



Flask

Programação Web III

Prof. Diego Max

Flask

Tópico 04.2: Cadastrando dados

CRUD com o banco de dados MySQL

Cadastrando dados no banco



Criaremos agora um formulário de cadastro para inserir novos registros no banco de dados.

Para isso, no arquivo “**estoque.html**” insira o código responsável pela criação do formulário conforme a seguir:

```
<h2>Controle de estoque</h2><br>
<details>
  <summary>[ Cadastrar novo jogo ]</summary>
  <div class="col-md-4">
    <form action="{{url_for('estoque')}}" method="POST">
      <input type="text" name="titulo" placeholder="Título" class="form-control" required>
      <input type="number" name="ano" placeholder="Ano" min="1970" max="2099" class="form-control" required>
      <input type="text" name="categoria" placeholder="Categoria" class="form-control" required>
      <input type="text" name="plataforma" placeholder="Plataforma" class="form-control" required>
      <input type="number" name="preco" placeholder="Preço" min=0 step="0.01" class="form-control" required>
      <input type="number" name="quantidade" placeholder="Quantidade" class="form-control" required><br>
      <input type="submit" value="Cadastrar" class="btn btn-success">
      <input type="reset" value="Limpar" class="btn btn-danger">
    </form>
  </div>
</details>
<hr>
```


Cadastrando dados no banco



Feito isso, no arquivo **“routes.py”**, iremos inserir uma **condição** para verificar **se o método da requisição** que está chegando até rota é do tipo **POST**, em caso positivo, as informações enviadas pelo formulário serão gravadas em suas respectivas variáveis.

Por fim, utilizaremos os métodos **“db.session.add()”** para inserir um novo registro no banco e o **“db.session.commit()”** para validar essas alterações de fato.

```
# CRUD - LISTAGEM E CADASTRO
@app.route('/estoque', methods=['GET', 'POST'])
def estoque():
    # Cadastra um novo jogo
    if request.method == 'POST':
        newgame = Game(request.form['titulo'], request.form['ano'], request.form['categoria'],
request.form['plataforma'], request.form['preco'], request.form['quantidade'])
        db.session.add(newgame)
        db.session.commit()
        return redirect(url_for('estoque'))
    else:
        # Armazena em "gamesestoque" todos os valores e encaminha para estoque.html
        gamesestoque = Game.query.all()
        return render_template('estoque.html', gamesestoque=gamesestoque)
```

Cadastrando dados no banco



Agora basta inserirmos as informações de um novo game no formulário e enviarmos para o banco de dados:

Controle de estoque

▼ [Cadastrar novo jogo]

Flask

Tópico 04.3: Deletando dados

CRUD com o banco de dados MySQL

Deletando dados do banco



Uma vez que já estamos conseguindo cadastrar dados no banco, agora iremos possibilitar a **exclusão** desses dados. Para isso, no arquivo “**estoque.html**” iremos incluir em nossa tabela HTML uma nova coluna com as ações **Excluir** e **Editar** (que implementaremos futuramente). Nesse link será passado o **ID do game** para a **rota estoque**.

```
<th>Quantidade:</th>  
<th style="text-align: center;" colspan="2">Ação:</th>  
</tr>  
</thead>  
<tbody>  
{% for g in gamesestoque %}  
  <tr>
```

```
  <tr>
```

```
    <td>{{g.titulo}}</td>
```

```
    <td>{{g.ano}}</td>
```

```
    <td>{{g.categoria}}</td>
```

```
    <td>{{g.plataforma}}</td>
```

```
    <td>{{g.preco}}</td>
```

```
    <td>{{g.quantidade}}</td>
```

```
    <td><a style="text-decoration: none; color: orange;" href="#">Editar</a></td>
```

```
    <td><a style="text-decoration: none; color: red;" href="{{url_for('estoque',  
id=g.id)}}" onclick="return confirm('Deseja mesmo excluir o jogo {{g.titulo}}?')">Excluir</a></td>
```

```
  </tr>
```

```
{% endfor %}
```

	Quantidade:	Ação:	
	10	Editar	Excluir

Deletando dados do banco



Agora no arquivo “**routes.py**”, iremos adicionar uma nova rota “**/estoque/delete**” que receberá como **parâmetro** a **ID** do game que deseja-se deletar. Esse game será buscado no banco através do método “**Game.query.get(id)**”, depois incluiremos o método para excluir o jogo conforme a ID:

```
# CRUD - LISTAGEM, CADASTRO E EXCLUSÃO
@app.route('/estoque', methods=['GET', 'POST'])
@app.route('/estoque/delete/<int:id>')
def estoque(id=None):
    if id:
        game = Game.query.get(id)
        # Deleta o cadastro pela ID
        db.session.delete(game)
        db.session.commit()
        return redirect(url_for('estoque'))
    # Cadastra um novo jogo
```

Para excluir um registro do banco utilizamos o método “**db.session.delete()**”.

Deletando dados do banco



Ao tentar excluir o game uma mensagem de confirmação é exibida através do `confirm()` do Javascript.

[The Games](#) [Home](#) [Games](#) [Cadastrar g](#)

Deseja mesmo excluir o jogo ?
OK Cancelar

The Games

Controle de estoque

► [Cadastrar novo jogo]

Jogos em estoque:

Título:	Ano:	Categoria:	Plataforma:	Preço:	Quantidade:	Ação:	
	0			0	0	Editar	Excluir
Minecraft	2009	Sandbox	PC (Windows)	100	10	Editar	Excluir

Feito a confirmação o registro **será deletado** do banco de dados.



Flask

Programação Web III

Prof. Diego Max