



Universidade Federal de Uberlândia
Faculdade de Computação - Prof. Daniel A. Furtado
10º Trabalho de Programação para Internet – Gestão da Informação
Introdução ao Desenvolvimento Web com Banco de Dados

Instruções Gerais

- Esta atividade deve ser realizada individualmente;
- Utilize apenas as tecnologias HTML5, CSS, JavaScript, Bootstrap 5, PHP e MySQL;
- Sintaxe da XHTML como `` ou `
` não é permitida (anulará o trabalho);
- O website deve ser hospedado e disponibilizado online, conforme orientações disponíveis no final deste documento;
- Ao construir o website, utilize dados fictícios (**jamais utilize** dados pessoais como seu nome, CPF, endereço, e-mail etc.);
- Esteja atento às **observações sobre plágio** apresentadas no final deste documento;
- Trabalhos com implementações utilizando trechos de códigos retirados de sites da Internet ou de trabalhos de semestres anteriores serão anulados;
- As páginas web não devem conter qualquer conteúdo de caráter imoral, desrespeitoso, pornográfico, discurso de ódio, desacato etc.;
- O website deve ser validado utilizando as ferramentas disponíveis nos endereços **validator.w3.org** e **jigsaw.w3.org/css-validator** (não deve conter nenhum erro ou *warning*);
- O trabalho deve ser entregue até a data/hora definida pelo professor. Não deixe para enviar o trabalho nos últimos instantes, pois eventuais problemas relacionados à eventos adversos como instabilidade de conexão, congestionamento de rede etc., não serão aceitos como motivos para entrega da atividade por outras formas ou em outras datas;
- Este trabalho deve ser feito **mantendo os trabalhos anteriores intactos**, ou seja, os trabalhos anteriores devem permanecer online conforme foram entregues;
- Trabalhos enviados por e-mail ou pelo MS Teams **não serão considerados**.

Material de Apoio e Dicas Gerais

<https://furtado.prof.ufu.br/site/teaching/PPI/PPI-Modulo7-Banco-de-Dados.pdf>

Para visualizar eventuais erros do MySQL/PHP, acesse sua conta do infinityfree e configure:

Accounts → if0_xxx → Control Painel → Software → Alter PHP Config → Alter PHP Directives → Display Errors → ON

Display Errors	<input type="radio"/> Off <input checked="" type="radio"/> On
MB String Input	<input type="text" value="auto"/>
PHP Timezone	<input type="text" value="America/New_York"/>
<input type="button" value="Alter PHP directives"/>	

Exercício 1

Descompacte o arquivo <https://furtado.prof.ufu.br/site/teaching/PPI/exemplos/Exemplos-MySQL-GGI.zip> e coloque os exemplos online seguindo os passos a seguir:

1. Crie um novo banco de dados no **infinityfree** conforme instruções do **slide 5**;
2. Siga os passos do **slide 7** para executar o código SQL disponibilizado no arquivo **sql-tabelas.sql** (veja pasta raiz dos exemplos). Observe que o código SQL cria as tabelas **aluno** e **cliente** e insere dois registros na tabela **aluno**;
3. Abra o arquivo **conexaoMysql.php** (na pasta raiz) e altere os dados de conexão inserindo os dados do seu banco de dados (veja figura do **slide 6**);
4. Copie a pasta raiz contendo todos os exemplos para o servidor. O exemplo deve ficar disponível no endereço **seudominio.com/trabalho10**. Neste trabalho não é necessário criar subpastas para cada exercício;
5. Ao acessar a pasta raiz dos exemplos no navegador deverá abrir um **menu de opções**. Abra o **Exemplo 1 – Hello World** e verifique se os dados dos alunos estão sendo listados adequadamente;
6. Abra o arquivo **index.php** do **Exemplo 1** e acrescente comentários no código PHP explicando as operações.

Exercício 2

1. Acesse o **Exemplo 2 - Cadastro de aluno (vulnerável a ataques)**, cadastre dois novos alunos e verifique os dados listados;
2. O código PHP do exemplo está vulnerável a ataques do tipo **SQL Injection**. Simule um ataque cadastrando um novo aluno e inserindo a string a seguir no campo **telefone** do formulário:
tol'); DELETE FROM aluno; -- comment
3. Acrescente comentários no código PHP do arquivo **cadastro-vulneravel.php** explicando o motivo da vulnerabilidade. Não utilize explicitamente na explicação as palavras Injeção, Injection, Ataques, XSS etc.;

Exercício 3

1. Acesse o **Exemplo 3 - Cadastro de Clientes com Prepared Statements**, cadastre dois novos clientes e verifique os dados listados;
2. Na página de listagem de clientes, exclua um cliente utilizando o botão “X” que aparece na linha dos dados do cliente;
3. Abra o exemplo no **VS Code** e analise o código dos arquivos:
 - a. index.html
 - b. cadastra-cliente.php
 - c. exclui-cliente.php
 - d. mostra-clientes.php
4. Volte no arquivo **cadastro-vulneravel.php** do exercício 2, coloque o código vulnerável dentro de um **comentário de bloco** e acrescente o código adequado utilizando **prepared statements** para corrigir a vulnerabilidade. Em seguida, cadastre novos alunos e repita o ataque de injeção para verificar se o problema foi resolvido.

Exercício 4

- a) Crie uma tabela no banco de dados para armazenar os dados dos funcionários do sistema para gestão da clínica médica (projeto da disciplina). Em seguida, crie o formulário HTML e o respectivo script PHP para permitir que os dados sejam cadastrados na tabela criada. Dicas:
- O código do funcionário não deve ser solicitado ao usuário (**auto_increment**);
 - A senha do funcionário não deve ser armazenada (armazene apenas o código hash);
 - Abra o **phpMyAdmin** no **infinityfree** e verifique se os dados estão sendo cadastrados;
 - O código deve prevenir ataques de injeção de SQL.

Funcionario
<u>Codigo</u>
Nome
Email
Senhahash
EstadoCivil
DataNascimento
Funcao

b) Crie um script PHP para listar os dados de todos os funcionários cadastrados. O script deve produzir uma página HTML dinâmica completa e os dados devem ser listados utilizando uma tabela HTML. Ao finalizar, altere o script de cadastro (item a) para que o usuário seja direcionado automaticamente para a página de listagem de dados (após o cadastro).

Disponibilização Online

As páginas dos exemplos modificadas devem ser disponibilizadas online utilizando o subdomínio gratuito registrado anteriormente, porém em pasta própria. Não altere ou exclua as pastas dos trabalhos anteriores.

Entrega

Além da disponibilização online, a pasta raiz contendo os exemplos atualizados (com comentários, explicações, correções etc.) exemplos deve ser compactada no formato zip e enviada pelo Sistema Acadêmico de Aplicação de Testes (SAAT) até a data limite indicada pelo professor em sala de aula.

Adicione também um arquivo de nome **link.txt**, na pasta raiz, contendo a URL do trabalho online (para a pasta raiz do trabalho).

Sobre Eventuais Plágios

Este é um trabalho individual. Os alunos envolvidos em qualquer tipo de plágio, total ou parcial, seja entre equipes ou de trabalhos de semestres anteriores ou de materiais disponíveis na Internet (exceto os materiais de aula disponibilizados pelo professor), serão duramente penalizados (art. 196 do Regimento Geral da UFU). Todos os alunos envolvidos terão seus **trabalhos anulados** e receberão **nota zero**.