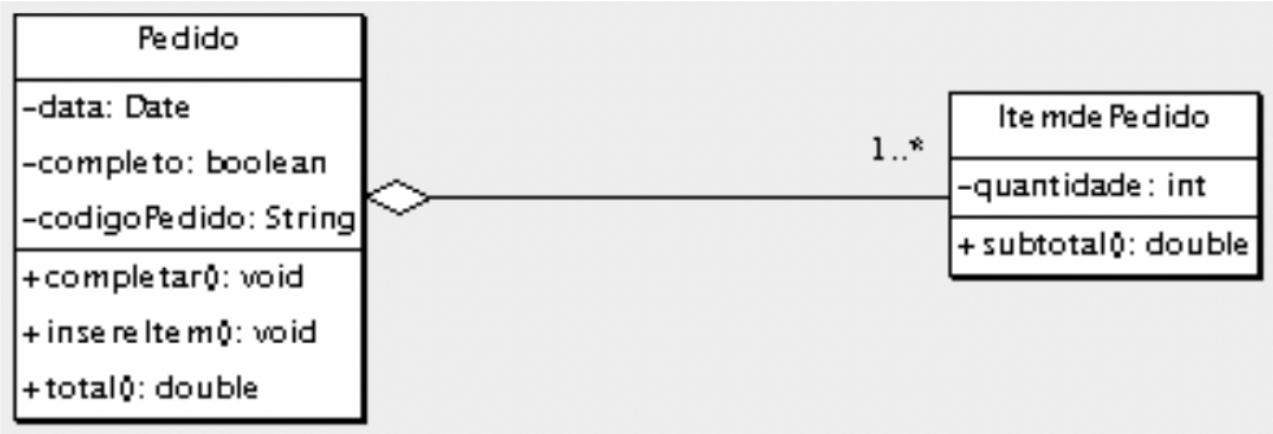


Existe a necessidade da criação de uma classe para a associação:

```
public class Matricula
{
      private Aluno aluno;
      private Turma turma;
}
```

Mapeando diagrama de classe para código

- Implementando agregação
  - Pode ser implementada usando um objeto para coleções
  - Este objeto é colocado como atributo da classe
  - Ele contém contém objetos da classe agregada;




```
public class Pedido {
        private Date data;
        private String codPedido;
        private boolean completo;
        private Vector todosItens;
        private List<ItemdePedido> todosItens;
}
```