

Integração de API com WooCommerce

Integrantes:

- Antônio Carlos Durães
- Nicolas Sampaio

A proposta deste repositório é o desenvolvimento de uma Interface de Programação de Aplicações (API) integrada ao banco de dados do Sistema de Gestão de Conteúdo (CMS) Wordpress e seu plugin para e-commerce / lojas virtuais, WooCommerce.

Tecnologias:

- Python 3.7 (Linguagem de programação escolhida)
- Flask 1.1.2 (Framework web p/ construir API)
- PyMySQL 0.10.0 (Driver para conexão e operações c/ MySQL)
- inquirer 2.7.0 (Biblioteca p/ construção de menu em lista)

Como configurar o projeto

Instalação de dependências

Após baixar e instalar o [Python 3.5+](#), siga o seguinte algoritmo para instalar suas dependências:

- Abra o terminal, Powershell ou CMD na pasta "src" do projeto;
- Digite o comando "*pip install -r requirements.txt*" (substitua "pip" por "pip3" se seu S.O. for Linux) e dê ENTER

Conexão com Banco de Dados

É necessário ter um banco rodando com as seguintes configurações ou alterar o arquivo "config_bd.py" para se adequar ao seu banco MySQL.

Configurações padrão do Banco de Dados:

```
{
    "host": "127.0.0.1",
    "port": 3306,
    "user": "root",
    "passwd": "",
    "database": "loja"
}
```

⚡ Como executar o projeto

Executar API

- Abra o terminal, Powershell ou CMD na pasta "src" do projeto;
- Digite o comando "python api"* e dê ENTER

Executar cliente

- Abra um **novo** terminal, Powershell ou CMD na pasta "src" do projeto;
- Digite o comando "python cliente"* e dê ENTER

*Obs.: Substitua "python" por "python3" se seu S.O. for Linux

📖 Descrição das tabelas e queries

As tabelas utilizadas foram a "wp_posts" (responsável por conter as colunas relativas ao ID do post, nome, conteúdo e tipo) e "wp_postmeta" (contém o preço, informações de estoque e outras relacionadas).

Exclusão de produto

Remoção de um produto com um ID específico.

```
-- Remova a postagem (com título e descrição) do produto especificado
DELETE FROM wp_posts WHERE ID = {id};

-- Remova as linhas relativas ao preço, quantidade e controle de estoque do produto
DELETE FROM wp_postmeta WHERE post_id = {id};
```

Listagem de produtos

Listagem de todos produtos. Notar que o preço do produto é salvo como uma linha da tabela "wp_postmeta" e a quantidade em uma nova linha, com atributos "meta_key" diferentes.

```
-- Selecione o ID, nome, descrição, preço e quantidade do produto
SELECT
    tb_post.ID AS id,
    tb_post.post_title AS nome,
    tb_post.post_content AS descricao,
    tb_preco.meta_value AS preco,
    tb_qtd.meta_value AS quantidade
FROM wp_posts AS tb_post

-- Selecione o preço do produto com ID recebido
```

```

INNER JOIN
(SELECT meta_value, post_id FROM wp_postmeta WHERE meta_key = '_price')
AS tb_preco ON tb_preco.post_id = ID

-- Selecione a quantidade do produto com ID recebido
INNER JOIN
(SELECT meta_value, post_id FROM wp_postmeta WHERE meta_key = '_stock')
AS tb_qtd ON tb_qtd.post_id = ID

-- Filtre somente os posts do tipo produto
WHERE post_type = 'product'

```

Busca de um produto

```

-- Select igual ao usado na listagem de produtos, porém com a cláusula WHERE
... AND WHERE ID = {id}

```

Registro de produto

```

/* Insira o nome e a descrição recebida na tabela de postagens do wordpress
marcada como uma postagem de produto */
INSERT INTO wp_posts (post_title, post_content, post_type)
VALUES ('{nome}', '{descricao}', 'product');

/* Insira o preço e a quantidade do produto, e habilite a gerência de estoque
passando o ID do post criado */
INSERT INTO wp_postmeta (post_id, meta_key, meta_value)
VALUES
({post_id}, '_price', {preco}),
({post_id}, '_stock', {quantidade}),
({post_id}, '_manage_stock', 'yes');

```

Atualização de produto

```

-- Atualize o título e a descrição do produto de ID recebido
UPDATE wp_posts
SET post_title = '{nome}', post_content = '{descricao}'
WHERE ID = {post_id};

-- Atualize a quantidade do produto de ID recebido
UPDATE wp_postmeta
SET meta_value = {quantidade}
WHERE post_id = {post_id} AND meta_key = '_stock';

-- Atualize o preço do produto de ID recebido
UPDATE wp_postmeta
SET meta_value = {preco}
WHERE post_id = {post_id} AND meta_key = '_price';

```

Obter estatísticas

```
-- Selecione o total de unid. vendidas, o preço total e calcule a média por produto
SELECT
    tb_total.qtd AS qtd_total_produtos,
    tb_total.preco AS valor_total_produtos,
    tb_total.preco / tb_total.qtd AS valor_medio_produtos
FROM
(
    -- Calcule a soma de unidades vendidas e a soma do preço de todos produtos
    SELECT
        SUM(pedido.product_qty) AS qtd,
        SUM(pedido.product_gross_revenue) AS preco
    FROM wp_wc_order_product_lookup AS pedido
) AS tb_total
```