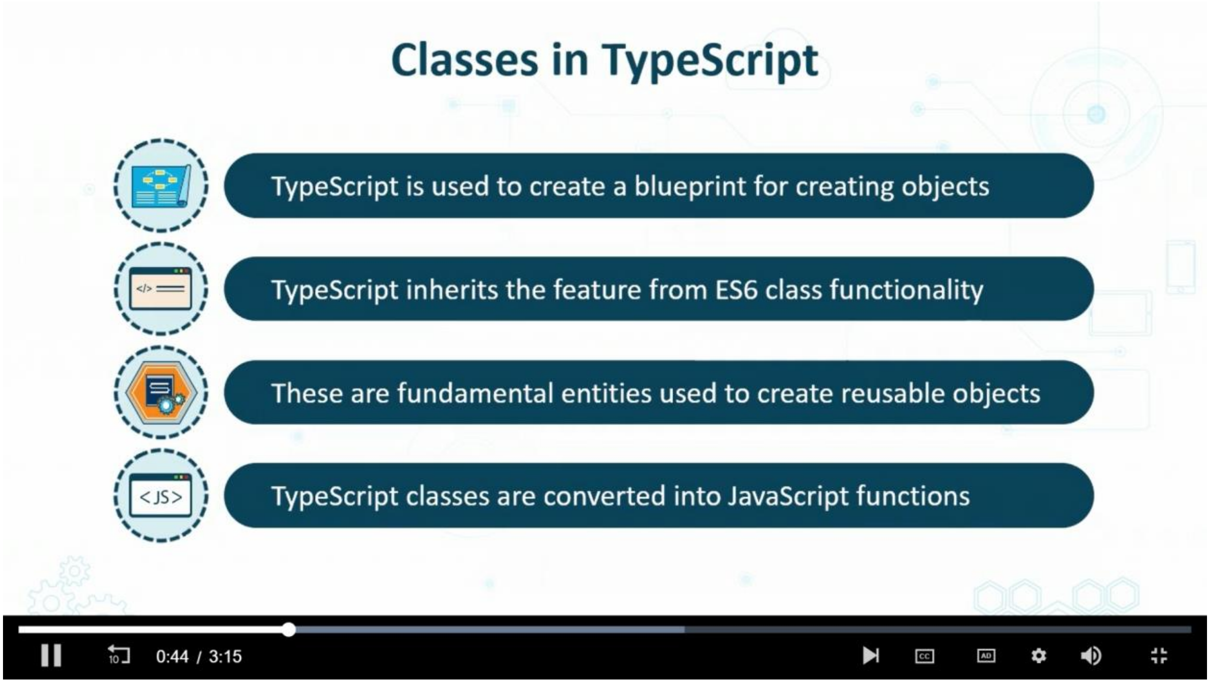
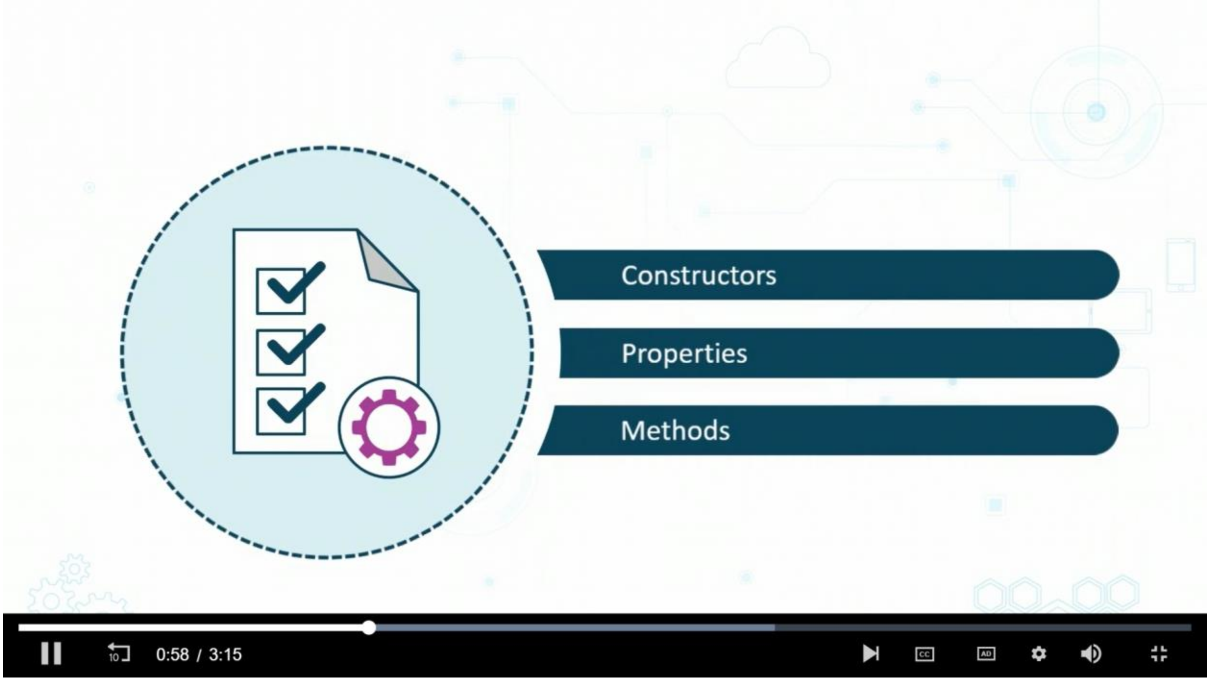
