



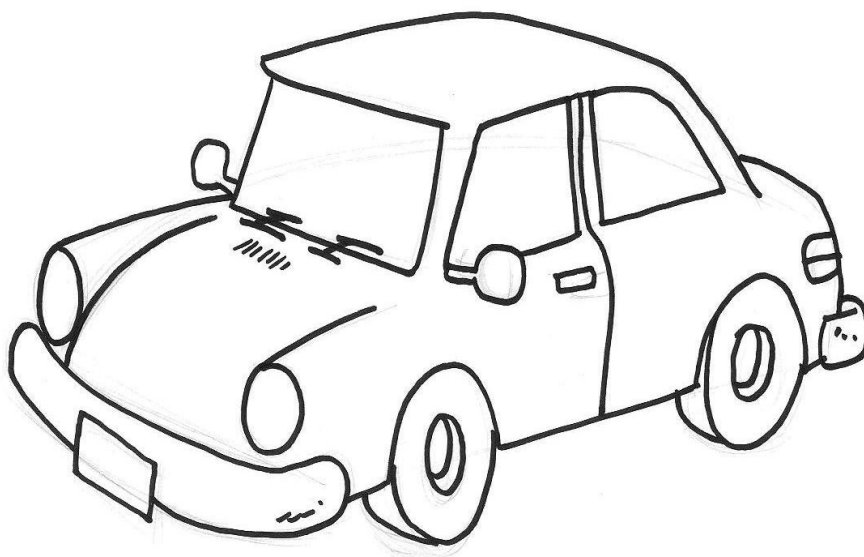
Instituto Politécnico de Castelo Branco
Escola Superior de Tecnologia



Programação II

2013/14

AlbiRent



Autores

20120961	Waldir Vigário	Engenharia Informática	C
20120896	Paulo Nicolau	Engenharia Informática	C

Classes Desenvolvidas

Classe Estação

Descrição

Para esta classe foram introduzidos dois atributos, id e nome, em id é armazenado o numero de identificação da estação e em nome o seu respetivo nome.

A classe tem um construtor que recebe o nome e o id como parâmetros de entrada.

Métodos:

- **setID**: adiciona um id a estação.
- **setNome** : executa o método de validação para posteriormente adicionar um nome à estação.
- **getID** : devolve o id da estação;
- **getNome**: devolve o nome da estação.
- **validaID**: verifica se o id recebido como parâmetro é valido e adiciona-o ao atributo “id”.
- **validaNome**: verifica se o nome recebido como parâmetro é valido e em caso afirmativo atribui o valor ao atributo “nome”.
- **equals**: verifica se o id inserido como parâmetro é igual ao id da estação.

Classe PeriodoTempo

Descrição

Cria um período indicando a hora inicial e final garantindo que a inicial é menor que a final. Tem 2 variáveis: *ini* e *fim*.

A classe tem 2 construtores:

- Um para criar um objeto com base no tempo inicial e final exatos;
- Um para criar um objeto com base no tempo inicial e na duração do período

Métodos:

- **getFim**: retorna o tempo final.
- **setFim**: altera o tempo final.
- **getIni**: retorna o tempo inicial.
- **setIni**: altera o tempo final.
- **toString**: retorna o período em forma de string.
- **estaDentro**: indica se um Tempo está dentro deste período de tempo.
- **intersecta**: Verifica se um, outro, período de tempo intersecta ou não este período.
- **estaContido**: indica se o nosso período está englobado dentro do período “x”.
- **contem**: indica se o período “x” está englobado no período.
- **junta**: Junta o período p ao nosso. A junção só é válida se os períodos se intersectarem, caso contrário, não há junção.
- **getUniao**: devolve a união do nosso período com o período “x” se os períodos não se intersectarem não há união. Se se intersectarem a união indica o maior período de tempo possível juntando os dois.
- **getInterseccao**: indica a interseção do nosso período com o período “x”, ou seja, qual o período de tempo que possuem em comum.

- **getDuracaoSegs**: devolve um período de tempo em segundos.

