

Aula 14

Professor:

Inês Dutra

Aplicação Biblioteca: Módulos

Conteúdo:

- Arquivos
- Esquema da Implementação
- Estruturas de dados e objetos
- Exemplos
- Possíveis melhoramentos

Arquivos

- **Forms**

- ▶ `formemprestimo.php`
- ▶ `formlivro.php`
- ▶ `formsolicitacao.php`
- ▶ `formusuario.php`
- ▶ `login.php`
- ▶ `logout.php`

- **Classes**

- ▶ `livro.php`
- ▶ `solicitacao.php`
- ▶ `tabelabd.php`
- ▶ `emprestimo.php`
- ▶ `usuario.php`

- **Funções de membro**

- ▶ `solicitar.php (livros)`
- ▶ `buscar.php`
- ▶ `consultar.php (solics e emprs)`
- ▶ `dadospessoais.php`

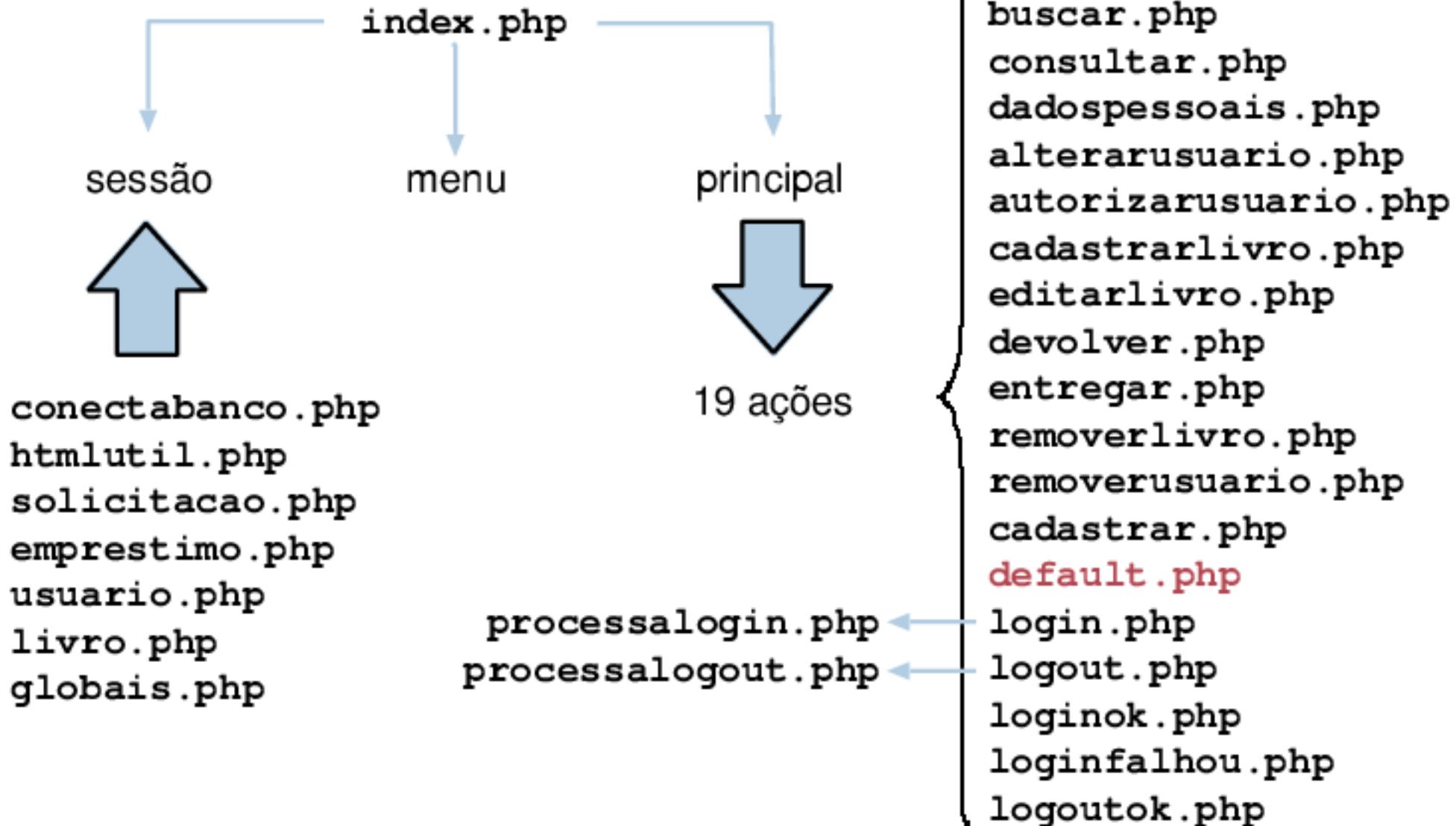
- **Funções de administração**

- ▶ `alterarusuario.php`
- ▶ `autorizarusuario.php`
- ▶ `cadastrarlivro.php`
- ▶ `editarlivro.php`
- ▶ `devolver.php`
- ▶ `entregar.php`
- ▶ `removerlivro.php`
- ▶ `removerusuario.php`

Arquivos

- **Funções de usuário qualquer**
 - ▶ `cadastrar.php`
- **Autenticação de login e senha**
 - ▶ `autenticacao.php`
- **Funções auxiliares**
 - ▶ `conectabanco.php`
 - ▶ `principal.php`
- **Telas**
 - ▶ `default.php`
 - ▶ `menu.php`
 - ▶ `index.php`
- **Estilos**
 - ▶ `estilos.css`
- **Constantes globais**
 - ▶ `globais.php`
- **Páginas de recibo**
 - ▶ `loginok.php`
 - ▶ `loginfalhou.php`
 - ▶ `logoutok.php`
- **Funções auxiliares para html**
 - ▶ `htmlutil.php`
- **Cria sessão e inicializa**
 - ▶ `sessao.php`
- **Métodos para login.php e logout.php**
 - ▶ `processallogin.php`
 - ▶ `process/logout.php`

Relações



index.php

```
<?php include ('sessao.php'); ?>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
    "http://www.w3.org/TR/html4/loose.dtd">

