



Centro Universitário de Excelência
Sistemas de Informação

Sistema de Gerenciamento de Pedidos

Autor: Manuela de Oliveira Figueira
Lorena Dias Freitas
Rafaela Canguçu Souza
Samuel Nascimento Santos
Joice da Rocha Silveira

Vitória da Conquista, 2025

Objetivo da Apresentação



- Simular o funcionamento básico de um restaurante
- Utilizar Python e estruturas de dados nativas
- Gerenciar itens e pedidos via menu interativo

Estrutura Geral



- Menu interativo em linha de comando
- Funções para itens e pedidos
- Uso de filas (FIFO) para organizar pedidos

Gerenciamento De Itens

Objetivo Principal:
Um Sistema em menu interativo (console) para gerenciar itens e pedidos

1. Cadastrar Item
Adiciona um novo produto ao menu.

2. Atualizar Item:
Modifica informações de um item existente

3. Consultar Itens
Exibe todos os itens disponíveis.

4. Dados do item
código, nome, descrição, preço e estoque

```
1 - Cadastrar item
2 - Consultar itens
0 - Sair
Escolha: 1

Cadastrar novo item do cardápio
Nome do item: Pão de forma
Descrição: integral
Preço: R$ 6,50
Digite um valor numérico válido para o preço.
Preço: R$ 5.50
Quantidade em estoque: 20
Item cadastrado com sucesso!

1 - Cadastrar item
2 - Consultar itens
0 - Sair
Escolha: 1

Cadastrar novo item do cardápio
Nome do item: coca cola
Descrição: zero acucar
Preço: R$ 10
Quantidade em estoque: 50
Item cadastrado com sucesso!

1 - Cadastrar item
2 - Consultar itens
0 - Sair
Escolha: 2

--- Cardápio ---
Código: 1, Nome: Pão de forma, Descrição: integral, Preço: R$5.5, Estoque: 20
Código: 2, Nome: coca cola, Descrição: zero acucar, Preço: R$10.0, Estoque: 50
```

```
1 - Atualizar item
2 - Consultar itens
3 - Cadastrar item
0 - Sair
Escolha: 1

Itens disponíveis:
Código: 1, Nome: lasanha, Preço: R$20.0, Estoque: 10
Digite o código do item que deseja atualizar: 1
Atualizando item: lasanha
Novo nome (pressione enter para manter): lasanha bolonhesa
Nova descrição (pressione enter para manter): massa, molho branco
Novo preço (pressione enter para manter): 20
Novo estoque (pressione enter para manter): 10
Item atualizado com sucesso!
```

Gerenciamento De Pedidos

Objetivo:
Criar, processar, atualizar ou consultar pedidos conforme regras e status.

1 Criar Pedido:
Adiciona um novo pedido ao sistema.

2. Processar Pedidos Pendentes
Modifica o status de um pedido que está AGUARDANDO APROVAÇÃO para ACEITO ou REJEITADO.

3. Atualizar Status de Pedido:
Atualiza status dos pedidos de acordo com suas posições na fila.

4 Consultar Pedidos:
Apresenta a lista de pedidos em aberto.

```
1 - Criar pedido
2 - Ver pedidos pendentes
0 - Sair
Escolha: 1

Itens disponíveis:
Código: 1, Nome: Pão de forma, Preço: R$5.5, Estoque: 20
Código: 2, Nome: coca cola, Preço: R$10.0, Estoque: 50
Digite o código do item que deseja adicionar: 2
Item 'coca cola' adicionado ao pedido.
Deseja adicionar outro item? (s/n): n
Deseja aplicar cupom de desconto (%)? Caso não, pressione enter:

Pedido 2 criado com sucesso! Total: R$10.00
```

```
1 - Criar pedido
2 - Ver pedidos pendentes
0 - Sair
Escolha: 2
Código: 1, Total: R$13.95, Status: AGUARDANDO APROVACAO
Código: 2, Total: R$10.00, Status: AGUARDANDO APROVACAO
```

Fluxo de Pedidos e Filas



- AGUARDANDO APROVAÇÃO→ todos pedidos que forem criados
- Fila de pedidos pendentes→ pedidos em análise
- ACEITO→ fila de pedidos aceitos

```
1 - Processar pedidos pendentes
2 - Ver pedidos aceitos
0 - Sair
Escolha: 1

Pedido 1 - Total: R$13.95 - Status: AGUARDANDO APROVACAO
Aceitar ou Rejeitar o pedido? (a/r): r
Pedido 1 REJEITADO.

Pedido 2 - Total: R$10.00 - Status: AGUARDANDO APROVACAO
Aceitar ou Rejeitar o pedido? (a/r): a
Pedido 2 ACEITO.

1 - Processar pedidos pendentes
2 - Ver pedidos aceitos
0 - Sair
Escolha: 2
Código: 2, Total: R$10.00, Status: ACEITO
```


Consultas



O sistema permite:

- Exibir todos os pedidos.
- Filtrar pedidos por status.

```
182 # Parte 5: Atualizar Status e Consultar Pedidos
183 fila_pedidos_prontos = []
184 fila_pedidos_entregues = []
185
186 ▼ def print_pedido(pedido):
187     print(f"\nCódigo: {pedido[0]}")
188     print(f"Status: {pedido[3]}")
189     print(f"Total: R${pedido[2]:.2f}")
190     print("Itens:")
191     for item in pedido[1]:
192         print(f"- {item[1]} (R${item[3]}, Estoque: {item[4]})")
193
194 ▼ def atualizar_status_pedido():
195     if not fila_pedidos_aceitos and not fila_pedidos_prontos:
196         print("Não há pedidos em andamento.")
197         return
198
199     todos_pedidos = fila_pedidos_aceitos + fila_pedidos_prontos
200     for pedido in todos_pedidos:
201         print(f"Código: {pedido[0]}, Status: {pedido[3]}, Total: R${pedido[2]:.2f}")
```

Considerações finais



- Sistema atendeu ao objetivo proposto
- Aplicação prática de conceitos de estruturas de dados
- Simulação clara do fluxo de pedidos em um restaurante
- Possibilidade de futuras melhorias (ex.: interface gráfica, banco de dados)

Ferramentas utilizadas



- Python
- VS Code
- Git & GitHub

Referências



- Materiais e orientações fornecidos pelo professor