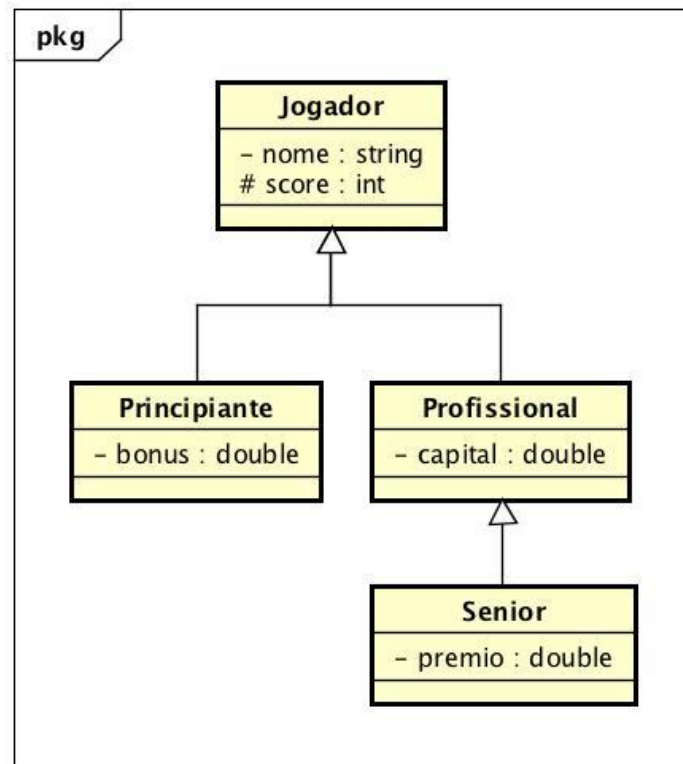


I - Hierarquia



powered by Astah

Considere a hierarquia de classes representado no diagrama acima. Os atributos são anotados com um símbolo "-" para indicar visibilidade *private* e com um símbolo "#" para indicar visibilidade *protected*. Implemente as classes com os atributos indicados no diagrama acima e com métodos da seguinte forma:

1. Classe **Jogador**
  - a. Método construtor
  - b. void imprimir() : imprime todos os atributos do objeto
2. Classe **Principiante**
  - a. Método construtor
  - b. void imprimir() : imprime todos os atributos do objeto
  - c. void ganhar(int p) :
    - i. adiciona p ao score;
    - ii. adiciona 10% de p ao bonus.
  - d. void perder(int p) :
    - i. subtrai p do score
    - ii. desconta 10% de p do bonus.

### 3. Classe **Profissional**

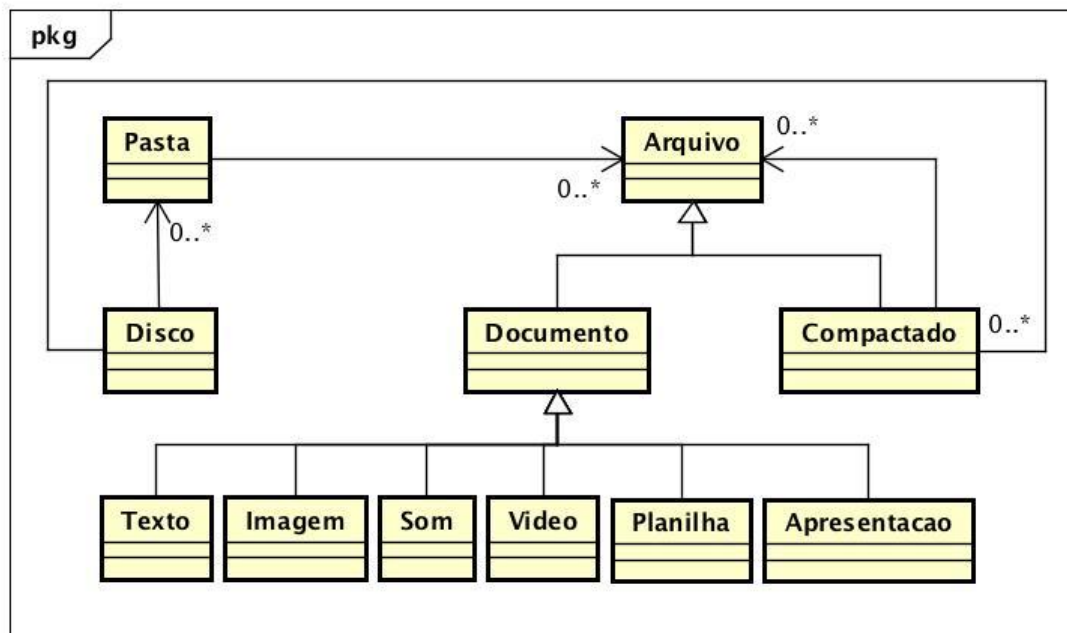
- a. Método construtor
- b. void imprimir( ) : imprime todos os atributos do objeto
- c. void ganhar(int p) :
  - i. adiciona p ao score;
  - ii. aumenta o capital de  $p * 4$
- d. void perder(int p) :
  - i. subtrai p do score
  - ii. diminui o capital de  $p * 4$