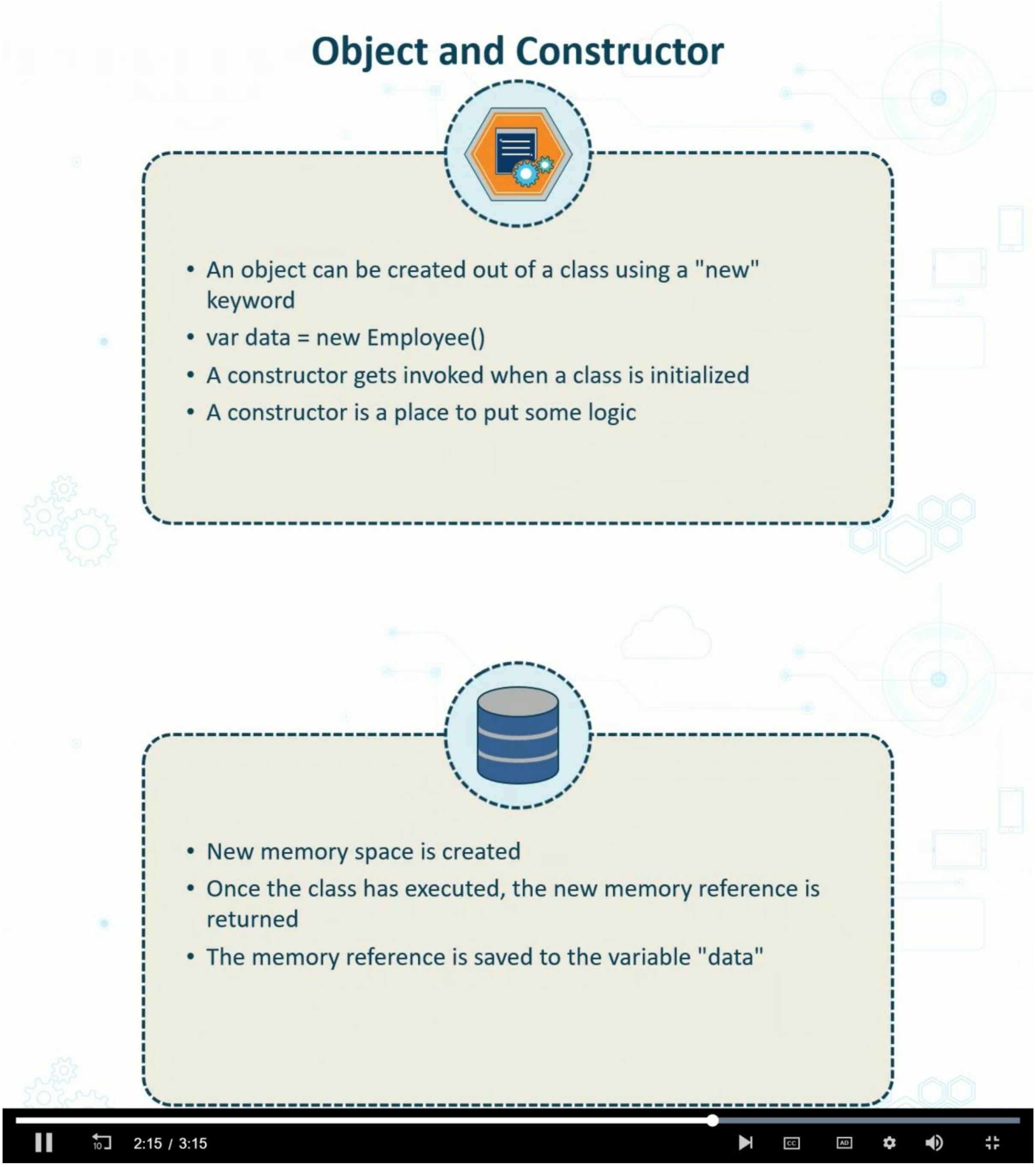


