INSTITUTO MAUÁ DE TECNOLOGIA

Linguagens I

Convenções e organização

Profº. Tiago Sanches da Silva

Organização

Quando um programador utiliza as classes feitas por outro, surge um problema clássico: como escrever duas classes com o mesmo nome?

Pode ser que a minha classe de **Data** funcione de um certo jeito, e a classe **Data** de um colega, de outro jeito. Pode ser que a classe de **Data** de uma biblioteca do **java** funcione ainda de uma terceira maneira diferente.

Como permitir que tudo isso realmente funcione? Como controlar quem quer usar qual classe de **Data**?

Organização

- Crie um novo projeto: OrganizandoJava
- Crie uma classe chamada Cliente
- Crie uma segunda classe chamada Cliente também

O que aconteceu? Por que?

Pacotes

Pacotes

O que é um pacote?

Um pacote é uma coleção de classes relacionadas que provê acesso protegido e gerenciamento de espaço de nomes (**Namespace**).

Em um **namespace** não pode existir classes, interfaces e arquivos com mesmo nome, ou seja, para criar duas classes Cliente é necessário coloca-las em um **namespace** diferente, em outras palavras, em pacotes diferentes.

Organização - Discussão

- Crie um novo pacote no projeto OrganizandoJava
- Crie finalmente a segunda classe chamada Cliente

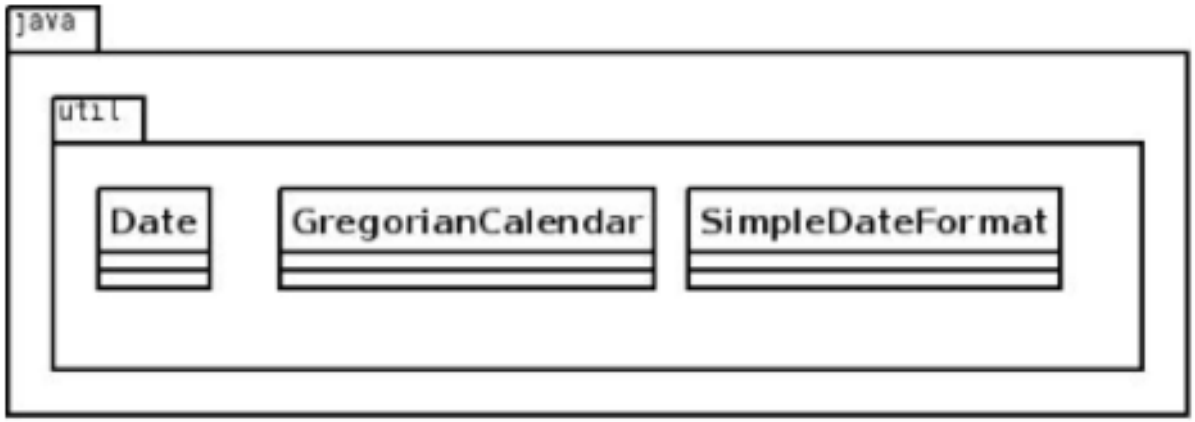
Deu certo?

Verifique a pasta **src** do projeto no sistema de pasta do SO. O que aconteceu?

Pacotes

Os **diretórios** estão diretamente relacionados aos **pacotes** e costumam agrupar classes de funcionalidades similares ou relacionadas.

Por exemplo, no pacote **java.util** temos as classes **Date**, **SimpleDateFormat** e **GregorianCalendar**; todas elas trabalham com datas de formas diferentes.



Padrão de nomenclatura dos pacotes

Padrão de nomenclatura

Padrão da nomenclatura dos pacotes

O padrão da sun para dar nome aos pacotes é relativo ao nome da empresa que desenvolveu a classe:

```
br.com.nomedaempresa.nomedoprojeto.subpacote
br.com.nomedaempresa.nomedoprojeto.subpacote2
br.com.nomedaempresa.nomedoprojeto.subpacote2.subpacote3
```

Os pacotes só possuem letras minúsculas, não importa quantas palavras estejam contidas nele. Esse padrão existe para evitar ao máximo o conflito de pacotes de empresas diferentes.

As classes do pacote padrão de bibliotecas não seguem essa nomenclatura, que foi dada para bibliotecas de terceiros.

https://docs.oracle.com/javase/tutorial/java/package/namingpkgs.html

Organização - Discussão

Fica fácil notar que a palavra chave **package** indica qual o pacote/diretório contém esta classe.

- Crie um novo pacote no projeto **OrganizandoJavaV2**
- Crie o nome do pacote utilizando br.com.<seuNome>.organizando
- Em duplas, criem a classe Cliente (cada um em seu projeto)
  - Adicione atributo nome
  - Um da dupla irá implementar **getNome**() o outro **retornaNome**()

Fully Qualified Name

Para utilizar uma classe que está em outro pacote eu posso utilizar o ***Fully Qualified Name*** da classe.

<pacote>.<Classe>

Que basicamente é o verdadeiro nome de uma classe, por isso duas classes com o mesmo nome em pacotes diferentes não conflitam.

```
br.com.tiago.lib.Cliente eu = new br.com.tiago.lib.Cliente();
```

Import

Caso você importe o pacote, não precisara se referenciar a classe através do ***Fully Qualified Name.***

```
import br.com.tiago.lib.Cliente;
```

package, import, class

É muito importante manter a ordem! Primeiro, aparece uma (ou nenhuma) vez o `package`; depois, pode aparecer um ou mais `imports`; e, por último, as declarações de classes.

Acesso a classe

As classes só são visíveis para outras no **mesmo pacote** e, para permitir que a classe de Teste veja e acesse a classe **Cliente** em outro pacote, a classe **Cliente** precisa ser **pública:**

```
class Cliente {

    public String name;
    public int idade;

}
```

```
public class Cliente {

    public String name;
    public int idade;

}
```

Vamos trabalhar – Discussão em sala

Vamos utilizar a classe criada pelo colega.

Como importar a classe do colega?

Vamos fazer juntos!

Perguntas?

Referências

- DevMedia (Robson Fernando)
- Oracle
- Caelum