<html>
<head>
    <meta content="text/html; charset=ISO-8859-1" http-equiv="content-type">
    <title>Aplicação de Biblioteca</title>
    <link rel="stylesheet" href="estilos.css" type="text/css">
</head>
<body>
<table style="text-align: left; width: 100%; border="0" cellpadding="2" cellspacing="0">
    <tbody>
        <tr align="center">
            <td colspan="2" style="background-color: rgb(0, 0, 102); color: rgb(255, 255, 153);">
                <h1>Cederj - Programação II</h1>
                <h1>Aplicação de Biblioteca</h1>
            </td>
        </tr>
```

Abrir editor

index.php (continuação)

```
<tr>
    <td class = "colunaMenu">
        <?php include ("menu.php");?>
    </td>      <td class="colunaPrincipal">
        <div class="areaPrincipal">
            <?php include ('principal.php');?>
        </div>
    </td>
</tr>
<tr>
    <td colspan="2" style="background-color: rgb(0, 0, 102); "></td>
</tr>
</tbody>
</table>
<br>
</body>
</html>
```

sessao.php

```
<?php  
header("Content-Type: text/html; charset=ISO-8859-1");  
  
// Configurações globais  
require_once ('globais.php');  
  
// O sistema de bibliotecas requer acesso ao banco de dados  
require_once ('conectabanco.php');  
  
// Algumas funções úteis para formatacao de html  
require_once ('htmlutil.php');  
  
// Para poder armazenar objetos em $_SESSION é preciso definir  
// suas classes antes de abrir a sessão  
require_once ('solicitacao.php');  
require_once ('emprestimo.php');  
require_once ('usuario.php');  
require_once ('livro.php');  
  
// Abrir sessão  
session_start ();
```

