

Classes internas

Classes aninhadas e anônimas

Tópicos

- Contextualização: Design Pattern Observer
- Classes internas aninhadas
- Classes anônimas
- Acesso externo às classes internas
- Classes de instância X. Classes “estáticas”
- Acesso a atributos e variáveis da classe externa, pela classe interna
- Builder pattern

Design Pattern Observer (Listener)

- Objetos se registram para “escutar” um evento que pode ocorrer
- Muito usado na camada de visualização de aplicações, quando ocorre um clique em botão, toque na tela de um celular, etc.
 - O listener se inscreve para ser observar um determinado evento
 - O evento é disparado quando ocorre a ação
 - O listener inscrito é chamado.

Observer - Exemplo

- pacote widget
 - Button - fornece 3 eventos: clique, soltar botão e duplo clique
 - Screen - Tela simples, possui apenas uma lista de botões
 - OnClickListener, OnDoubleClickListener e OnReleaseListener - possuem as assinaturas dos métodos que serão chamados.
- pacote exemploListener
 - MyScreen - inclui um botão de login e dois observers para os eventos de clique e soltar.
 - ExemploMain - simula uma sequência de ações do usuário, para disparar os respectivos eventos.

Classes internas aninhadas

- Uma classe não precisa necessariamente estar em um arquivo próprio
- Pode ser declarada como uma classe interna, dentro de outra classe ou método
- Classe mais externa deve ser obrigatoriamente do mesmo nome do arquivo
- `pacote exemploClasseAninhada`

Classes anônimas

- Classes internas não precisam necessariamente ser nomeadas
- Classe anônima é declarada “instanciando” a classe abstrata e implementando os métodos abstratos *on the fly*.
- Útil quando classe é usada apenas uma vez (na própria declaração)
- MyScreen no pacote exemploClasseAnônima
- obs: classes anônimas não podem ter construtores.

Acesso externo às classes internas

- Como acessar classes internas a partir de outras classes?
- Declarar as classes internas como públicas ou *package*. Acesso deve ser com o prefixo da classe externa que a contém: *Externa.Interna*
- Exemplo: EventTypeEnum em Button no pacote widgetWithExt está público. No ExemploMain, é acessado como Button.EventTypeEnum.

Classes de instância X. Classes “estáticas”

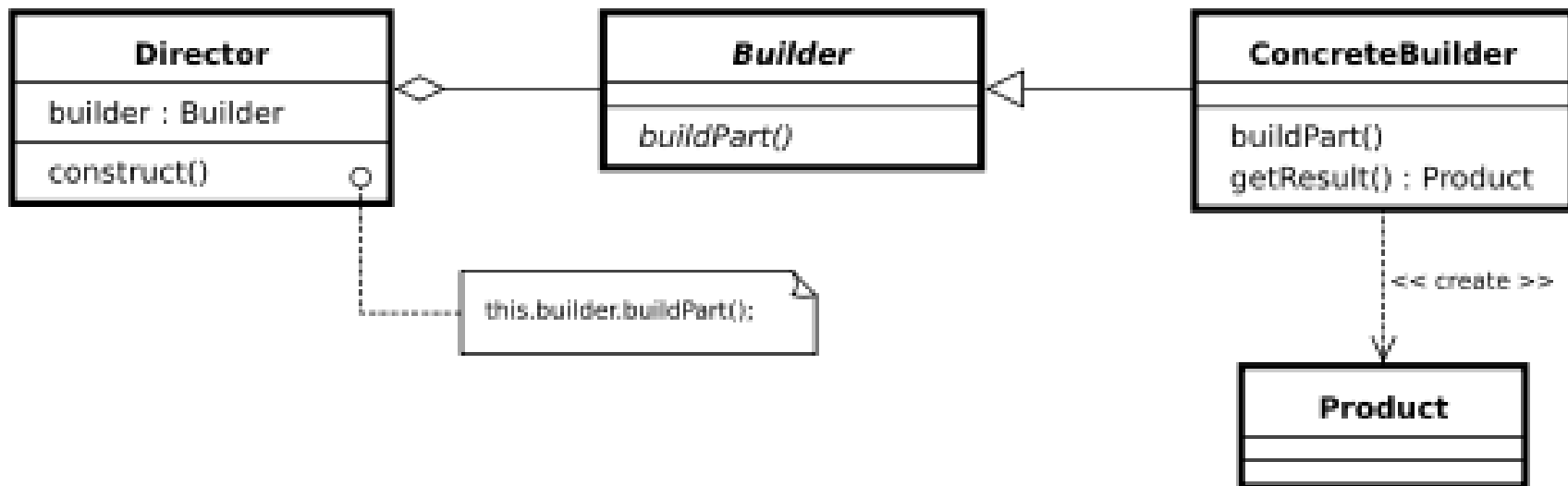
- Classe interna de instância (ver em Button)
 - Depende de uma instância da classe externa para que seja acessada, instanciada, etc.
 - Possui acesso aos atributos da classe externa (label do botão por exemplo)
- Classe interna “estática”
 - Comporta-se como uma *outer class* qualquer, como se estivesse em seu próprio arquivo
 - Não necessita de instância para ser acessada, tendo uma sintaxe mais amigável
 - Não possui acesso a atributos de instância da classe externa

Acesso a variáveis da classe externa

- Atributos da classe externa são acessíveis pelas classes internas, como visto no método `getButtonLabel` da classe interna de instância
- Variáveis são acessíveis, desde que não sejam mutáveis
 - Exemplo: no pacote `widgetWithExt`, no construtor de `MyScreen`, a variável `buttonName` é acessível pelo `OnClickListener`, pois possui o modificador ***final***.
- Motivo: escopo da classe interna é diferente; variável imutável é passada por cópia debaixo dos panos
- A partir do java 8, caso a variável não seja modificada, ela é implicitamente declarada como `final` (effectively final)

O design pattern Builder

- Cada “parte” é construída separadamente
- método para construir gera o Produto final
- Evita construtores massivos, com diversos parâmetros



Resumindo...

- Classes internas podem ser:
 - Aninhadas ou Anônimas
 - podem ter vários níveis de aninhamento
 - Internas à classe ou a métodos
 - de instância ou estáticas
- São acessíveis a partir de classes terceiras, desde que a visibilidade permita
- Os atributos da classe externa (que contém a interna) são visíveis pela classe interna

Dúvidas?



Referências

<http://blog.caelum.com.br/classes-aninhadas-o-que-sao-e-quando-usar/>

<http://www.inf.ufes.br/~vitorsouza/wp-content/uploads/java-br-curso-basico-novo-slides08.pdf>

<http://www.caelum.com.br/apostila-java-testes-xml-design-patterns/interfaces-graficas-com-swing/#5-10-classes-internas-e-anonimas>

https://en.wikipedia.org/wiki/Builder_pattern

Código dos exemplos: <https://github.com/joaoscalett/classes-internas>