

# Diazo

**Módulo 7**

Tema Diazo

A thick, solid red horizontal bar with rounded ends, positioned below the text 'Tema Diazo' and extending to the right edge of the slide.

# Diazo >> O que é?

É uma ferramenta para criação de temas.

Permite que o designer construa um tema puramente HTML, sem se preocupar com as particularidades do Plone.

Funciona através de um mapeamento (arquivo rules.xml) entre o tema estático (criado pelo designer) e o tema Plone.

# Componentes de um tema diazo

Além de outras coisas, um tema diazo é composto pelo **tema estático** e pelo arquivo **rules.xml**.

Tema estático:

Um tema qualquer, usando HTML, CSS, Javascript, etc.

Arquivo rules.xml:

Diz aonde cada informação apresentada no tema padrão do Plone será mostrada no tema estático.

# Exercícios

- Exercício 1 > Criando um novo pacote
- Exercício 2 > Adicionando o pacote ao buildout
- Exercício 3 > Ativando o novo pacote em um site Plone.

# Estrutura do novo pacote

treinamento.theme/setup.py: Contém instruções instalação destinadas ao buildout.

treinamento.theme/treinamento/: Um pacote *namespace*. Contém um `__init__.py`.

treinamento.theme/treinamento/theme: A raiz, de fato, do projeto. Todos os arquivos Python, ZCML, HTML, XML, CSS, etc estarão nessa pasta ou nas suas subpastas.

`__init__.py`: Geralmente um arquivo vazio. Usado pelo Python para diferenciar um pacote Python de uma pasta comum do sistema operacional.

# O arquivo rules.xml

Mostrar o rules.xml gerado pelo zopeskel.

# Diretivas diazo (1)

As diretivas a seguir são válidas dentro do elemento **<rules />**

**<theme />**

Usado para especificar arquivo html do tema.

Exemplo:

```
<theme href="theme.html" />
```

Podem ser usados atributos **if-content** para aplicar o tema apenas se certas condições forem satisfeitas. Exemplo:

```
<theme href="theme.html" css:if-content="#id_de_algun_elemento" />
```

**<notheme />**

Usado para desligar o tema em certas condições.

## Diretivas diazo (2)

`<replace />`

Usado para substituir um elemento (ou atributo de elemento) do tema por um elemento (ou atributo) do Plone.

Exemplos:

```
<replace theme="/html/head/title" css:content="/html/head/title" />
```

```
<replace css:theme"title" css:content="title" />
```

Obs: Podemos referenciar um elemento HTML no diazo usando seletores XPath ou CSS.



# Diretivas diazo (3)

`<before />` e `<after />`

Inserir um elemento do Plone antes ou depois de um elemento do tema estático, respectivamente.

Exemplos:

```
<before css:theme="#content" css:content="#info-box" />
```

```
<after css:theme="#content" css:content="#info-box" />
```

Se quiséssemos adicionar o elemento `#info-box` imediatamente antes de quaisquer elementos dentro de `#content`, poderíamos usar:

```
<before css:theme-children="#content" css:content="#info-box" />
```

# Diretivas diazo (4)

## <drop />

Usado para remover elementos (ou atributos de elementos) do tema estático ou do Plone. Essa é única diretiva que aceita os tanto o atributo **theme** quanto o atributo **content**, mas não os dois juntos.

Exemplo:

```
<drop css:content="#portal-content .about-box" />
```

```
<replace css:theme="#content" css:content="#portal-content" />
```

## <copy />

Usado para copiar atributos de elementos do Plone para o tema estático.

Exemplo:

```
<copy attributes="class" css:theme="body" css:content="body" />
```

# Atributos das diretivas

Podemos usar alguns atributos dentro das diretivas. Os mais comuns:

**if-content (ou css:if-content):** A diretiva será executada apenas se a condição for satisfeita. Por exemplo:

```
<rules css:if-content="#visual-portal-wrapper">
```

**attributes:** Algumas diretivas podem ser usadas tanto para elementos HTML (tags) quanto para os atributos dos elementos. Usando **attributes** estaremos aplicando a diretiva aos atributos do elemento HTML. Exemplo:

```
<merge attributes="class" css:theme="body" css:content="body" />
```

# Exemplo

Explicar as diretivas no rules.xml gerado pelo ZopeSkel.

# Requisitos do tema do Projeto Alfa

- Faz parte da nossa identidade visual a cor **vermelha** (**#c40929**).
- Nossa logo:

**projeto alfa**

- Nosso portal precisa ter largura fixa.
- Ao contrário do padrão do Plone, o menu principal deve estar acima da logo.

# Antes de começar...

Vamos deletar a última diretiva do rules.xml:

```
<after css:theme="body p" css:content="#visual-portal-  
wrapper" />
```

Ela está colocando o layout inteiro do Plone, vamos pegar apenas partes dele.

*Agora vamos construir o novo tema juntos...*

# 1o Requisito: O menu acima da logo > index.html

```
<body>  
  <div id="menu_principal"> MENU </div>  
  <div id="logo"> LOGO </div>  
</body>
```

Verifique como está o seu site após a modificação.

# 1o Requisito: O menu acima da logo > rules.xml (1)

Duas formas práticas de construir uma regra:

Usando o Inspetor HTML do seu browser.

Usando o editor de temas Diazo.