### 4. Classe **Senior**

- a. Método construtor
- b. void imprimir( ) : imprime todos os atributos do objeto
- c. void ganhar(int p) :
  - i. chama o método ganhar da superclasse
  - ii. dobra o valor do prêmio
- d. void perder(int p) :
  - i. chama o método perder da superclasse
  - ii. reduz o valor do prêmio à metade

Deve ser implementada uma classe adicional denominada **Teste** contendo o método *main* que deverá criar um objeto de cada tipo de jogador (Principiante, Profissional e Senior) e chamar todos os métodos disponíveis para esses objetos.

## II - Polimorfismo



powered by Astah

Implemente as classes representadas no diagrama acima, de acordo com a seguinte definição de atributos e métodos.

### a. Atributos

- **Disco:** um nome (apenas uma letra - char) e uma coleção de pastas
- **Pasta:** um nome (String) e uma coleção de arquivos
- **Arquivo:** um nome (String), o endereço (int) do primeiro bloco em disco, o total (int) de blocos
- **Compactado:** uma coleção de arquivos

### b. Métodos

- **Disco:** criar uma pasta, remover uma pasta, listar pastas
- **Pasta:** inserir arquivo, remover arquivo, listar arquivos, compactar, abrir todos os arquivos, duplicar um arquivo, imprimir\_nome
- **Arquivo:** abrir, criar uma réplica, imprimir\_nome
- **Documento:** não possui métodos, além do construtor
- **Compactado:** abrir, listar arquivos
- **Planilha:** abrir
- **Apresentacao:** abrir
- **Texto:** abrir
- **Imagem:** abrir
- **Som:** abrir
- **Video:** abrir

Cada implementação do método abrir na hierarquia de classes com raiz na classe Arquivo deve acionar um aplicativo apropriado. Por exemplo, na classe Planilha, o aplicativo acionado pode ser o Excel. Para fins de teste, o acionamento de um aplicativo é simbolizado simplesmente com a escrita do seu nome. Na classe Arquivo, o método abrir deve acionar um aplicativo que permita a edição de bytes; para fins de teste, a implementação deve escrever o nome do aplicativo, o endereço do primeiro byte e o total de bytes no arquivo. Na classe Compactado, o método abrir deve criar uma pasta com os respectivos arquivos.

A implementação deve ser testada com a classe Sistema abaixo.

```
public class Sistema {
    public static void main(String[] args) {
        Disco d = new Disco('C');

        Pasta escola = d.criar_pasta("Escola");
        Pasta pessoal = d.criar_pasta("Pessoal");

        Texto t = new Texto("Trabalho 10", 8, 4); // nome, primeiro bloco, número de blocos
        escola.inserir(t);
        Apresentacao a = new Apresentacao("Defesa Projeto 2", 27, 3);
        escola.inserir(a);
        Planilha p = new Planilha("Calculo 2", 40, 6);
        escola.inserir(p);
    }
}
```

```
escola.listar();

Video v = new Video("Aniversario", 33, 5);
pessoal.inserir(v);
Imagem i = new Imagem("Foto RJ", 70, 2);
pessoal.inserir(i);
Som s = new Som("Musica Chico", 58, 7);
pessoal.inserir(s);

pessoal.listar();

Compactado k = pessoal.compactar();

d.listar(); // lista as pastas Escola e Pessoal e o arquivo compactado

d.remove("Pessoal");

k.abrir(); // a pasta Pessoal deve ser recriada no disco

d.listar();

escola.abrir_todos_arquivos();
pessoal.abrir_todos_arquivos();

escola.remove("Trabalho 10");
escola.duplicar("Defesa Projeto 2");
escola.listar();
}
}
```