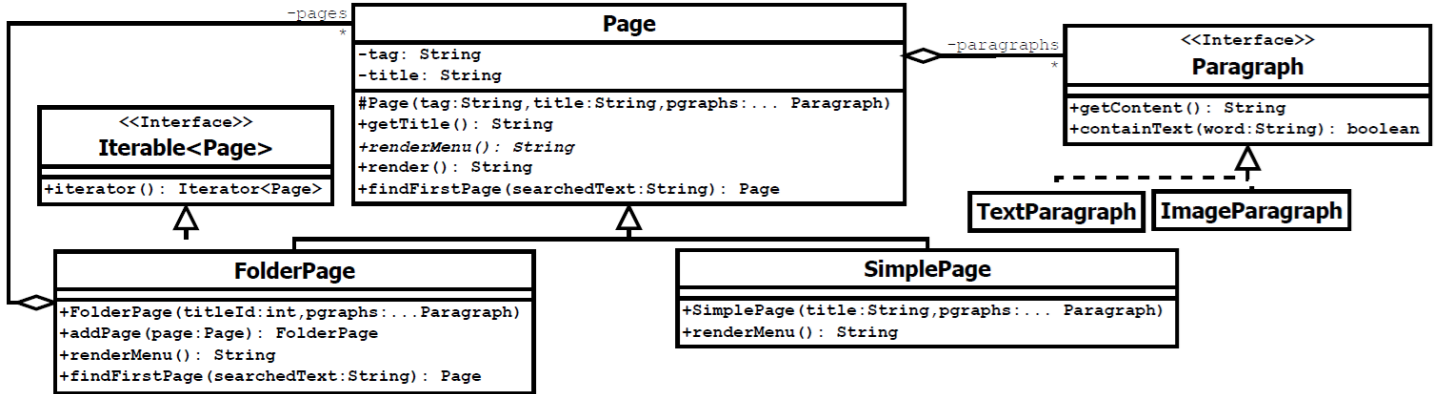


Pretende-se implementar uma hierarquia de classes que represente a organização e funcionamento de um gestor de conteúdos de páginas (CMS – *Content Management System*). Cada página (*Page*) contém um título, um menu e um conjunto de parágrafos (*Paragraph*). Um parágrafo pode ser de texto (*TextParagraph*) ou de imagem (*ImageParagraph*). Uma página pode ser simples (*SimplePage*) ou uma pasta (*FolderPage*). Uma pasta tem zero ou mais páginas dependentes.



Tendo em conta o diagrama estático de classes, os troços de código e os respetivos *outputs*, defina:

- [5] A Interface **Paragraph** e as classes **TextParagraph** e **ImageParagraph**. Instâncias de **ImageParagraph** não contêm texto, instâncias de **TextParagraph** contêm o texto caso a *string* passada no construtor o contenha. O seguinte troço de código exemplifica o que deve ser retornado pelo método **getContent** tendo em conta o valor recebido por parâmetro no construtor.

```
Paragraph txt = new TextParagraph("1ª imagem."); System.out.println(txt.getContent());
```

```
<txt>1ª imagem.</txt>
```

```
Paragraph img= new ImageParagraph( "image1.jpg" ); System.out.println( img.getContent() );
```

```

```

- [5] A classe abstrata **Page**. No construtor recebe por parâmetro a *tag*, o título, e os parágrafos. O método **getTitle** retorna o título e não pode ser redefinido. O método **renderMenu** é abstrato. O método **findFirstPage** retorna a própria página caso o título ou os parágrafos contenham a *string* passada por parâmetro, caso não contenham retorna **null**. O método **render** constrói uma *string* em que a primeira linha é a *tag* entre os símbolos '<' e '>', a segunda linha é o elemento título, nas linhas seguintes é o menu retornado pelo método **renderMenu**, e por fim o conteúdo dos parágrafos um por linha (ver o output de **SimplePage**).

- [2] A classe **SimplePage**. No construtor recebe por parâmetro o título, e os parágrafos. A *tag* de uma página simples é "page". O menu de uma página simples possui apenas um item com a indicação para voltar à página anterior.

```
Paragraph txt = new TextParagraph( "1ª imagem.");
Paragraph img= new ImageParagraph( "image1.jpg" );
SimplePage pg= new SimplePage( "1ª página", txt, img );
System.out.println ( pg.renderMenu() );
```

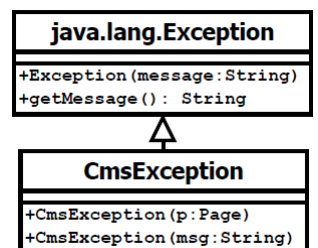
```
<menu><item>voltar</item></menu>
```