Abrir editor

sessao.php (continuação)

```
// A função assegura_privilegio deve ser chamada por qualquer módulo que
// necessite de um privilégio mínimo de maneira a impedir
// a execução por usuários não credenciados.
// $privilegio = 'M' ou 'A' identificando prilégio de membro
//                 ou administrador

function assegura_privilegio ($privilegio) {
    if ($_SESSION['privilegio'] != 'A' &&
        $_SESSION['privilegio'] != $privilegio) {
        echo ('<h2> Acesso a esta parte do sistema é proibida a não membros ou
              a não administradores. Por favor peça cadastramento ou faça login.
              Se não possuir login, solicite cadastramento. </h2>');
        exit;
    }
}

// Asseguro que variável $_SESSION['privilegio'] tem algum valor,
// pelo menos o do privilégio default que é 'I':
if (!isset ($_SESSION['privilegio']))
    $_SESSION['privilegio'] = 'I';
?>
```

menu.php

```
<?php

// Estrutura do menu principal
$menu_principal=array (
    array("principal", "Página Inicial", "IMA"),
    array("buscar", "Procurar Livros", "IMA"),
    array("consultar", "Consultar Empréstimos", "MA"),
    array("dadospessoais", "Alterar/Consultar Dados Cadastrais", "MA"),
    array("login", "Login", "I"),
    array("cadastrar", "Cadastrar-se", "I"),
    array("logout", "Sair", "MA"));

// Estrutura do menu de administração de livros
$menu_admin_livro = array (
    array("entregar", "Gerenciar Empréstimos", "A"),
    array("devolver", "Gerenciar Devoluções", "A"),
    array("cadastrarlivro", "Incluir novo Livro", "A"),
    array("removerlivro", "Remover livro", "A"),
    array("editarlivro", "Editar cadastro de livro", "A")
);
```

Abrir editor

menu.php (continuação)

```
// Estrutura do menu de administração de usuários.  
$menu_admin_usuario = array (  
    array("alterarusuario", "Alterar/Consultar Dados Cadastrais", "A") ,  
    array("autorizarusuario", "Autorizar Cadastramento", "A") ,  
    array("removerusuario", "Remover Usuário", "A")  
);
```

menu.php (continuação)

```
// Exibe um menu. Os itens só são exibidos se o usuário tem  
// privilégios para acessar o link.  
// $titulo = Título do menu.  
// $itens = Array de itens do menu, cada qual um array com elementos  
//         acao, rótulo e privilégio.  
function exibe_menu ($titulo,$itens) {  
    echo "<div class='menu'>";  
    echo "<div class='tituloMenu'>" . $titulo . "</div>\n";  
    echo "<div class='areaMenu'>\n";  
    foreach ($itens as $item) {  
        // Emitir itens para os quais usuário tem privilégio  
        list($acao,$rotulo,$privilegio) = $item;  
        if (strpos($privilegio,$_SESSION['privilegio']) !== false) {  
            echo "<a href='index.php?acao=$acao'>$rotulo.</a> <br> \n";  
        }  
    }  
    echo "</div>\n";  
    echo "</div>\n";  
}
```

menu.php (continuação)

```
// Exibe menus
exibe_menu ("Menu Principal", $menu_principal);
if ($_SESSION['privilegio']=='A') {
    exibe_menu ("Gerência de Livros", $menu_admin_livro);
    exibe_menu ("Gerência de Usuários", $menu_admin_usuario);
}
?>
```

principal.php

```
<?php  
// Executa uma ação dependendo do valor de $_GET['acao']  
switch ($_GET['acao']) {  
    case 'login':  
        include ('login.php');  
        break;  
    case 'loginok':  
        include ('loginok.php');  
        break;  
    case 'loginfalhou':  
        include ('loginfalhou.php');  
        break;  
    case 'logout':  
        include ('logout.php');  
        break;  
    case 'logoutok':  
        include ('logoutok.php');  
        break;  
    case 'cadastrar':  
        include ('cadastrar.php');  
        break;
```

Nota:

Onde a professora falou "\$acao = alguma coisa",
leia-se "acao=\$acao", onde \$acao é algum serviço
solicitado pelo usuário.

