Obs: Para todos os exercícios, criar os atributos privados e acessar os mesmos através de métodos públicos “getters and setters”.

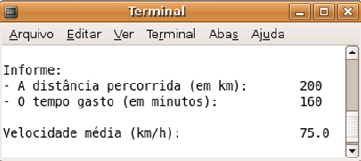
1. - Operadores

Crie um projeto chamado **CalculaVelocidadeMed**, que calcule a velocidade média de um veículo com base em uma distância por ele percorrida e no tempo gasto para isso.

O projeto deve solicitar a distância percorrida em **kilometros** e o tempo em **minutos**.

Ao final, deve apresentar a **velocidade média** do veículo em kilometros por hora (**km/h**).

A figura abaixo ilustra um exemplo das saídas que deverão ser produzidas:



Obs: Fazer a conversão de Minutos para Horas.

2. - Operadores

Crie um projeto chamado **AnalisaNumero**, que solicite um número inteiro e que indique se ele é um número par ou ímpar. A figura abaixo ilustra as saídas que deverão ser produzidas:

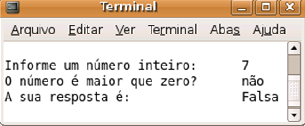


3. - Operadores

Crie um novo projeto, chamado **ValidaTamanhoNumero**, que solicite um número inteiro ao usuário e lhe pergunte, em seguida, se o número por ele informado é maior que zero.

Analisando o número informado e a resposta do usuário, o projeto deve determinar se a resposta é verdadeira ou falsa.

A figura abaixo ilustra as saídas que deverão ser produzidas:



4. - Operadores

Crie um novo Java Project, chamado **CalculaImpostoDeRenda**, que solicite o **salário** do usuário e calcule a **alíquota** e o **valor do imposto de renda** devido.

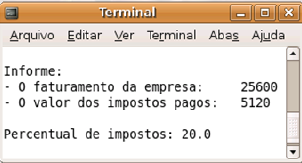
A figura abaixo ilustra as saídas que deverão ser produzidas.

A alíquota do imposto a ser aplicada sobre o salário é definida pela tabela abaixo.

|  |  |
| --- | --- |
| Aliquotas | Saida |
|  |  |

5. - Operadores

Crie um novo Java Project, chamado **CalculaPercentualmposto**, que solicite o faturamento mensal de uma empresa qualquer e o valor pago por ela a título de impostos e que calcule o percentual do faturamento que foi gasto com os impostos. A figura abaixo ilustra as saídas que deverão ser produzidas:



Obs: No setter de valor imposto, criar um IF que verifica o valor do imposto pago. Se for maior que o faturamento mensal, o valor do imposto deverá receber faturamento mensal.

6. – Operadores - InstanceOf

Crie um novo Java Project, chamado **InstanciasObjeto**, e em seguida crie os 3 pacotes em SRC:

* br.com.treinamento.hp.beans
* br.com.treinamento.hp.main
* br.com.treinamento.hp.interfaces

Em ... **beans**, crie as seguintes classes:

* Aranha, Cachorro, Cavalo, ExtraTerrestre, Fusca, Gol, Homem, Leao, Mulher, Palio

Em ... **interfaces**, crie as seguintes interfaces:

* Animal, Pessoa, Veiculo

Em ... **main**, crie uma classe chamada Principal.

* Cada classe deve **implementar** sua respectiva interface de contrato.
* As classes Aranha e ExtraTerrestre **não implementam** e **não estendem** nenhuma outra classe.
* Cria as variaveis de instância para Todas as Classes declaradas.

Ex: Animal cachorro = new Cachorro();

Utilizando “instanceOf”, crie regras para verificar se a variável de referência é uma instância do tipo declarado.

Ex Se (cavalo **instanceof** Animal) ?

Se sim, imprimir “Cachorro é uma instância de Animal”.

Fazer essa validação para todas as variáveis criadas de acordo com seus tipos.