Classes incorporáveis - JPA

Casses incorporáveis são como componentes de outras entidades. Por exemplo, minhas entidades de **Restaurante** e de **Pedido** usam um **Endereço**, nesse caso eu poderia mapear elas para que fizessem algum tipo de relação.

Ou poderia tornar **Endereço** um c**omponente incorporável,** a Ideia é que Endereço seja incorporado a entidade já existente sem a necessidade de criar um relacionamento, então para que Endereço se torne incorporável é usado a anotação **@Embeddable:**

```java
@Embeddable
public class Endereco {

    @Column(name = "endereco_cep")
    private String cep;

    @Column(name = "endereco_logradouro")
    private String logradouro;

    @Column(name = "endereco_numero")
    private String numero;

    @Column(name = "endereco_complemento")
    private String complemento;

    @Column(name = "endereco_bairro")
    private String bairro;

    @ManyToOne
    @JoinColumn(name = "endereco_cidade_id")
    private Cidade cidade;
}
```

A Entidade que incorporar Endereço terá seus atributos refletidos nela mesma, então é bom dar uns nomes bem legíveis*

A classe que vai incorporar precisa obviamente anotar a propriedade como incorporado usando **@Embedded:**

```java
@Embedded
private Endereco endereco;
```

id	endereco_bairro	endereco_cep	endereco_complemento	endereco_logradouro	endereco_numero	nome
1	NULL	NULL	NULL	NULL	NULL	TemakiOdio
2	NULL	NULL	NULL	NULL	NULL	MiguelJoquim

==========

Payload grande

Novamente, se as **classes de modelo forem as mesmas da representação dos recuros**, então eles tendem a ficar com um corpo muito grande durante as requisições:

O que **novamente nos leva a uma análise de que endpoint realmente vale a pena ter todos os dados da representação de um recurso**, numa listagem de todos os recursos talvez não, em um Singleton provavelmente sim.