Abrir editor

principal.php (continuação)

```
case 'dadospessoais';
    include ('dadospessoais.php');
    break;
case 'buscar';
    include ('buscar.php');
    break;
case 'solicitar':
    include ('solicitar.php'); // mais acoes! Não estao aqui porque não cabem no slide :-
    break;
default:
    include ('default.php');
    break;
}
?>
<br><br>
```

Exemplo de classe: `solicitacao.php`

```
<?php
//
// Define classes e funcoes para gerenciar solicitacoes
//

require_once ('tabelabd.php');

// Representa uma linha da tabela solicitacao
class solicitacao extends tabela_bd
{
    // variaveis
    var $idlivro;      // Id de um livro
    var $idusuario;   // Id de um usuario
    var $hora;        // Hora em que foi feita a solicitação
```

[Abrir editor](#)

solicitacao.php (continuação)

Método entrega

```
// Registra a entrega deste livro ao usuário através da remoção deste registro  
// da tabela 'solicitacao' e da inserção de um registro correspondente  
// na tabela emprestimo.  
// Retorna false em caso de erro.  
  
function entrega ( ) {  
    // Computo a data de devolução baseada na data de hoje e  
    // na constante DIAS_POR_EMPRESTIMO  
    $devolucao = date ("Y-m-d", time() +DIAS_POR_EMPRESTIMO *24*60*60);  
    // removo e insiro  
    $resultado = mysql_query ("delete from solicitacao ".  
                             "where idlivro=$this->idlivro ".  
                             "and idusuario=$this->idusuario " .  
                             "and hora=' $this->hora' ");  
    if (!$resultado)  
        return false;  
    return mysql_query ("insert into emprestimo ".  
                        "values($this->idlivro, $this->idusuario, '$devolucao')" );  
}
```

solicitacao.php (continuação)

Método lugar_fila

```
// Retorna o lugar na fila desta solicitação. 0 significa
// que o livro está disponível
// e pode ser entregue para o usuário.
function lugar_fila () {
    $consulta = "select (select count(*) from solicitacao ".
        "      where idlivro = $this->idlivro ".
        "      and hora < '$this->hora') ".
        "  + (select count(*) from emprestimo ".
        "      where idlivro = $this->idlivro) ".
        "  - (select exemplares from livro ".
        "      where id = $this->idlivro)";

    $resultado = mysql_query ($consulta);
    $array = mysql_fetch_row($resultado);
    $lugar = $array [0]+1;
    if ($lugar<0)
        $lugar = 0;
    return $lugar;
}
```

solicitacao.php (continuação)

Método exibe

```
// Exibe uma solicitação. Primeiro obtém do bd os dados do livro e do usuario.
function exibe() {
    $livro = new livro_autores;
    $livro->busca_idlivro ($this->idlivro);
    $usuario = new usuario;
    $usuario->busca ("id = $this->idusuario");
    echo "<table class='exibe'>\n";
    echo "<tr><th rowspan=2>Usuário:<th>Nome<th>E-Mail</tr>\n";
    echo "<tr><td>$usuario->nome<td><a href='mailto:$usuario->email'>
        $usuario->email</a></tr>\n";
    echo "<tr><th rowspan=2>Livro:<th>Título<th>Autores</tr>\n";
    echo "<tr><td>".$livro->livro->titulo."<td>".implode("<br>\n",
        $livro->autores())."</tr>\n";
    echo "<tr><th>Hora da solicitação:<td colspan=2>";
    echo strftime ("%e/%b/%Y, %T", strtotime($this->hora));
    echo "</tr>\n";
    echo "<tr><th>Situação:<td colspan=2>";
    $lugar = $this->lugar_fila();
    echo ($lugar == 0) ? "Disponível para entrega" : "Em $lugar lugar na fila";
    echo "</tr>\n";
    echo "</table>\n";
}
} // fecha classe solicitacao
```

solicitacao.php (continuação)

Função de busca

```
// Realiza uma busca no banco de dados por uma ou mais solicitações dados  
// o nome/login de usuário e/ou título/autor de livro.  
// Retorna um array de objetos do tipo solicitacao que satisfazem os critérios.  
// $nome = nome de usuário solicitante.  
// $login= login de usuário solicitante.  
// $titulo= titulo de livro solicitado.  
// $autor = nome de autor de livro solicitado.  
function busca_solicitacao ($nome='', $login='', $titulo='', $autor='')  
{  
    // Variáveis string são inicializados com consulta básica:  
    // Obter todas as solicitações para as quais existe um livro disponível.  
    $select = "select sol.idlivro,sol.idusuario,sol.hora ";  
    $from = "from solicitacao sol,livro ";  
    $where = "where sol.idlivro = livro.id ".  
        "and livro.exemplares > ".  
        " (select count(*) from emprestimo ".  
        " where emprestimo.idlivro = sol.idlivro) +".  
        " (select count(*) from solicitacao sol2 ".  
        " where sol2.idlivro = sol.idlivro ".  
        " and sol2.hora < sol.hora) ";
```

solicitacao.php (continuação)

