

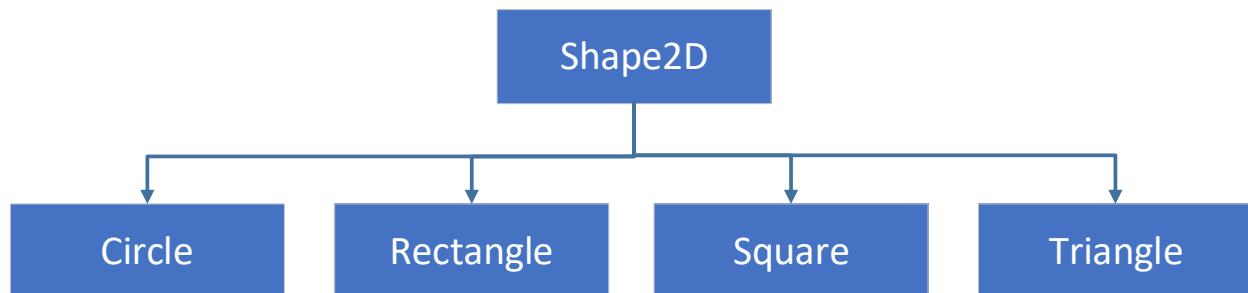
## Lista de Exercícios

### Tema 11. Herança e polimorfismo.

#### Recomendações:

- Utilizando a linguagem de programação da sua preferência (Java, Python ou ambas) implemente as classes especificadas nos exercícios a seguir.
- Utilize apropriadamente os conceitos de herança, sobrescrita de métodos, ocultação de atributos e polimorfismo discutidos na sala de aulas
- Crie em cada caso uma classe cliente, ou aplicação simples onde mostre o uso da classe implementada.

1. Considere a seguinte figura que ilustra uma hierarquia com diversas formas geométricas bidimensionais:

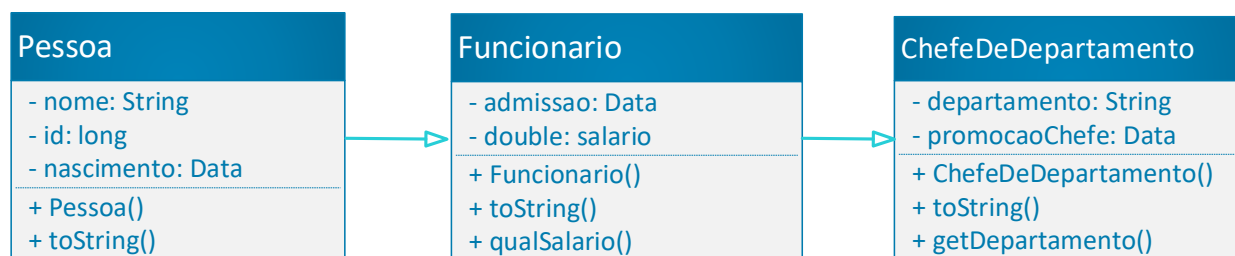


O objetivo é construir um conjunto de classes que permitam representar e manipular essas figuras considerando os seguintes requisitos.

- a. O desenvolvedor deve escolher quais os atributos e métodos para a forma bidimensional em particular segundo os requisitos exigidos
- b. Uma figura pode ser construída: (i) com o valor padrão dos atributos (definidos pelo desenvolvedor), (ii) usando os valores dos atributos como parâmetros, e (iii) a partir de uma figura criada anteriormente
- c. Ao criar uma figura deve-se verificar se os atributos fornecidos são consistentes.

- d. Suas classes deverão “desenhar” as figuras, considere a ação de desenhar como construir uma string formatada com as propriedades do objeto que pode ser impressa na tela.
- e. Deve existir um método para calcular a área da figura
- f. Deve existir um método para calcular o perímetro da figura
- g. Desenvolva uma classe adicional que utiliza o pacote de classe **Shape2D** para criar uma coleção de figuras na tela e utilize polimorfismo para executar os métodos desenvolvidos.

2. Considere a estrutura de classes implementada no tema de herança para representar **Pessoas, Funcionários e ChefeDeDepartamento**



Desejamos incluir nesse pacote de classes uma nova funcionalidade para calcular a capacidade de endividamento de um indivíduo segundo as seguintes regras:

- a. O valor mínimo de empréstimo independente da condição da pessoa é de R\$ 1000,00.
- b. Se a pessoa for um funcionário o valor do empréstimo é o maior valor entre 75% do salário, e o valor mínimo de R\$ 1000,00.
- c. Se a pessoa é chefe de departamento o valor do empréstimo corresponde a 3 vezes o valor do salário.

Modifique as classes **Pessoa**, **Funcionario** e **ChefeDeDepartamento** para conter uma funcionalidade **CalculaEmprestimo**. Utilize as classes criadas em um programa que mostre o cálculo do empréstimo para diversas instâncias de (**Pessoa**, **Funcionario** e **ChefeDeDepartamento**), considere duas alternativas (i) utilizando polimorfismo de subclasse, e utilizando polimorfismo de classe.

3. Escreva uma classe **Político** que herda da classe **Pessoa** e tem dois campos adicionais para representar as siglas e o número do partido político. Verifique se algum

método da classe *Pessoa* precisa ser sobrescrito, se necessário implemente as modificações necessárias. Escreva também as classes **Prefeito** e **Governador** que herdam da classe político e que contenham um campo para representar a cidade ou estado governado. Caso necessário sobrescreva o método **toString()** da classe **Politico**. Escreva também uma aplicação que demonstre o uso de instâncias destas classes, se criarmos uma lista de políticos e desejarmos imprimir os dados deles, qual recurso de POO devemos utilizar?

4. Faça um programa de agenda que permita guardar dois tipos de contato: pessoa física e pessoa jurídica. Para pessoa física, guarde o nome, o CPF, o endereço, a data de aniversário, o telefone e o e-mail. Para pessoa jurídica, guarde a razão social (nome), o CNPJ, o endereço, o faturamento, o telefone e o e-mail. Permita tanto listar todos os contatos da agenda quanto buscar um contato específico pelo seu CPF, se for pessoa física, ou pelo CNPJ, se for pessoa jurídica. Use herança e polimorfismo, utilize a classe *Data* criada anteriormente no curso.