Revision
Node,js

JavaScript Classes

MODULES WITH DEPENDENCIES
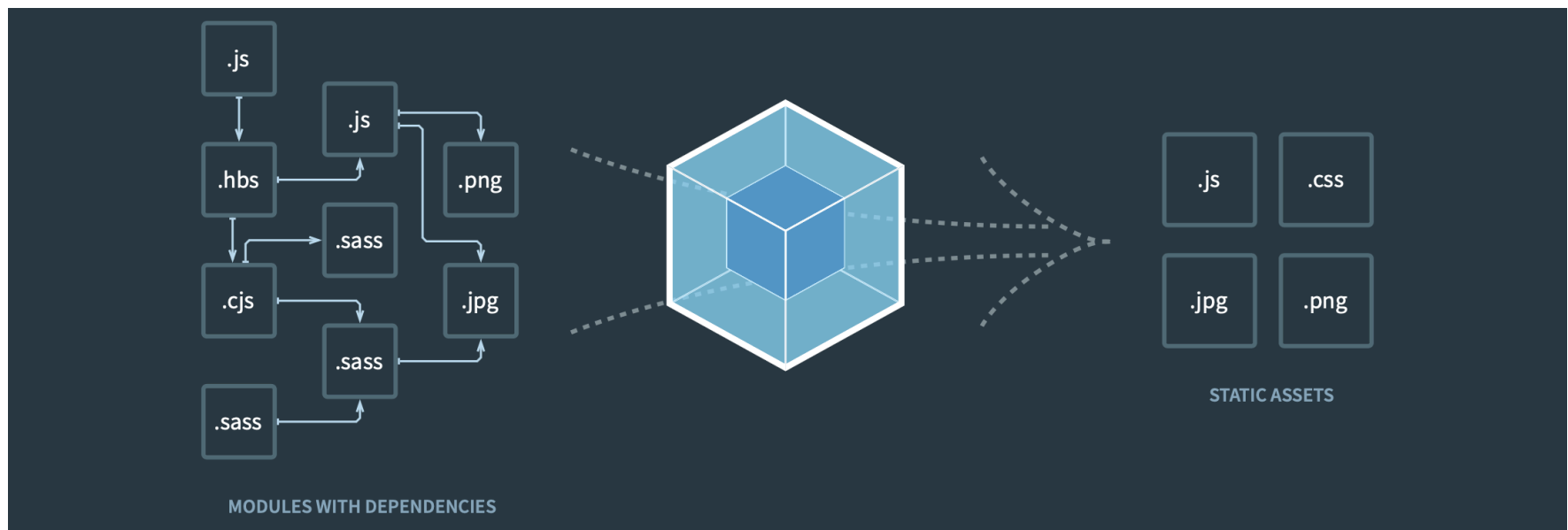
STATIC ASSETS

Basic Class Structure

```
class Ball {
     // The constructor will run when a object is created.
    constructor(diameter) {
        this.diameter = diameter;
    }

    // These methods will be added to the prototype.
    Roll() {
    }

    // This will be available on the Ball class Ball.setColour();
    static setColour(colour) {
    }
}
```

Creating Private Properties Using WeakMap

```
// WeakMaps can be used to implement private properties and methods
// There is no way to access private members implemented using WeakMaps from the
outside of an object.

//A _ is used to indicate that this is a private property
const _width = new WeakMap();
const _height = new WeakMap();
const _depth = new WeakMap();
```

```
class Box {
    constructor(width, height, depth) {
        _width.set(this, width); //Set() is used to assign a value to the private property
        _height.set(this, height);
        _depth.set(this, depth);
    }

    createBox() {
        //Get() is used to access a value in the private property
        console.log('Box width = ' + _width.get(this));
    }
}
```

Inheritance

```
// Extend allows us to Inherit from anther class
class box extends Shape {
    constructor(color) {
        // Super is used to call the parent constructor
        super(color);
    }

    draw() {
        // super is used to call the base method
        super.draw();
    }
}
```