

## PROGRAMAÇÃO ORIENTADA À OBJETOS – PF

- 1) Crie as classes **Veículo**, **Carro** e **Moto** conforme o diagrama de classes abaixo, incluindo os métodos capturadores e os métodos modificadores dos atributos de cada classe.

<b>Veículo</b>
- ano: int - nome: String - preço: float - passageiros: int
+ getAno(): int + setAno(ano): void + getNome(): String + setNome(nome): void + getPreco(): float + setPreco(preco): void + getPassageiros(): int + setPassageiros(passageiros): void

<b>Carro</b>
- Marchas: int - categoria: String - Modelo: String
+ getMarchas(): int + setMarchas(marchas): void + getCategoria(): String + setCategoria (categoria): void + getModelo(): String + setModelo (modelo): void + aulas(): int

<b>Moto</b>
- Marca: String - Bagagem: int - Consumo: int
+ getMarca(): String + setMarca(marca): void + getBagagem(): int + setBagagem(bagagem): void + getConsumo(): int + setConsumo(consumo): void + aulas(): int

- 2) Na classe **Carro**, crie o método **Aulas** que deve calcular e retornar a quantidade de horas de aulas de direção, conforme a quantidade de marchas e a quantidade de passageiros, de acordo com a fórmula abaixo

$$\text{Horas} = (\text{marchas} * 10 + \text{passageiros} * 3)$$

- 3) Na classe **Moto**, crie o método **Aulas** que deve calcular e retornar a quantidade de horas de aula de direção, conforme o consumo e a quantidade de passageiros, de acordo com a fórmula abaixo.

$$\text{Horas} = (\text{consumo} * 12 + \text{passageiros} * 2)$$

- 4) Crie um método construtor em cada classe que preencha todos os atributos da classe.
- 5) Crie a classe **Dirigir** que crie um objeto da classe **Carro** e outro objeto da classe **Moto**, criando os métodos construtores, fazendo a entrada dos dados pelo usuário (atributos da classe mãe e da classe filha), validando todas as entradas, exibindo todas as informações desses objetos, incluindo a quantidade de horas de curso de direção. O programa deve tratar das exceções ... `NumberFormatException`.