A regra:

```
<replace
  css:theme="div#menu_principal"
  css:content="ul#portal-globalnav"
/>
```



## 1o Requisito: O menu acima da logo > rules.xml (2)

A logo vamos colocar no tema estático, em index.html:

```
<div id="logo">  
  <a href="#">  
      
  </a>  
</div>
```

*Obs: A logo projetoalfa.png precisa estar na pasta seutema/diazo\_resources/static/.*

# 1o Requisito: O menu acima da logo > rules.xml (3)

Observe que o link da logo ainda não está correto, ele precisa apontar sempre para a página inicial, vamos resolver isso copiando o **href** do link da logo original:

```
<copy  
  attributes="href"  
  css:theme="#logo a"  
  css:content="a#portal-logo" />
```

**Teste:** Verifique que o link da logo agora aponta para página inicial, independente do contexto.

## 2o Requisito: A largura do portal precisa ser fixa (1)

(A largura será de 60em)

Vamos implementar isso usando CSS, em index.html:

## 2o Requisito: A largura do portal precisa ser fixa (2)

Vamos implementar isso usando CSS, em index.html:

```
<head>  
  <link rel="stylesheet" type="text/css" href="static/  
projetoalfa.css" />  
</head>
```

Criar o arquivo projetoalfa.css na pasta static.

## 2o Requisito: A largura do portal precisa ser fixa (3)

(A largura será de 60em)

No projeto\_alfa.css:

```
body {  
    width: 60em;  
    margin: 0 auto;  
}
```

# Completando o Layout >> Conteúdo (1)

Agora vamos colocar em nosso tema toda a parte do conteúdo abaixo da navegação principal.

# Completando o Layout >> Conteúdo (2)

Agora vamos colocar em nosso tema toda a parte do conteúdo abaixo da navegação principal.

Em index.html:

```
<div id="corpo">  
  Corpo  
</div>
```

# Completando o Layout >> Conteúdo (3)

Agora vamos colocar em nosso tema toda a parte do conteúdo abaixo da navegação principal.

Em rules.xml:

```
<replace  
  css:theme-children="div#corpo"  
  css:content="div#portal-columns"  
>
```



# Melhorando a aparência

Colocar uma quebra de linha entre a logo e o corpo. (<br />)

Centralizar a logo e colocar o background **vermelho** (#c40929).

Retirar o espaço em branco entre o menu principal e a logo.

# Melhorando a aparência

Centralizar a logo e colocar o background **vermelho** (**#c40929**).

```
#logo {  
    text-align: center;  
    background-color: #c40929;  
}
```

Retirar o espaço em branco entre o menu principal e a logo.

```
ul#portal-globalnav {  
    margin: 0;  
}
```

## 3o Requisito > Identidade com a cor vermelha

Mude a cor do background do link selecionado na navegação principal para **vermelho (#c40929)**.

Mude a cor do texto dos links não selecionados para **vermelho (#c40929)**.

Mude a cor dos links no conteúdo para **vermelho (#c40929)**.

DICA: Mudar a regra que adiciona os styles do Plone no seu tema de **after** para **before**. Com isso, basta que os seus seletores igualem a força dos seletores do tema padrão para que suas modificações sejam acatadas.

## 3o Requisito > Identidade com a cor vermelha

```
.plain a, #content a:link {  
    color: #c40929;  
}
```

```
#portal-globalnav .selected a,  
#portal-globalnav a:hover {  
    background-color: #c40929;  
}
```

# Desafio 1

A thick, solid red horizontal bar with rounded ends, positioned below the title.

Mudar a cor da barra de gestão de conteúdo do verde para vermelho (#c40929).

# Desafio 1 > Solução

```
#edit-bar,  
#content-views,  
#contentActionMenus,  
#contentActionMenus dl.actionMenu a,  
#contentActionMenus dl.actionMenu.activated dd {  
    background-color: #c40929;  
    border-color: #c40929;  
    border-bottom-color: #c40929;  
}
```

## Desafio 2

Ainda não colocamos o *dropdown* do `user_actions` (onde vemos “admin”, “Configuração do site”, “Sair”, etc).

Colocar no nosso tema.

# Desafio 2 > Solução

## INDEX.HTML

```
<!--Menu Principal -->  
<div id="user_actions">  
    User Actions  
</div>  
<!-- Logo -->
```

## RULES.XML

```
<replace  
    css:theme="#user_actions"  
    css:content="#portal-personaltools"  
>
```



## Desafio 3

Coloque o campo de busca no novo tema.

## Desafio 3 > Solução

### INDEX.HTML

```
<!-- User Actions -->  
<div id="busca"> Busca </div>  
<!-- Logo -->
```

### RULES.XML

```
<replace  
  css:theme=„#busca"  
  css:content="#portal-searchbox"  
>
```

## Desafio 4

Utilize a diretiva `<drop />` para remover o botão “Buscar”.

Mude a cor do texto logo abaixo do campo de busca “apenas nesta seção” para branco.

# Desafio 4 > Solução

## RULES.XML

```
<drop  
  css:content="#portal-searchbox .searchButton"  
>
```

## PROJETO\_ALFA.CSS

```
#portal-searchbox .searchSection {  
  color: white;  
}
```

# Desafio 5

Colocar o footer do Plone no nosso tema.

# Desafio 5 > Solução

## INDEX.XML (no final de body)

```
<div style="clear:both"> </div>
```

```
<div id="footer">
```

```
  Rodape
```

```
</div>
```

## RULES.XML

```
<replace
```

```
  css:theme="div#footer"
```

```
  css:content="div#portal-footer"
```

```
/>
```

**Perguntas?**

