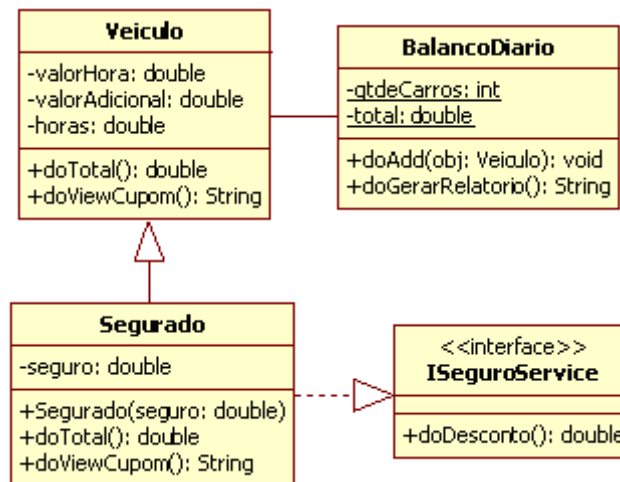


## 1. Classe de modelagem

- 1.1. No eclipse, crie um projeto chamado ddd-veiculos.
- 1.2. Crie uma classe Veiculo e Segurado com os seguintes métodos e atributos:
- 1.3. Crie os métodos de acesso *getters* e *setters*.



### 1.4. Regras:

- 1.4.1. Regra para o método **doTotal()**: Retorna o resultado do valor a pagar para o estacionamento através do cálculo **valorHora + valorAdicional \* (horas - 1)**.
- 1.4.2. Regra para o método **doViewCupom()**: Retorna a String formatada.
- 1.5. Crie a classe **segurado** com seus atributos e métodos
- 1.6. Regras para a classe **Segurado**
  - 1.6.1. Codifique a classe conforme diagrama. Não esqueça que a classe **Segurado** herda os dados de **Veiculo** e implementa a interface **ISeguroService**.
  - 1.6.2. Regra para o construtor **Segurado(double seguro)**: Altera o atributo **seguro** com o valor passado como parâmetro. Obs.: use o *this*.
  - 1.6.3. Regra para o método **doDesconto()**: Retorna o valor do desconto para veículos segurados através do cálculo **super.doTotal() \* seguro / 100**
  - 1.6.4. Regra para o método **doTotal()**: Retorna o resultado do valor a pagar para o estacionamento através do cálculo **super.doTotal() - doDesconto()**.
  - 1.6.5. Regra para o método **doViewCupom()**: Retorna a String formatada.

### Revendo conceitos:

- **Herança**: A classe herda os atributos e métodos da superclasse
- **Sobrescrita** de método: Sobrescrevo um método da superclasse, isso ocorre porque na subclasse a regra de negócio é diferente, eu tenho que aplicar uma outra lógica na subclasse.

### 1.7. Criar a classe **BalancoDiario**

- 1.7.1. Codifique a classe conforme diagrama. O sublinhado indica que o atributo é da classe e deve conter o modificador *static*.
- 1.7.2. Não é necessário criar os métodos de acesso *getters* e *setters*, pois os atributos não serão acessados diretamente e, sim, pelo método **doGerarRelatorio()**
- 1.7.3. Regra para o método **doAdd(Veiculo obj)**: Incrementa a quantidade de carros e incrementa o total com o valor do método **doTotal()** do objeto **obj**.
- 1.7.4. Regra para o método **doGerarRelatorio()**: Retorna a String formatada.

### 1.8. Criar a classe **TesteEstacionamento**.

- 1.8.1. Crie uma classe de teste chamada de **TestaEstacionamento** no pacote **test**.
- 1.8.2. Esta classe precisará conter o método **main**.
- 1.8.3. Instancie um objeto do tipo **Segurado** chamado de **veicSeg**. Passe o valor 10 como parâmetro do construtor.
- 1.8.4. Altere o valor do atributo **valorHora** para 5, **valorAdicional** para 2 e **horas** para 0.
- 1.8.5. Exiba na tela o resultado do método **doViewCupom()**.
- 1.8.6. Instancia um objeto do tipo **BalancoDiario** chamado **bd**.
- 1.8.7. Adiciona o objeto **veicSeg** no objeto **bd** através do método **doAdd()**.
- 1.8.8. Exibe o resultado do método **doGerarRelatório()**.
- 1.8.9. Execute a classe e verifique o que ocorre

Organize a indentação de TODAS as classes com **CTRL + SHIFT + F**