**RESUMO OO – Orientação a Objetos**

1. Classe: é a forma que será dada ao objeto. Ela possui nome, propriedades ou atributos e métodos ou eventos. Ela não pode ser executada.
2. Objeto: nasce a partir da instanciação de uma classe. Nele se guardar dados nos atributos e executa-se os métodos (são semelhante a funções).
3. Instanciar: Criar um objeto.  
   Classe objeto = new Classe(); 🡪 Intanciando o objeto a partir da classe.
4. Classe

frmMarca

---------------------------

+ frmMarca()

+ listarMarca()

- btnInserirActionPerformed(ActionEvent)

+ main(args[]) 🡪 executa a aplicação

Marca

---------------------------

- id\_marca: int

- nome: String

---------------------------

+ getId\_marca(): int

+ setId\_marca(id\_marca)

+ getNome(): String

+ setNome(nome)

Marca

---------------------------

- id\_marca: int

- nome: String

---------------------------

+ getId\_marca(): int

+ setId\_marca(id\_marca)

+ getNome(): String

+ setNome(nome)

Marca

---------------------------

- id\_marca: int

- nome: String

---------------------------

+ getId\_marca(): int

+ setId\_marca(id\_marca)

+ getNome(): String

+ setNome(nome)

JFrame

1. Acessibilidade

Private (-): acesso privado apenas dentro da classe

Public (+): Acesso público dentro do sistema

1. **Encapsulamento**: atributos (dados) são privados e que os métodos (funções) são públicas.
2. Herança (***extends***): a classe filha herda tudo da classe mãe.
3. *Package*: é uma pasta, uma divisão física de arquivos em locais diferentes.
4. Import: faz a importação de uma classe ou biblioteca
5. MVC: Padrão de projetos que tem o objetivo de separar a tela do acesso em banco de dados por meio de uma classe de controle, semelhante a uma ponte  
   M: Model – Acesso ao banco de dados + métodos de cálculo  
   V: View – Telas do sistema  
   C: Controler – A classe que faz a interligação do Model com o View

BD

Model

Controller

View

1. Quando vai fazer a ligação com o banco de dados eu preciso de um ***Driver***.  
   Este driver dependa da linguagem (Java) e dos banco de dados (MySQL)  
   Este driver é o cano físico que liga a linguagem ao banco de dados.  
   Cada linguagem X banco de dados tem o seu próprio driver.
2. Exceção: Exception – try..catch (tenta executar...pego o erro)