Função de busca

```
// Critérios envolvendo usuário
if ($nome != '' || $login != '') {
    // Incluo critérios para busca por usuário
    $from .= ",usuario ";
    $where .= "and sol.idusuario=usuario.id ";
    if ($nome != '') {
        $where .= "and usuario.nome like '%$nome%' ";
    }
    if ($login != '') {
        $where .= "and usuario.login='$login'";
    }
}

// Critérios envolvendo livro
if ($titulo != '' || $autor != '') {
    // incluo critérios para busca por livro
    if ($titulo != '') {
        $where .= "and titulo like '%$titulo%' ";
    }
    if ($autor != '') {
        $from .= ",escreveu,autor ";
        $where .= "and autor.id = escreveu.idautor ". "and livro.id = escreveu.idlivro ".
            "and autor.nome like '%$autor%' ";
    }
}
```

solicitacao.php (continuação)

Função de busca

```
// Faço a consulta ordenando os resultados
// pela hora da solicitação

$resultado = mysql_query ($select.$from.$where." order by hora");
$solicitacoes = array();
while ($array = mysql_fetch_array ($resultado)) {
    $solicitacao = new solicitacao;
    $solicitacao->atribui_de_array($array);
    $solicitacoes[] = $solicitacao;
}
return $solicitacoes;
} // termina função de busca

?> // termina solicitacao.php
```

Exemplo de form: formsolicitacao.php

```
<?php
require_once ('htmlutil.php');

// Exibe um formulário para busca de solicitações.
// $acao = url que processa o formulário.
// $default = array com defaults para os campos.
function form_busca_solicitacao ($acao, $default) {
?>
<form method="post" action="<?=$acao?>" name="formbuscasolicitacao">
<table class="formulario">
  <tbody>
    <tr> <th colspan=2>Dados do Usuário</th>
    </tr>
    <tr> <th>Nome</th>
      <td>
        <input maxlength="80" size="60" value="<?=$usuario->nome?>" name="nome"></td>
      </td>
    </tr>
    <tr> <th>Login</th>
      <td>
        <input name="login" value="<?=$usuario->login?>"></td>
      </td>
    </tr>
```

[Abrir editor](#)

Exemplo de form: formsolicitacao.php (continuação)

```
<tr> <th colspan=2>Dados do Livro</th>
</tr>
<tr> <th>Título</th>
    <td><input maxlength="120" size="50" name="titulo" value="<?=$default['titulo']?>"></td>
</tr>
<tr> <th>Autor</th>
    <td><input maxlength="100" size="50" name="autor" value="<?=$default['autor']?>"></td>
</tr>
</tbody>
</table>
<button name="enviar_busca_solicitacao">Buscar</button><br>
</form>
<?php
}
```

Exemplo de form: **formsolicitacao.php** (continuação)

```
// Exibe um formulário solicitações e um checkbox de
// confirmação para cada solicitação.
// $solicitacoes = array com objetos da classe 'solicitacao'
function form_confirm_entrega ($acao,&$solicitacoes) {
    echo "<form method='post' action='".$acao'
          name='formconfirmasolicitacao'>\n";
    echo "<table class='formulario'>\n";
    foreach ($solicitacoes as $indice=>$solicitacao) {
        echo "<tr><td><input name='".$indice'
              type='checkbox'></td>\n";
        echo "<td>";
        $solicitacao->exibe();
        echo "</td></tr>\n";
    }
    echo "</table>\n";
    echo "<button name='enviar_confirm_entrega'>Confirmar</button><br>\n";
    echo "</form>\n";
}?
```