Classe Carro

Descrição

Cria um carro, com os atributos: id, modelo, classe, estacao e alugueres.

A classe tem um construtor que recebe como parâmetro o id, o modelo, a classe e a estacao.

Métodos:

- **setID**: altera o ID.
- **setModelo**: altera o modelo.
- **setClasse**: altera a classe.
- **setEstacao**: altera a estação.
- **setAluguer**: adiciona um novo aluguer.
- **getID**: devolve o id do carro.
- **getModelo**: devolve o modelo do carro.
- **getClasse**: devolve a classe do carro.
- **getEstacao**: devolve a estação inicial do carro.
- **getAluguer**: devolve os alugueres.
- **validaID**: valida o valor do id, e retorna o valor validado
- **equals**: verifica se os carros são iguais.
- **estaDisponivel**: diz se o carro esta disponível no tempo indicado.
- **estaDisponivel**: recebe um período de tempo e verifica se o carro esta disponível nesse periodo.
- **ondeEsta**: diz em que estação se encontra o carro.

Classe Classe

Descrição

Representa uma classe de carro, e tem como atributos id, descricao, custoDia, custoHora, taxaViagem, carros.

A classe tem um construtor que recebe como parâmetros de entrada id, descricao, custoDia, custoHora, taxa.

Métodos:

- **setCarros**: acrescenta carros à classe.
- **removeCarro**: retira o carro da classe.
- **getCarros**: devolve os carros da classe.
- **getNumCarros**: retorna o numero de carros na classe.
- **validaID**: valida o valor recebido e atribui o valor ao atributo "id".
- **validaDescricao**: valida o valor recebido e atribui o valor ao atributo 'descricao'.
- **validaCusto**: valida o valor recebido e retorna o valor validado.
- **validaTaxa**: valida o valor recebido e atribui o valor ao atributo 'taxa'.
- **setID**: define o id da classe.
- **setDescricao**: define a descrição.

- **setCustoDia:** modifica o custo diário.
- **setCustoHora:** modifica o custo por hora.
- **setTaxa:** define a taxa.
- **getID:** retorna o id da classe
- **getDescricao:** retorna a descrição
- **getCustoDia:** retorna o custo diário
- **getCustoHora:** retorna o custo por hora
- **equals:** verifica se as classes são iguais.
- **getNumCarrosDisponiveis:** retorna o numero de carros disponíveis, no período indicado.
- **getCarrosDisponiveis:** retorna os carros disponíveis.
- **calcCustoMesmaEst:** retorna o custo do aluguer do carro se for entregue na mesma estação.
- **calcCustoOutraEst:** retorna o custo do aluguer do carro se for entregue em outra estação

Classe Aluguer

Descrição

Representa os alugueres, e tem como atributos: periodo, estLevant, estEntrega, custo, carro.

A classe tem um construtor que recebe os seguintes parâmetros de entrada: periodo, estLevant, estEntrega, custo e carro.

Métodos:

- **getEstLevant:** retorna a estação onde se levanta o carro
- **getPeriodo:** retorna o período do aluguer.
- **getEstEntrega:** retorna a estação onde se entrega o carro.
- **getCusto:** retorna o custo do aluguer.
- **getCarro:** retorna o carro.
- **setPeriodo:** define o período de tempo do aluguer.
- **setEstLevant:** define a estação onde se levanta o carro.
- **setEstEntrega:** define a estação onde se entrega o carro
- **setCusto:** define o custo do aluguer.
- **setCarro:** define o carro a ser alugado.

Objetivos Atingidos e Não Atingidos

Objectivo	Grau de implementação			
	Sem bugs	Quase tudo	Quase nada	Não se fez
PeriodoTempo implementada corretamente	X			
Estacao implementada corretamente	X			
Carro implementada corretamente	X			
Classe implementada corretamente	X			
Aluguer implementada corretamente	X			
Fazer aluguer	X			
listar preços por classe	X			
listar carros disponíveis	X			
Ver alugueres	X			
Ver Carros	X			
Ver classes	X			
Ver carros numa estação	X			

Bugs verificados

No momento de entrega não foi verificado nenhum bug.