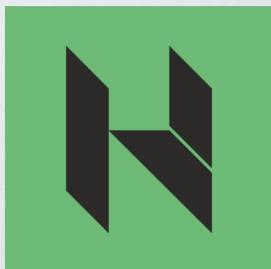


PROGRAMAÇÃO COM OBJECTOS

<https://hackerschool.tecnico.ulisboa.pt>
hackerschool@tecnico.ulisboa.pt

Simão Martins
Ricardo Marques

@HackerSchoolIST



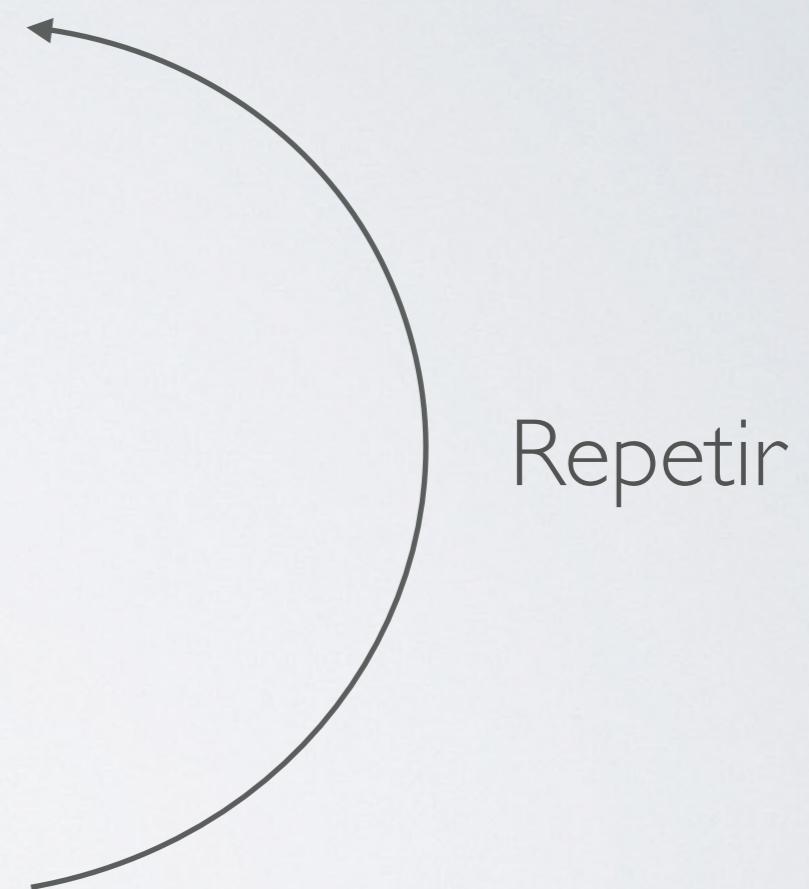
PLANEAMENTO

- 1º Workshop - Conceitos básicos de OOP
- 2º Workshop - Eclipse
- 3º Workshop - Padrões de desenho
- 4º Workshop - ?
- Mini-projecto



PLANEAMENTO

- Apresentar um a dois conceitos
- Exemplo dos conceitos
- Live demo
- Exercício
- Mini projeto



Repetir



O QUE É PROGRAMAÇÃO COM OBJECTOS?

- Maneira de pensar (Paradigma)
- Independente da linguagem

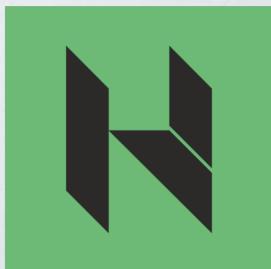




UM CARRO S.F.F.

