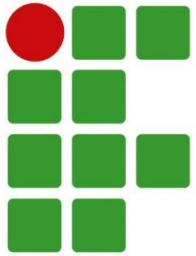


INSTITUTO FEDERAL
Goiás

Instituto Federal de Goiás
Câmpus Goiânia
Bacharelado em Sistemas de Informação
Disciplina: Programação Orientada a Objetos I

Associação de Classes: Dependência

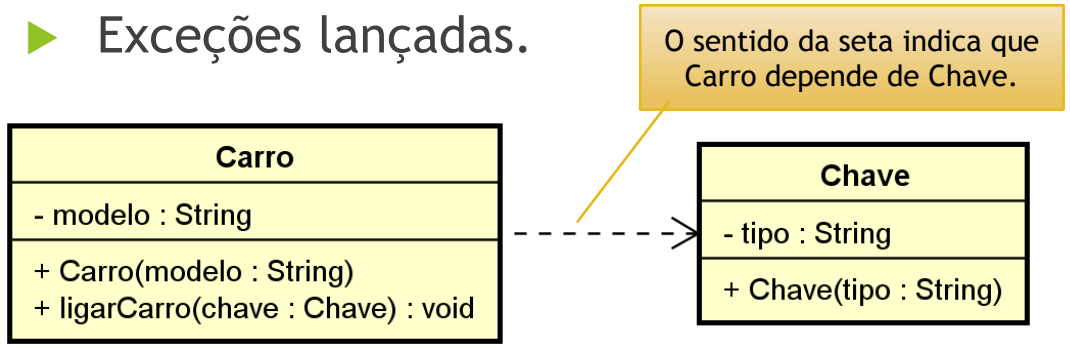
Prof. Ms. Renan Rodrigues de Oliveira
Goiânia - GO

Dependência

Uma dependência é um relacionamento não estrutural (de "uso") que tenta apontar que mudanças em uma classes pode afetar outra classe que a usa, mas não necessariamente o inverso.

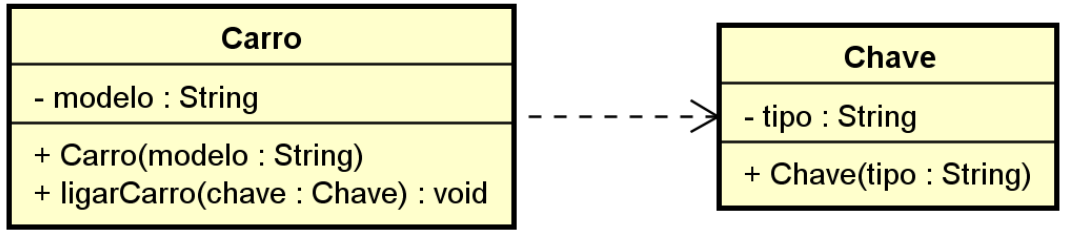
É caracterizada por um objeto depender de outro de forma temporária, onde provavelmente será :

- Parâmetros de entrada de métodos;
- Tipos de retorno de métodos;
- Utilizado dentro do código de métodos;
- Exceções lançadas.



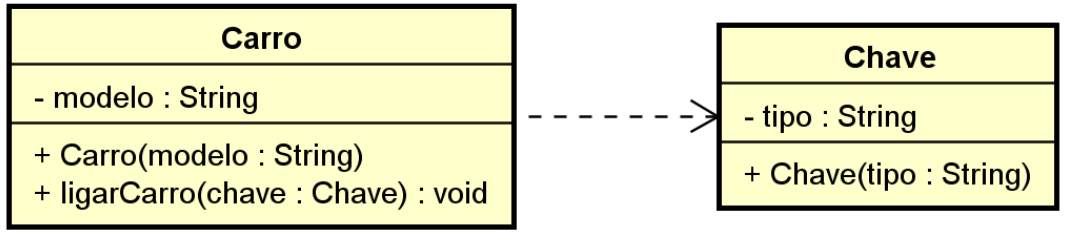
Um Carro "Depende" da Chave "Temporariamente".

