

Criando o Message-Driven Bean

No pacote `br.com.casadocodigo.loja.service`, crie a classe `EnviaEmailCompra`, que será o nosso [Message-Driven Bean \(MDB\)](http://docs.oracle.com/javaee/6/tutorial/doc/gipko.html) (<http://docs.oracle.com/javaee/6/tutorial/doc/gipko.html>).

Faça com que a classe implemente a interface `MessageListener`, e implemente o método `onMessage` desta interface:

```
package br.com.casadocodigo.loja.service;

import javax.jms.Message;
import javax.jms.MessageListener;

public class EnviaEmailCompra implements MessageListener {

    @Override
    public void onMessage(Message message) {

    }
}
```

Registre o `EnviaEmailCompra` como um MDB, para isso utilize a anotação `@MessageDriven`, do pacote `javax.ejb.MessageDriven`:

```
@MessageDriven(activationConfig = {
    @ActivationConfigProperty(
        propertyName="destinationLookup",
        propertyValue="java:/jms/topics/CarrinhoComprasTopico")
})
public class EnviaEmailCompra implements MessageListener {

    //...

}
```

Receba um `MailSender` e um `CompraDao` via injeção de dependências:

```
// ...
public class EnviaEmailCompra implements MessageListener {

    @Inject
    private MailSender mailSender;

    @Inject
    private CompraDao compraDao;
```

Faça com que dentro do método `onMessage()`, o e-mail que antes era enviado no `PagamentoService` seja enviado na classe `EnviaEmailCompra` (mova o código).

Obtenha o uuid da compra a partir do `Message`. Dessa forma você poderá utilizar o e-mail do usuário e uuid da compra para construir o e-mail a ser enviado:

```
TextMessage textMessage = (TextMessage) message;  
  
Compra compra = compraDao.buscaPorUuid(textMessage.getText());
```