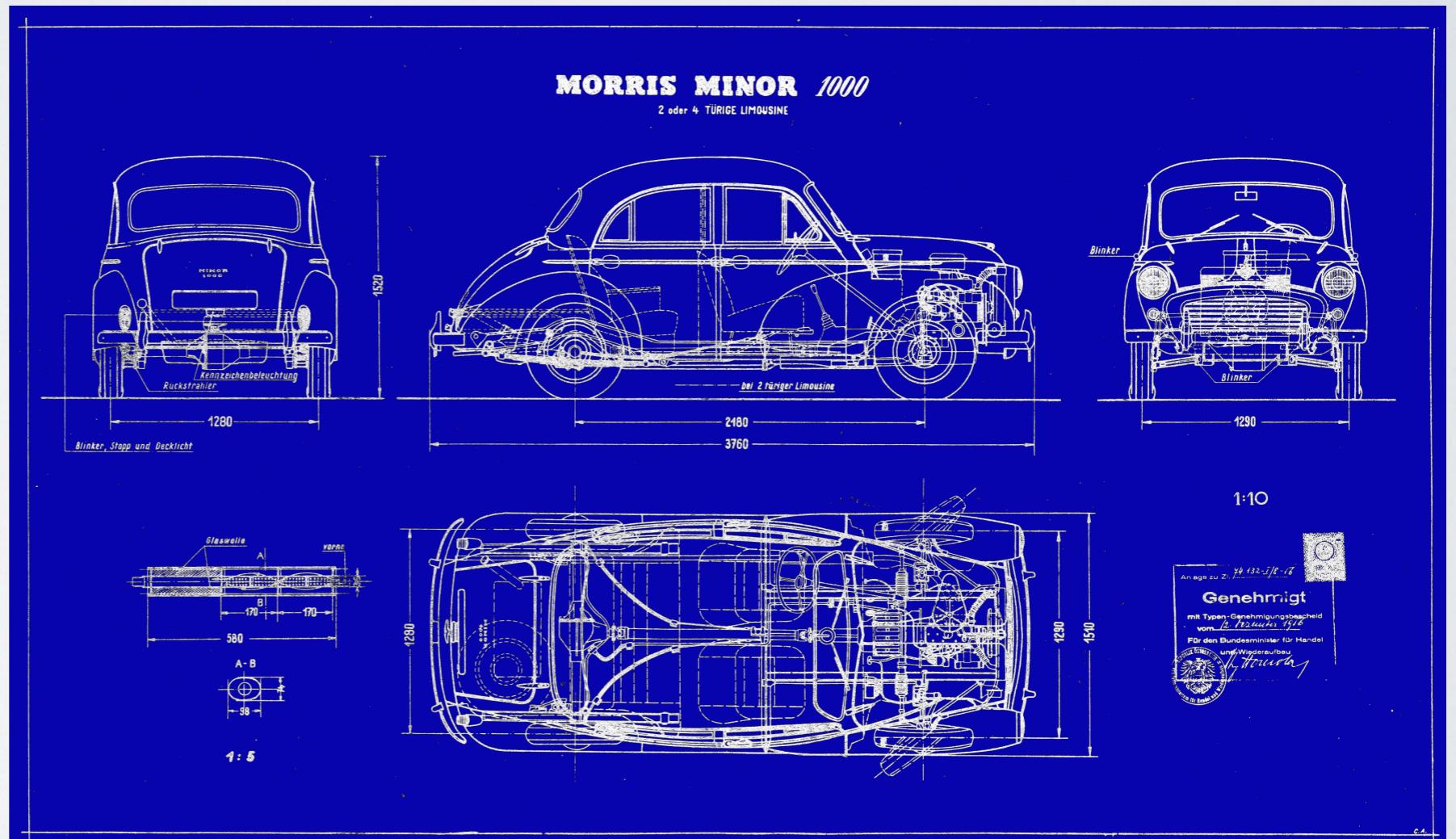
O que é preciso?

@HackerSchoolIST



O QUE É PRECISO?