Exemplo de ação: `solicitar.php`

```
<h2> Solicitação de Livro</h2>
<?php
// O livro cujo id é $_GET['id'] foi clicado numa busca e o usuário
// deve ser inquirido se quer tomá-lo emprestado ou reservá-lo.
require_once('livro.php');
require_once('htmlutil.php');
require_once('solicitacao.php');

assegura_privilegio ('M');

// Obtém livro solicitado
$livro_autores = new livro_autores;
$idlivro = $_GET['id'];
if ($livro_autores->busca_idlivro($idlivro)) {
    $livro_autores->exibe();
    $disponiveis = $livro_autores->livro->exemplares_disponiveis();
    if (isset($_GET['confirma'])) {
        $solicitacao=new solicitacao;
        $solicitacao->idlivro = $idlivro;
        $solicitacao->idusuario = $_SESSION['idusuario'];
        $solicitacao->hora = strftime('%Y-%m-%d %H:%M:%S');
        if (!$solicitacao->inclui()) {
            echo mysql_error();
            exit;
        }
    }
}
```

[Abrir editor](#)

Exemplo de ação: `solicitar.php`

(continuação)

```

echo "<p> <b>Solicitacao Confirmada</b> </p>\n";
if ($disponiveis > 0) {
    echo "<p>Dirija-se ao balcão para retirar o livro</p>\n";
} else {
    echo "<p>Enviaremos um e-mail para você quando um
    exemplar deste livro se tornar disponível</p>\n";
}
}

else {
    if ($disponiveis > 0) {
        echo "<p>Livro tem $disponiveis exemplares disponíveis. ".
        "Deseja solicitar um exemplar?</p>";
    } else {
        echo "<p>Livro não tem nenhum exemplar disponível. ".
        "Deseja realizar uma reserva?</p>";
    }
    botao ("Sim", "index.php?acao=solicitar&confirma=1&id=$idlivro");
    botao ("Não, Voltar", "javascript:history.back();");
}
}

else {
    msg_erro ("Livro não encontrado");
}

link_pagina_principal ();
?>

```

processallogin.php

```
<?php
// Processamento do formulário de login
require_once ('sessao.php');
require_once ('autenticacao.php');
$usuario = autentica ($_POST['login'], $_POST['senha']);
if ($usuario) {
    // Autenticação bem-sucedida -> redirecionar para página de boas vindas
    $_SESSION['login']=$usuario->login; ←
    $_SESSION['idusuario']=$usuario->id; ←
    $_SESSION['privilegio']=$usuario->privilegio; ←
    header("Location: index.php?acao=loginok");
} else {
    // Autenticação falhou -> redirecionar para página com mensagem de erro
    $_SESSION['privilegio']='I'; ←
    header("Location: index.php?acao=loginfalhou");
}
?>
```

Abrir editor

Guarda info do usuário desta sessão

processlogout.php

```
<?php
// Processamento do formulário de confirmação de logout
require_once ('sessao.php');
if ($_POST["sair"]=="Sair") {
    // Usuário confirmou logout -> exibir mensagem de adeus
    session_destroy();
    header("Location: index.php?acao=logoutok");
}
else {
    // Usuário não confirmou -> redirecionar para página principal
    header("Location: index.php");
}
?>
```

[Abrir editor](#)

Outros arquivos

- Além dos arquivos da aplicação, estão disponíveis outros arquivos:
 - ▶ **incluirusuario.php**, utilizado para incluir usuários na biblioteca (por exemplo, foi utilizado para inicializar o banco de dados com uma entrada para o administrador) [Abrir editor](#)
 - ▶ **biblioteca.sql**, com os comandos para inicialização do banco de dados com a tabela de livros [Abrir editor](#)
 - ▶ **crialivros.php**, para preencher as tabelas livro, autor e escreveu com os dados da tabela tablivros [Abrir editor](#)

Melhoramentos

- Envio automático de email nas situações:
 - ▶ Usuário foi autorizado pelo administrador e se tornou membro da biblioteca
 - ▶ Usuário não devolveu livro no prazo determinado (máximo 14 dias)
 - ▶ Livro reservado está disponível
- Validação mais precisa do email digitado pelo usuário (no momento não há teste para verificação se o email e domínio existem)
- Não há código para cancelamento de solicitações, quando o usuário nunca vai buscar o livro
- Outros que vocês possam pensar!

Executando a aplicação

- Clique no botão abaixo para executar a aplicação em seu navegador.

Abrir navegador