

Tarefa em Grupo

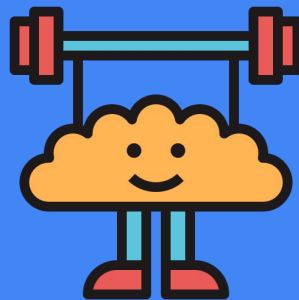
NAP II

Orientações Iniciais

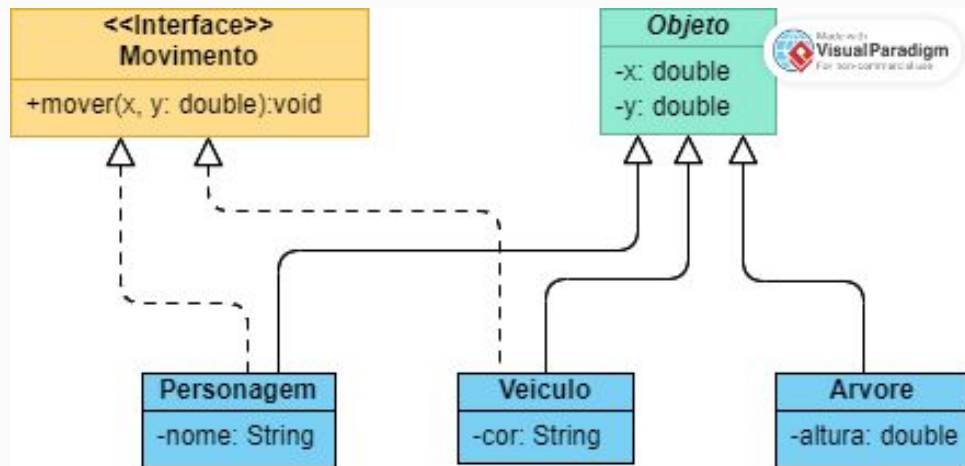
- Atividade em **grupo** cadastrada como **tarefa** no SIGAA
- Um conjunto de classes e interface, programadas em **Java**, foi fornecido nesta **atividade prática**
 - Cada classe possui **partes que precisam ser completadas**
 - As partes que precisam ser completadas estão sinalizadas com **//TODO**
 - Conclua a implementação das classes, representadas no diagrama a seguir
- Qualquer IDE Java pode ser utilizada
 - Crie um projeto vazio nessa IDE
 - Copie as classes e interface fornecidas na tarefa para o projeto criado
 - Complete os trechos de código incompletos
- Quando concluir a atividade
 - Envie as mesmas classes e interfaces em um único arquivo comprimido (**.zip**) pelo SIGAA
 - Acrescente nos comentários do formulário de submissão os nomes e matrículas dos membros do grupo
 - Apenas um membro do grupo deve submeter a resposta e novas submissões podem ser realizadas

Exercício - Parte I

Como seria implementar os objetos que se movimentam ou não em um jogo?

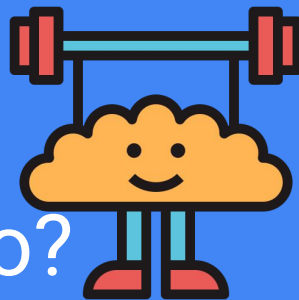


- Imagine um **Game** com
 - Personagens, veículos e árvores
- Objeto** é uma classe abstrata que representa os objetos do jogo e identifica a localização dos mesmos (coordenadas x e y)
 - Personagem**, **Veículo** e **Árvore** são classes concretas de Objeto
- Movimento** é uma interface que define o método `mover(x, y)`, indicando os objetos que podem se movimentar no jogo
 - Atribui os argumentos x e y no objeto
 - Personagem** e **Veículo** podem se mover
 - Árvore** não pode se mover no jogo pois não implementa Movimento



Exercício - Parte II

Como seria implementar os objetos que se movimentam ou não em um jogo?



- Sobrescreva o método de representação do objeto como uma String em **todas** as classes do diagrama (*Objeto*, *Personagem*, *Veiculo* e *Arvore*)
 - Usar **todos** os atributos do objeto na representação como String
 - Em subclasses considerar os atributos da superclasse Objeto
 - Sobrescrever
 - `public String toString()`
- Criar uma **classe/programa principal** que
 - Crie uma lista de Objeto
- Ainda na **classe/programa principal** que
 - Popule a lista com pelo menos cinco instâncias de Objeto
 - Considere instanciar Personagem, Veiculo e Arvore
 - Imprima cada objeto da lista
 - Crie uma lista de Movimento
 - Popule a lista com pelo menos duas instâncias de Movimento
 - Considere instanciar Personagem e Veiculo
 - Para cada objeto da lista chamar o método `mover(x, y)`
 - Considere usar valores de x e y aleatoriamente gerados
 - Imprima o objeto movido

**O TRABALHO ÁRDUO
VENCE O TALENTO,**
QUANDO O TALENTO NÃO
TRABALHOU ARDUAMENTE.

Kevin Durant

