1. Como fazer com que tenha interações no programa:

Supondo que minha classe é o Mago e eu quero criar um personagem:

Atribuo na classe dele os parâmetros

String nome;  
 double vida;  
 int ataque;  
 double defesa;  
 int mana;  
  
 public Mago (String nome, double vida, int ataque, double defesa, int mana) {  
 this.nome = nome;  
 this.vida = vida;  
 this.ataque = ataque;  
 this.defesa = defesa;  
 this.mana = mana;  
  
 }  
  
public void imprimirEstado() {  
 System.*out*.println("Nome: " + this.nome + " | Vida: " + this.vida + " | Mana: " + this.mana);  
}

Na classe “App” utilizo a função de Scanner e mando imprimir um “título” digamos assim.

Scanner in = new Scanner(System.*in*);  
System.*out*.println("Crie seu proprio mago:");

Para os próximos itens então vou colocando as informações que meu usuário irá preencher:

System.*out*.print("Nome: ");  
String nome = in.next();  
System.*out*.print("Vida: ");  
double vida = in.nextDouble();  
System.*out*.print("Ataque: ");  
int ataque = in.nextInt();  
System.*out*.print("Defesa: ");  
double defesa = in.nextDouble();  
System.*out*.print("Fé: ");  
int fe = in.nextInt();

1. Retornar os dados para que seja possível a interação

O que é esse “char” e o in.next

char tipo = in.next().charAt(0);

O in.next() retorna o próximo texto que o usuário digitar

então poderia ser assim:String texto = in.nextE a String nada mais é que uma sequência de caracteres né? E caractere é um char. Sendo uma sequência, assim como uma lista, cada caractere tem um índice (lembra do for percorrendo a lista?). Então a String oferece um método que o charAt(index) que retorna qual é o caractere que está naquele índice. No mu exemplo estou passando zero, logo, vou pergar o primeiro caractere (e ignorar o resto do texto);

Sendo assim – é utilizado quando eu peço para meu usuário retornar com uma letra, nesse caso foi utilizado a primeira letra da palavra.

System.*out*.print("Tipo: ");  
char tipo = in.next().charAt(0);  
System.*out*.print("Nome: ");  
String nome = in.next();  
System.*out*.print("Vida: ");  
double vida = in.nextDouble();  
System.*out*.print("Ataque: ");  
double ataque = in.nextDouble();  
System.*out*.print("Defesa: ");  
double defesa = in.nextDouble();

1. Utilizando as ferramentas

Em que casos eu uso os seguintes itens:

|  |  |
| --- | --- |
| System.out.print | * Imprimir na mesma linha; |
| System.out.println | Imprimir em linhas diferentes; |
| System.in | * Acho que é para imprimir o que está nos parâmetros |
| Scanner | * Acho que ele acompanha o System.in |
| in.next() | * Deve ser para que o usuário preencha |
| in.nextInt() | * Para que o usuário coloque números inteiros |
| in.nextDouble() | * Para que o usuário retorne número com virgulo |

1. Quando utilizar o IF

Será sempre na forma booleano

if (mana > 0) {

lancarMagia();

}

boolean possuiMana = mana > 0 // guardo o resultado da comparação (true ou false) na variável "possuiMana"

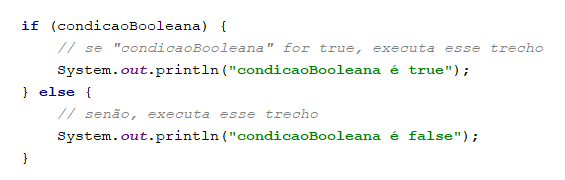
// esse if possui o mesmo comportamento do anterior, porem uma melhor legibilidade (lê-se "se possui mana, então lançar magia")

if (possuiMana) {

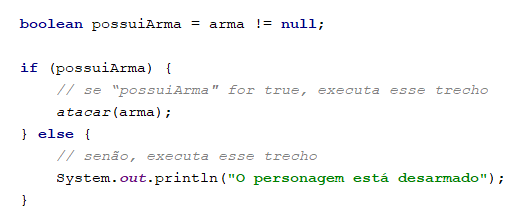
lancarMagia();

}

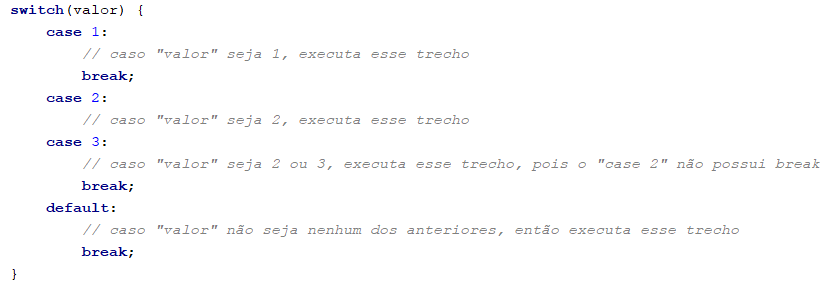
Esse é um caso em que não se usa o else , pq é implícito o uso de somente uma ação, se ela for falsa não trará nenhum “dano” para a sequencia.

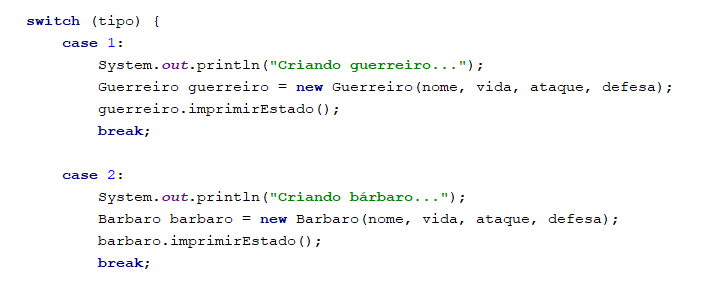
Ultilizar o else quando eu tenho uma “tarefa” que será executada se a resposta for não

Como no caso a seguir:



1. Quando utilizar o SWITCH

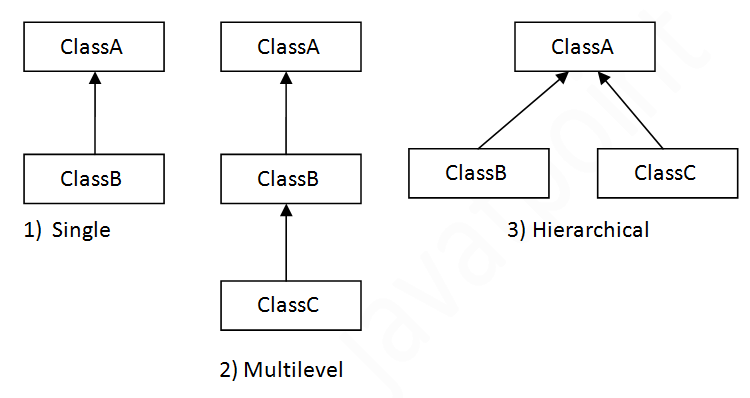




1. Quando utilizar HERANÇA

Quando a classe tem similaridade com outra classe, no exemplo visto em aula falamos sobre a classe pai “Veiculo”, a classe filha “Carro” tem atributos que na classe pai tem, porém com as suas particularidades.

Podemos seguir outras ordens como na imagem abaixo.



1. Quando usar o poliformismo