

```

class Pony {
  get color() {
    console.log('get color');
    return this._color;
  }

  set color(newColor) {
    console.log(`set color ${newColor}`);
    this._color = newColor;
  }
}
const pony = new Pony();
pony.color = 'red';
// 'set color red'
console.log(pony.color);
// 'get color'
// 'red'

```

And, of course, if you have classes, you also have inheritance out of the box in ES6.

```

class Animal {
  speed() {
    return 10;
  }
}
class Pony extends Animal {
}
const pony = new Pony();
console.log(pony.speed()); // 10, as Pony inherits the parent method

```

Animal is called the base class, and Pony the derived class. As you can see, the derived class has the methods of the base class. It can also override them:

```

class Animal {
  speed() {
    return 10;
  }
}
class Pony extends Animal {
  speed() {
    return super.speed() + 10;
  }
}
const pony = new Pony();
console.log(pony.speed()); // 20, as Pony overrides the parent method

```