Dependência
Implementando a Classe Chave




```java
package dependencia;

public class Chave {

    private String tipo;

    public Chave(String tipo) {
        this.tipo = tipo;
    }

    public String getTipo() {
        return tipo;
    }

    @Override
    public String toString() {
        return "Chave [tipo=" + tipo + "]";
    }
}
```

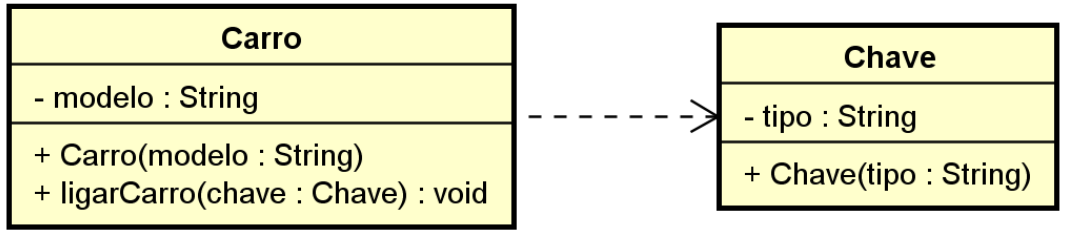
Dependência
Implementando a Classe Carro




```java
package dependencia;

public class Carro {

    private String modelo;
    private boolean ligado;
    private boolean andando;

    public Carro(String modelo) {
        this.modelo = modelo;
    }

    public void ligarCarro(Chave chave) {

        if (chave == null) {
            throw new NullPointerException("A referência da Chave não pode ser nula!");
        }

        System.out.println("Usando a " + chave + " para ligar o carro.");
        this.ligado = true;
    }
```

CONTINUA >>

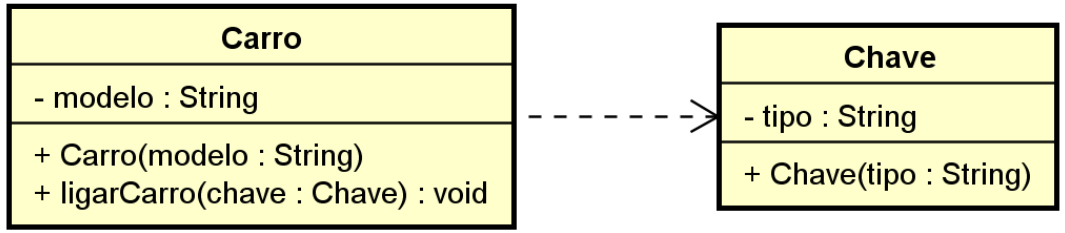
Dependência
Implementando a Classe Carro




```
23      public boolean andar() {
24
25          if (ligado) {
26              System.out.println("O Carro está andando.");
27              andando = true;
28          } else
29              System.out.println("Não é possível andar, pois o Carro está desligado.");
30
31          return andando;
32      }
```

CONTINUA >>

Dependência
Implementando a Classe Carro




```
36      public String getModelo() {
37          return modelo;
38      }
39
40      public boolean isLigado() {
41          return ligado;
42      }
43
44      public boolean isAndando() {
45          return andando;
46      }
47
48      @Override
49      public String toString() {
50          return "Carro [modelo=" + modelo + ", ligado=" + ligado + ", andando="
51                  + andando + "]";
52      }
53  }
```

Dependência
Programa Principal

```
1  package dependencia;
2
3  public class TesteCarro {
4
5      public static void main(String[] args) {
6
7          Chave chave = new Chave("ABC");
8          Carro carro = new Carro("New Fiesta");
10         System.out.println(carro);
11
12         carro.andar();
13
14         carro.ligarCarro(chave);
15         carro.andar();
16
17         System.out.println(carro);
18     }
19 }
```

Saída do Programa

```
Carro [modelo=New Fiesta, ligado=false, andando=false]
Não é possível andar, pois o Carro está desligado.
Usando a Chave [tipo=ABC] para ligar o carro.
O Carro está andando.
Carro [modelo=New Fiesta, ligado=true, andando=true]
```