Classes in TypeScript

- TypeScript is used to create a blueprint for creating objects
- TypeScript inherits the feature from ES6 class functionality
- These are fundamental entities used to create reusable objects
- TypeScript classes are converted into JavaScript functions



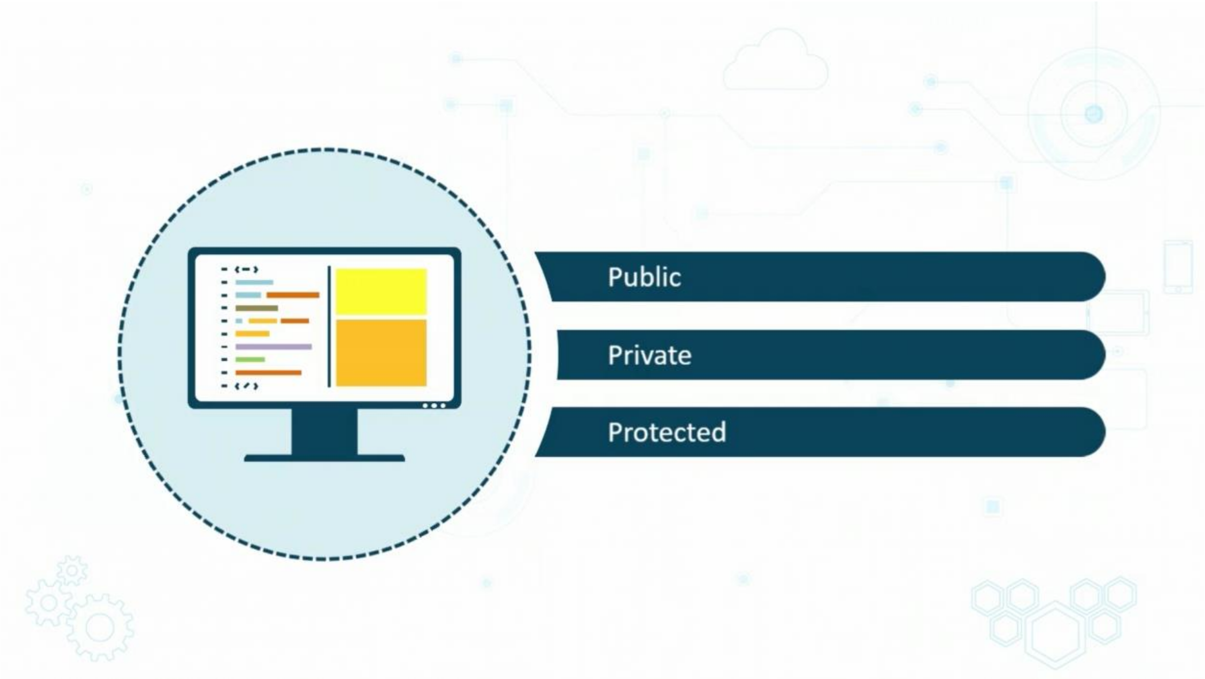
- Constructors
- Properties
- Methods



Object and Constructor

- An object can be created out of a class using a "new" keyword
- var data = new Employee()
- A constructor gets invoked when a class is initialized
- A constructor is a place to put some logic

- New memory space is created
- Once the class has executed, the new memory reference is returned
- The memory reference is saved to the variable "data"