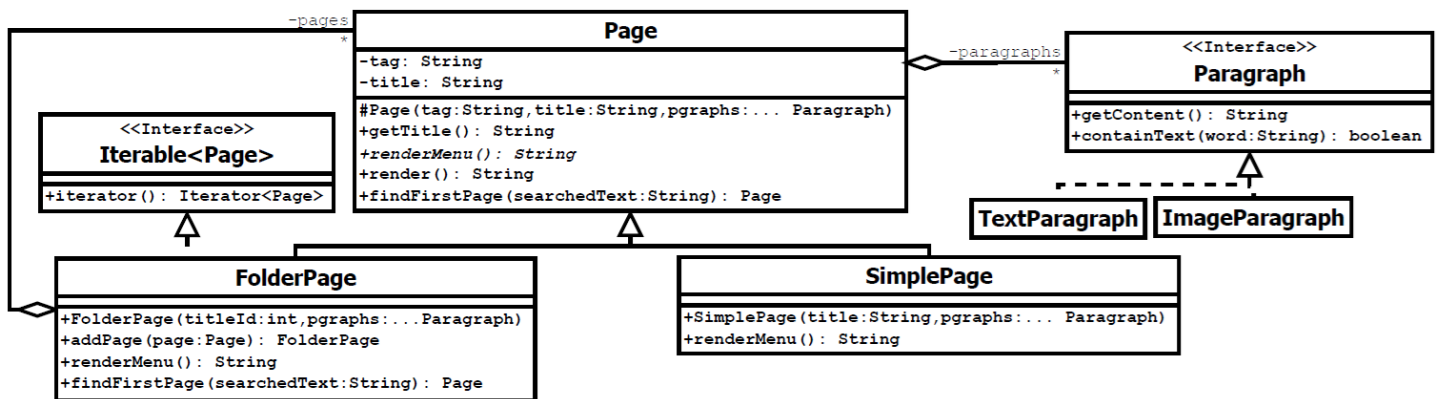
```
System.out.println( pg.render() );
```

```
<page>
  <title>1ª página</title>
  <menu><item>voltar</item></menu>
  <txt>1ª imagem.</txt>
  
</page>
```

- [2] Defina a classe **CmsException** para que o método **getMessage** herdado de **Exception** retorne:

- A *string* "Duplicate page:" seguida do título da página entre aspas, caso tenha sido instanciado com o construtor com um parâmetro do tipo **Page**;
- A *string* passada por parâmetro no construtor caso tenha sido instanciado com o construtor com o parâmetro do tipo **String**.





5. [6] Defina a classe **FolderPage** tendo em conta que:

- No construtor recebe por parâmetro, o identificador do título, e os parágrafos. A **tag** de uma pasta é **"folder"** e o título é a **string** "Pasta " concatenada com o **titleId**.
- O método **findFirstPage** procura a **string** no título e nos parágrafos da própria página (chamada ao método **findFirstPage** herdado) caso encontre retorna a própria. Caso não encontre procura nas páginas dependentes e retorna a referência para a primeira página onde se encontra a **string**. Caso também não exista nas dependentes retorna **null**.
- O método **addPage** caso a página já exista lança a exceção **CmsException** passando-lhe por parâmetro a página. Caso contrário adiciona-a e retorna a própria pasta.
- O menu de uma pasta contém uma lista de itens com os títulos das páginas dependentes (ver o output do exemplo).
- O método **iterator** retorna o **Iterator<Page>** para as páginas dependentes.

```

Page pg1 = new SimplePage( "1ª página", new TextParagraph("1ª imagem"), new ImageParagraph("img1.jpg" ));
Page pg2 = new SimplePage( "2ª página", new TextParagraph("2ª imagem"), new ImageParagraph("img2.jpg"));
    
```

```

FolderPage f = new FolderPage( 1, new TextParagraph("Conjunto de páginas." ));
try {
    f.addPage( pg1 ).addPage( pg2 ).addPage(pg2);
}
catch( CmsException e ) {
    System.out.println( e.getMessage() );
}
    
```

Lança exceção

Duplicate page: "2ª página"

```
System.out.println( f.render() );
```

```

Page foundPage = f.findFirstPage("imagem");
if ( foundPage != null ) System.out.println( foundPage.render() );
    
```

```

<folder>
  <title>Pasta 1</title>
  <menu>
    <item>1ª página</item>
    <item>2ª página</item>
  </menu>
  <txt>Conjunto de páginas.</txt>
</folder>
    
```

```

<page>
  <title>1ª página</title>
  <menu><item>voltar</item></menu>
  <txt>1ª imagem.</txt>
  
</page>
    
```