- Especificações
- Material Físico

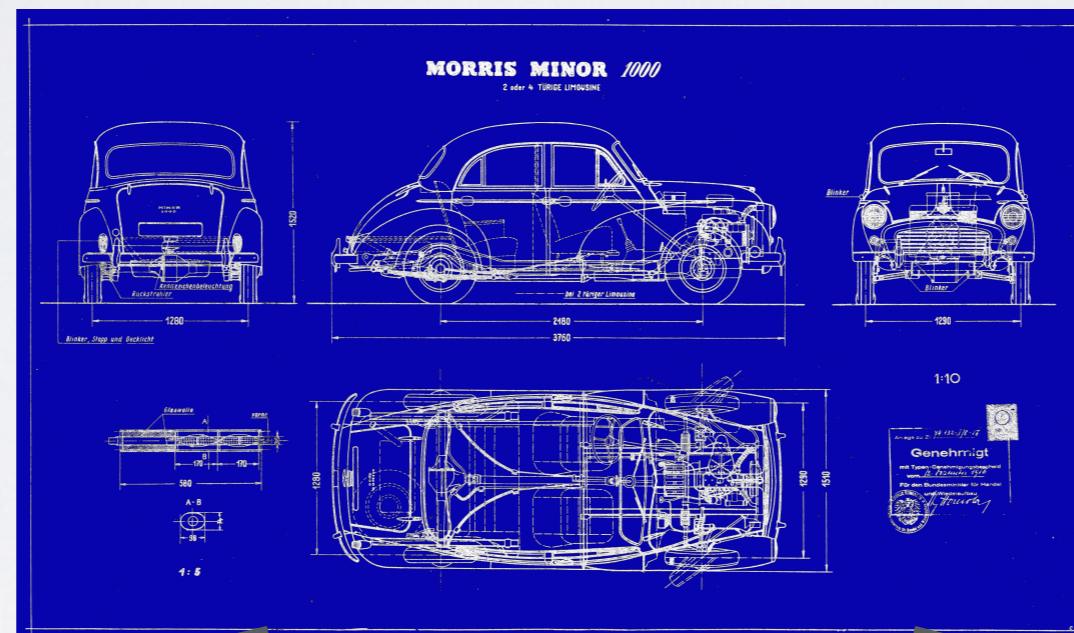


@HackerSchoolIST



CLASS, INSTANCE AND OBJECT

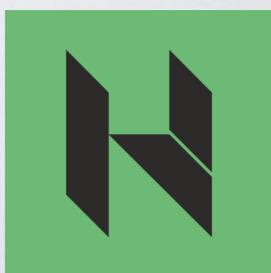
Um objecto é uma instância de uma classe



ANATOMIA D'UMA CLASSE

(em Java)

```
public class Car {  
    //Fields or Attributes  
    //Constructors  
    /*Getters and Setters  
     * Additional methods*/  
}
```



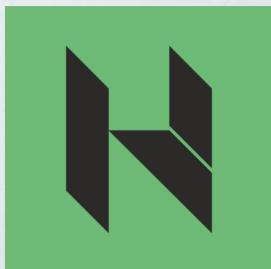
FIELDS OR ATTRIBUTES

Guardam o estado do objecto



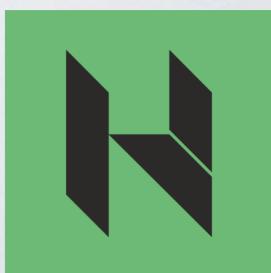
CONSTRUCTORS

Permitem criar uma nova instância duma classe



EXEMPLO

```
public class Car {  
    private String color;  
  
    public Car(String initialColor) {  
        color = initialColor;  
    }  
  
    ...  
}
```



LIVE DEMO

@HackerSchoolIST



EXERCÍCIO

Criar a classe **City** com:

- Atributos:
 - Name;
 - Latitude;
 - Longitude;
 - Neighbor cities;
- Construtores a receber:
 - Name
 - Name, Latitude e Longitude
- `toString`



GETTERS

Permitem obter o valor das propriedades do objecto



SETTERS

Permitem definir o valor das propriedades do objecto



EXEMPLO

```
public class Car {  
    String color;  
  
    String getColor() {  
        return color;  
    }  
  
    void setColor(String myColor) {  
        color = myColor  
    }  
}
```



LIVE DEMO

@HackerSchoolIST



EXERCÍCIO

Adicionar **getters** e **setters** à classe **City**



VISIBILITY MODIFIERS

Controlam o acesso aos membros de uma classe



EXEMPLO

```
public class Car {  
    private String color;  
  
    protected String getColor() {  
        return color;  
    }  
  
    public void setColor(String myColor) {  
        color = myColor  
    }  
}
```



EXEMPLO

```
public class Car {
```

```
    public static void main(String[] args) {  
        //This is our main method :)
```

```
}
```

```
}
```



LIVE DEMO

@HackerSchoolIST



EXERCÍCIO

Modificar os **getters** e **setters** da classe **City** para serem públicos e os **fields** para privado



INHERITANCE

Uma mota tem características iguais a um carro. Quais?



EXEMPLO

```
public class Vehicle {  
    // I'm a generic vehicle ^.^  
}
```

```
public class Car extends Vehicle {  
    // Brum brum...  
}
```

```
public class Motorcycle extends Vehicle {  
    // Beeem...  
}
```



LIVE DEMO

@HackerSchoolIST



EXERCÍCIO

Criar a classe **Car**, **MotorCycle** e **Vehicle**

