

Disciplina: Linguagem de Programação I

Semestre: 2023.1

Professore: Derzu Omaia

Projeto de Programação

INSTRUÇÕES:

1. **O projeto é individual, dupla ou trio.**

Temas:

- | | | |
|----------------------------|--------------------------|-------------------------|
| 1. Supermercado | 18. Loja Roupas | 36. Cartório |
| 2. Mercado | 19. Loja Genérica | 37. Imóveis/Imobiliária |
| 3. Banca Revista | 20. Distribuidora | 38. Vendedor Autônomo |
| 4. Lanchonete | 21. Artesanato | 39. Assessoria Geral |
| 5. Doceria | 22. Artista Plástico | 40. Teatro |
| 6. Churrascaria | 23. Controle estoque | 41. Agricultura. |
| 7. Churrasquinho | 24. Farmácia | 42. Pecuária |
| 8. Sanduicheria | 25. Consertos Gerais | 43. Serigrafia |
| 9. Restaurante | 26. Marcenaria | 44. Floricultura |
| 10. Bar | 27. Oficina | 45. Taxi |
| 11. Quentinha/Marmita | 28. Equipadora de Carros | 46. Escola |
| 12. Material de construção | 29. Costura | 47. Cabelereiro |
| 13. Construtora | 30. Entrega de água | 48. Manicure |
| 14. Consultório | 31. Entrega de gás | 49. Bodega |
| 15. Clínica | 32. Personal Trainer | 50. Revendedor |
| 16. Livraria | 33. Advocacia | Avon/Jequiti/Natura |
| 17. Papelaria | 34. Bijuteria | 51. Outros |
| | 35. Cosméticos | |

Especificações:

1. Criar um sistema CRUD para cadastro de estoque/clientes ou realizar vendas relacionado a um dos temas sugeridos. O Menu deve possuir no mínimo essas opções:
 1. Inserir
 2. Listar todos
 3. Exibir um (pesquisar por nome ou por índice)
 4. Alterar (pesquisar por nome ou por índice)
 5. Remover (pesquisar por nome ou por índice)
 6. Exibir Relatório
 7. Sair

Obs: faça um método para pesquisar por nome ou índice, e reuse ele nas opções, exibir um, alterar, remover. Este método deve retornar o endereço do objeto, e deve receber como parâmetro o string do nome e o índice a ser buscado. Sugestão de assinatura do método:

`Objeto * Pesquisar(std::string nome, int indice);`

2. Faça a modelagem das classes que serão utilizadas no sistema utilizando diagrama UML de classe.
3. O objeto principal deve ter pelo menos 4 atributos, entre eles um que seja do tipo de outra classe criada pela equipe (composição). Ex.: Endereço, Data, etc.
4. Utilize pelo menos **herança** com pelo menos 1 superclasse e 2 subclasses.
5. Utilize uma lista/vetor com **polimorfismo**. Implemente pelo menos um método que sobrescreva o algum método da superclasse.

6. Utilize uma classe que vai gerenciar as operações CRUD. Não faça na main();
7. Usar bastante métodos.
8. Gerar relatório com um resumo das informações. Ex: Relatório de vendas, relatório de estoque, relatório de clientes. No relatório devem ser exibidas informações como, quantidade de elementos cadastrados, valor total, entre outros.
9. O programa deve rodar indefinidamente, até que o usuário escolha sair.
10. Todos os dados armazenados no programa devem ser salvos em arquivo, para que ao sair do programa os dados não se percam. E ao abrir o programa novamente os dados salvos devem ser carregados.

Carregar o arquivo automaticamente ao entrar na aplicação. Salvar em arquivo ao finalizar o programa. Todas as operações de inserção, exibição, edição e remoção deve ser realizada no vetor de objetos, e não diretamente no arquivo.

Entrega:

- Apresentação rápida (até 10min) durante a aula;
- Entregar código fonte (arquivo zip ou link do repositório) e Diagrama de Classes;
- Data: 25 e 26/09/2023.