Durante as nossas aulas, estamos utilizando uma divisão de pacotes bem específica: para cada parte do projeto, ou camada, criamos um pacote diferente. Por isso trabalhamos com controller, model, repository e outros pacotes. Esse estilo de organização é chamado de **Package by Layer**, ou pacotes por camadas.

O Package by Layer é uma abordagem que diz que você deve dividir seu código com base em suas responsabilidades funcionais. Isso pode incluir coisas como 'model', 'view', 'controller', e 'repository'. Cada camada tem uma responsabilidade específica. Por exemplo, a camada 'view' manipula a interface do usuário, enquanto a camada 'controller' lidará com a lógica de negócio.

Vamos considerar um exemplo simples. Digamos que estamos construindo um aplicativo de blog. Com a abordagem 'Package by Layer', poderíamos ter pacotes assim:

com.myblog

.controller

.PostController

.CommentController

.model

.Post

.Comment

.repository

.PostRepository

.CommentRepository

COPIAR CÓDIGO

Neste exemplo, todas as classes relacionadas aos posts do blog estão espalhadas por diferentes pacotes, baseados na função que desempenham. O mesmo se aplica às classes de comentários.

Porém, existe um outro tipo de organização, utilizado, por exemplo, na formação Spring Boot. Ele é chamado **Package by Feature**, ou pacotes por funcionalidades. Ele sugere que você deve organizar seu código com base nos recursos individuais do seu aplicativo. Em vez de dividir seu código com base em sua função, você divide com base no recurso que ele implementa.

Usando o mesmo exemplo do blog, com 'Package by Feature', teríamos algo assim:

com.myblog

.post

.Post

.PostController

.PostRepository

.comment

.Comment

.CommentController

.CommentRepository

COPIAR CÓDIGO

Neste exemplo, todas as classes relacionadas aos posts estão no mesmo pacote. O mesmo se aplica aos comentários. Cada pacote é autossuficiente e contém tudo o que precisa para implementar um recurso específico.

**Quando usar cada um**

Então, qual abordagem você deve usar? Depende. 'Package by Layer' pode ser útil se você tiver uma equipe grande e complexa, na qual muitas pessoas podem estar trabalhando em diferentes camadas ao mesmo tempo. Ele separa as responsabilidades claramente, portanto, é menos provável que as pessoas pisem nos pés umas das outras.

No entanto, 'Package by Feature' é muitas vezes preferido para projetos menores e mais ágeis. Ele mantém todas as classes relacionadas a um recurso juntas, tornando mais fácil para um desenvolvedor entender completamente um recurso. Também é mais fácil de manter, porque quando um recurso é adicionado ou removido, você sabe exatamente onde todas as classes relacionadas estão.

Aqui, optamos por utilizar o Package by Layer, mas é interessante que você analise todas as condições para ver a estrutura que melhor se adequa a seu projeto.