

CURSO DE BACHARELADO EM SISTEMAS DE INFORMAÇÃO – 3º PERÍODO
DISCIPLINA INTRODUÇÃO À PROGRAMAÇÃO ORIENTADA A OBJETOS

LISTA DE EXERCÍCIOS 03 – CONCEITOS BÁSICOS DA ORIENTAÇÃO A OBJETOS
(PARTE II)

1. Implemente em Java o código-fonte da **Classe Retangulo** com seus respectivos atributos e operações, mantendo total compatibilidade com a modelagem. Ao final, codifique a **Classe TesteRetangulo**, onde deve-se instanciar a Classe Retangulo e testar o seu funcionamento.
2. Implemente em Java o código-fonte da **Classe Quadrado** com seus respectivos atributos e operações, mantendo total compatibilidade com a modelagem. Ao final, codifique a **Classe TesteQuadrado**, onde deve-se instanciar a Classe Quadrado e testar o seu funcionamento.
3. Implemente em Java o código-fonte da **Classe Circulo** com seus respectivos atributos e operações, mantendo total compatibilidade com a modelagem. Ao final, codifique a **Classe TesteCirculo**, onde deve-se instanciar a Classe Circulo e testar o seu funcionamento.
4. Implemente em Java o código-fonte da **Classe Triangulo** com seus respectivos atributos e operações, mantendo total compatibilidade com a modelagem. Ao final, codifique a **Classe TesteTriangulo**, onde deve-se instanciar a Classe Triangulo e testar o seu funcionamento.
5. Implemente em Java o código-fonte da **Classe Ponto** no Plano Cartesiano com seus respectivos atributos e operações, mantendo total compatibilidade com a modelagem. Ao final, codifique a **Classe TestePonto**, onde deve-se instanciar a Classe Ponto e testar o seu funcionamento.
6. Implemente em Java o código-fonte da **Classe NumeroInteiro** com seus respectivos atributos e operações, mantendo total compatibilidade com a modelagem. Ao final, codifique a **Classe TesteNumero**, onde deve-se instanciar a Classe NumeroInteiro e testar o seu funcionamento.
7. Implemente em Java o código-fonte da **Classe Hora** com seus respectivos atributos e operações, mantendo total compatibilidade com a modelagem. Ao final, codifique a **Classe TesteHora**, onde deve-se instanciar a Classe Hora e testar o seu funcionamento.
8. Implemente em Java o código-fonte da **Classe Data** com seus respectivos atributos e operações, mantendo total compatibilidade com a modelagem. Ao final, codifique a **Classe TesteData**, onde deve-se instanciar a Classe Data e testar o seu funcionamento.