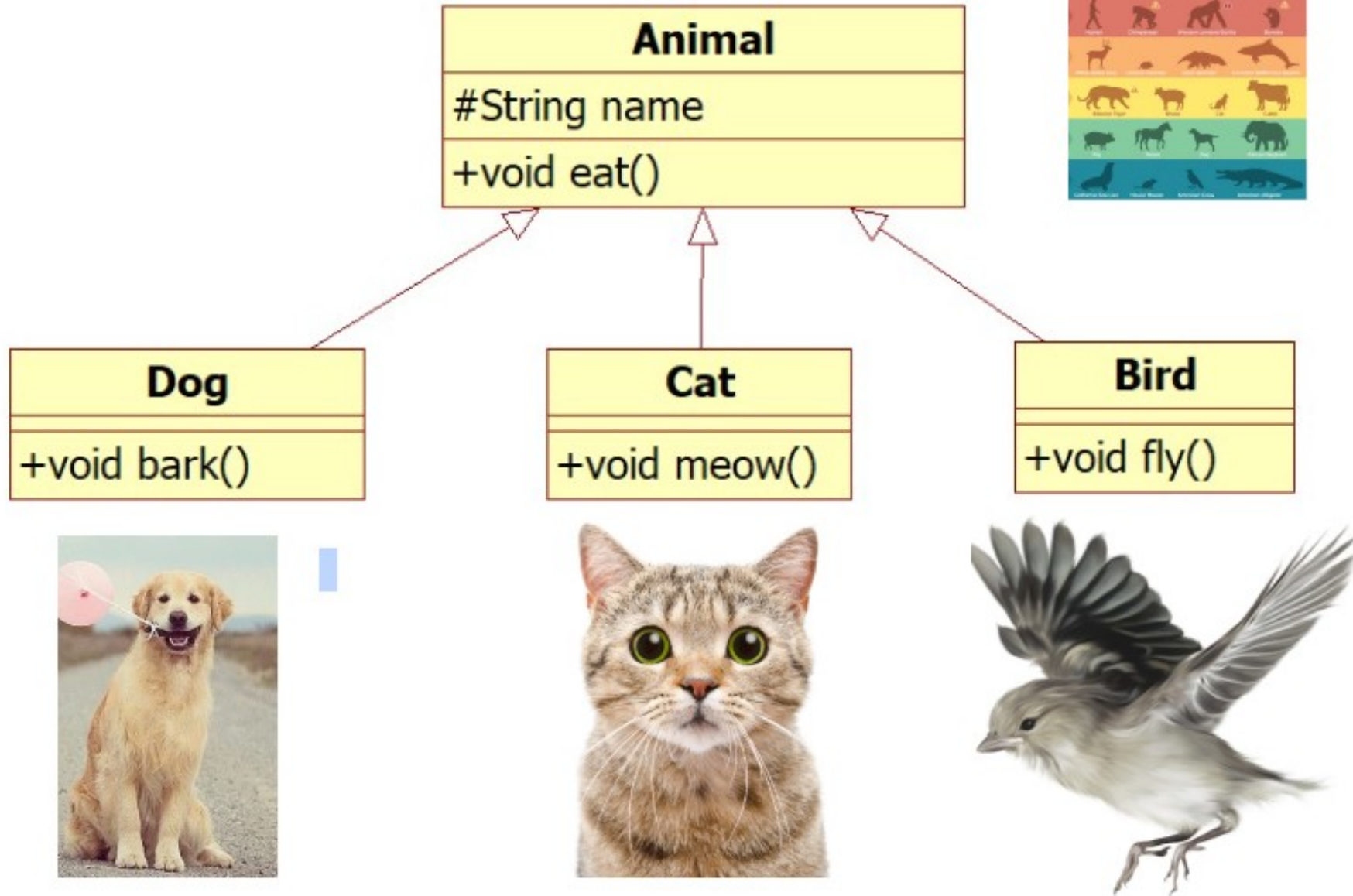


Kế thừa – Inheritance TypeScript




Kế thừa

- Kế thừa là một trong những nguyên tắc cơ bản của lập trình hướng đối tượng.
- Trong TypeScript, chúng ta có thể sử dụng kế thừa để tạo ra các lớp con (subclasses) dựa trên các lớp cha (superclasses) đã tồn tại.





Ví dụ về kế thừa



```
// shape.ts
✓ class Shape {
  |   constructor(public x: number, public y: number) {}
  |
  |   calculateArea(): number {
  |     | return 0; // Diện tích mặc định của hình là 0
  |     | }
  |
  |   getInfo(): string {
  |     | return `Hình ở (${this.x}, ${this.y})`;
  |     | }
  |   }
  | }

export default Shape;
```

```
// circle.ts
import Shape from './shape';

class Circle extends Shape {
  constructor(public x: number, public y: number, public radius: number) {
    super(x, y);
  }

  calculateArea(): number {
    return Math.PI * this.radius * this.radius;
  }

  getInfo(): string {
    return `Hình tròn ở (${this.x}, ${this.y}) với bán kính ${this.radius}`;
  }
}

export default Circle;
```

```
// rectangle.ts
import Shape from './shape';

class Rectangle extends Shape {
  constructor(public x: number, public y: number, public width: number, public length: number) {
    super(x, y);
  }

  calculateArea(): number {
    return this.width * this.length;
  }

  getInfo(): string {
    return `Hình chữ nhật ở (${this.x}, ${this.y}) có chiều rộng ${this.width} và chiều dài ${this.length}`;
  }
}

export default Rectangle;
```