

# LẬP TRÌNH JAVA

NGUYỄN CHÍ HIẾU



1

NGUYỄN CHÍ HIẾU



## 1. GIỚI THIỆU JAVA

2021 2

2

## 1. GIỚI THIỆU JAVA

- Là ngôn ngữ lập trình hướng đối tượng
- Lập trình Android, Web, Desktop, ...
- Chạy trên nhiều nền tảng: Windows, Linux, MacOS, ...
- IDE: Eclipse, NetBeans, IntelliJ IDEA, ...

## 2. LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG



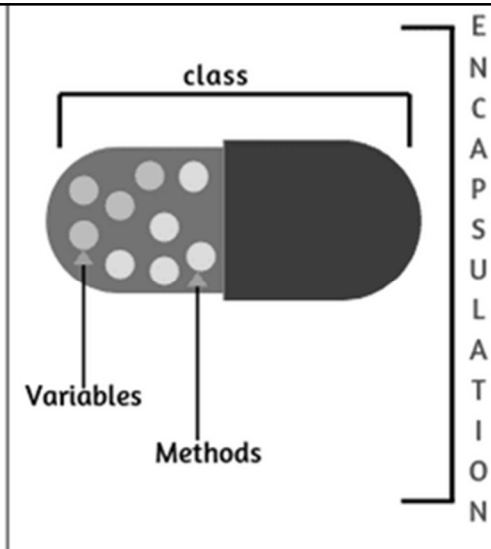
## 2. LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

### • Các khái niệm cơ bản:

- Đóng gói (*Encapsulation*)
- Kế thừa (*Inheritance*)
- Đa hình (*Polymorphism*)
- Trừu tượng (*Abstraction*)

```
class
{
    data members
    +
    methods (behavior)
}
```

ENCAPSULATION



## TÍNH ĐÓNG GÓI

```

public static class Car {
    private String manufacturer;
    private String model;
    private int maxSpeed;

    public Car(String manufacturer, String model, int maxSpeed) {
        this.manufacturer = manufacturer;
        this.model = model;
        this.maxSpeed = maxSpeed;
    }
    //...
    public int getMaxSpeed() {
        return maxSpeed;
    }
    public void setMaxSpeed(int maxSpeed) {
        this.maxSpeed = maxSpeed;
    }
}

```

2021 7

7

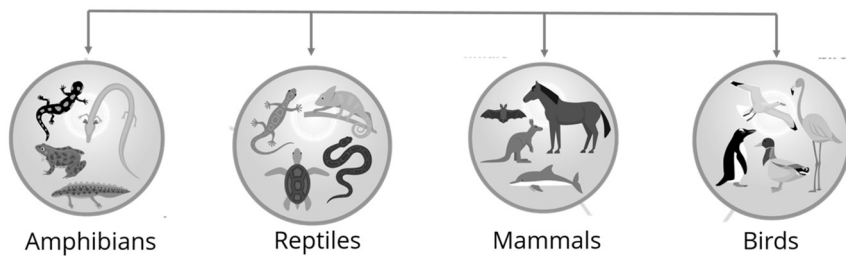
# TÍNH KẾ THỪA

Super Class



Animals

Child Class



Amphibians

Reptiles

Mammals

Birds

8

```

public static class Sedan extends Car {
    public Sedan(String manufacturer, String model, int maxSpeed) {
        super(manufacturer, model, maxSpeed);
    }
}

public static class F1Car extends Car {
    private String team;

    public F1Car(String manufacturer, String model, int maxSpeed, String team) {
        super(manufacturer, model, maxSpeed);
        this.team = team;
    }
}

```

## TÍNH ĐA HÌNH



```
public static class Car {  
    ///...  
    public void printSeat() { System.out.print("no seat"); }  
}  
  
public static class Sedan extends Car {  
    public void printSeat() { System.out.print(4); }  
}  
  
public static class F1Car extends Car {  
    public void printSeat() { System.out.print(1); }  
}
```

11

```
public static void main(String[] args) {  
    Car car = new Car("Audi", "A4", 210);  
    car.printSeat();  
  
    Car bmw = new Sedan("BMW", "X7", 245);  
    bmw.printSeat();  
  
    Car rb = new F1Car("Red Bull Racing", "RB16B", 362, "Red Bull Racing");  
    rb.printSeat();  
}
```

12

```

public abstract static class Car {
    //...
    public abstract void printSeat();
}

public static class Sedan extends Car {
    @Override
    public void printSeat() { System.out.print(4); }
}

public static class F1Car extends Car {
    @Override
    public void printSeat() { System.out.print(1); }
}

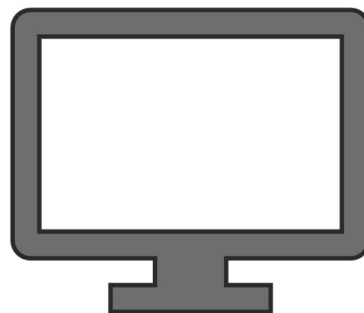
```

## TÍNH TRỪU TƯỢNG

2021 13

13

## KHÁI NIỆM INTERFACE



2021 14

14

```
public interface IController {
    public void accelerate();
    public void brake();
}

public static class Sedan extends Car implements IController {
    public Sedan(String manufacturer, String model, int maxSpeed) {
        super(manufacturer, model, maxSpeed);
    }

    @Override
    public void accelerate() { System.out.println("Sedan is accelerating ..."); }
    @Override
    public void brake() { System.out.println("Sedan is braking ..."); }
}
```

15

```
public static class Bicycle implements IController {
    private String manufacturer;
    private String model;

    public Bicycle(String manufacturer, String model) {
        this.manufacturer = manufacturer;
        this.model = model;
    }

    @Override
    public void accelerate() { System.out.println("Bicycle is accelerating ..."); }
    @Override
    public void brake() { System.out.println("Bicycle is braking ..."); }
}
```

16



### 3. JAVA COLLECTION