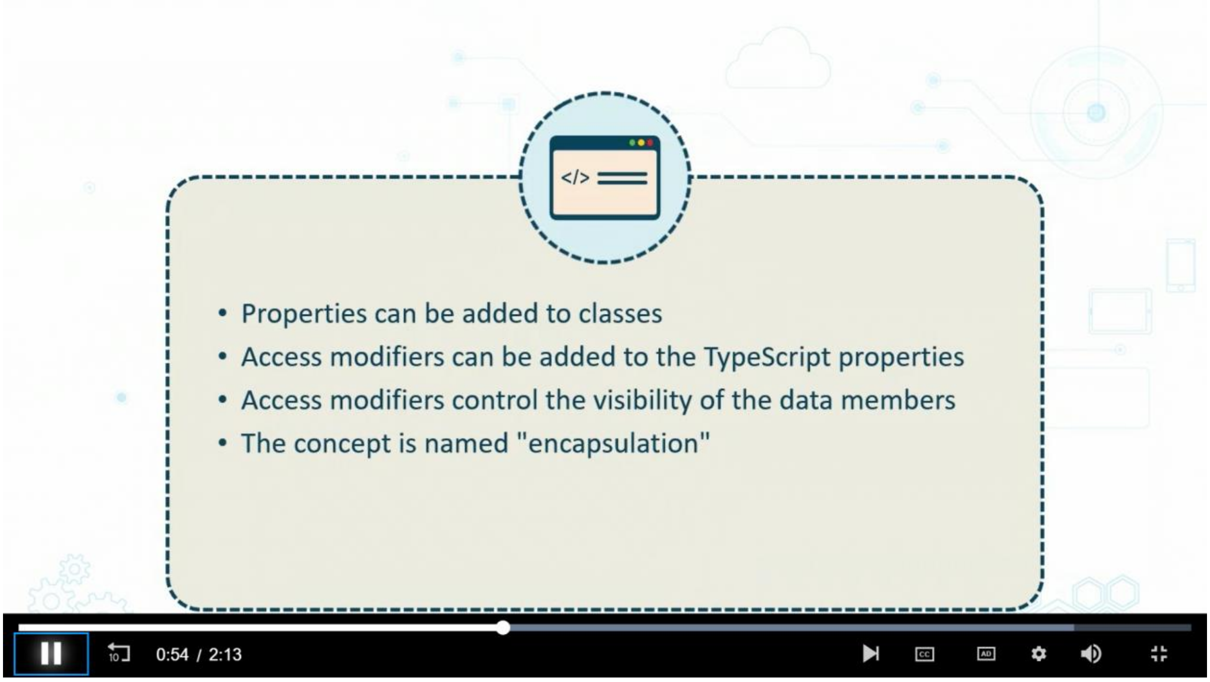


Public

Private

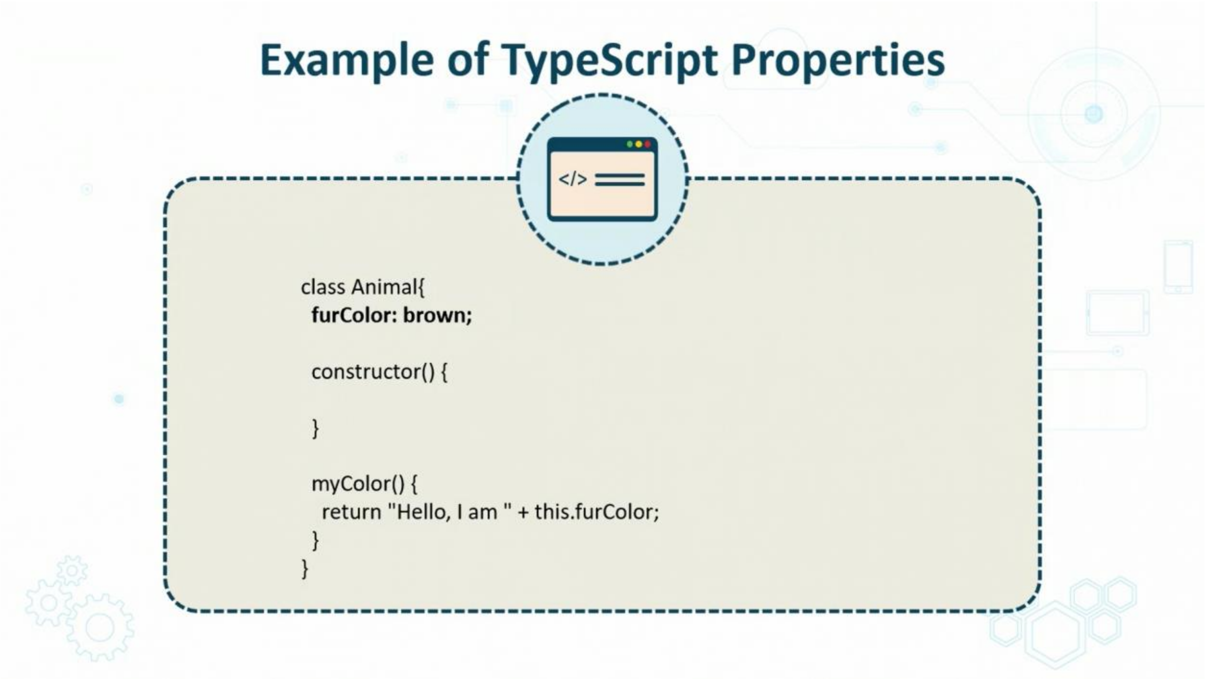
Protected

Properties



- Properties can be added to classes
- Access modifiers can be added to the TypeScript properties
- Access modifiers control the visibility of the data members
- The concept is named "encapsulation"

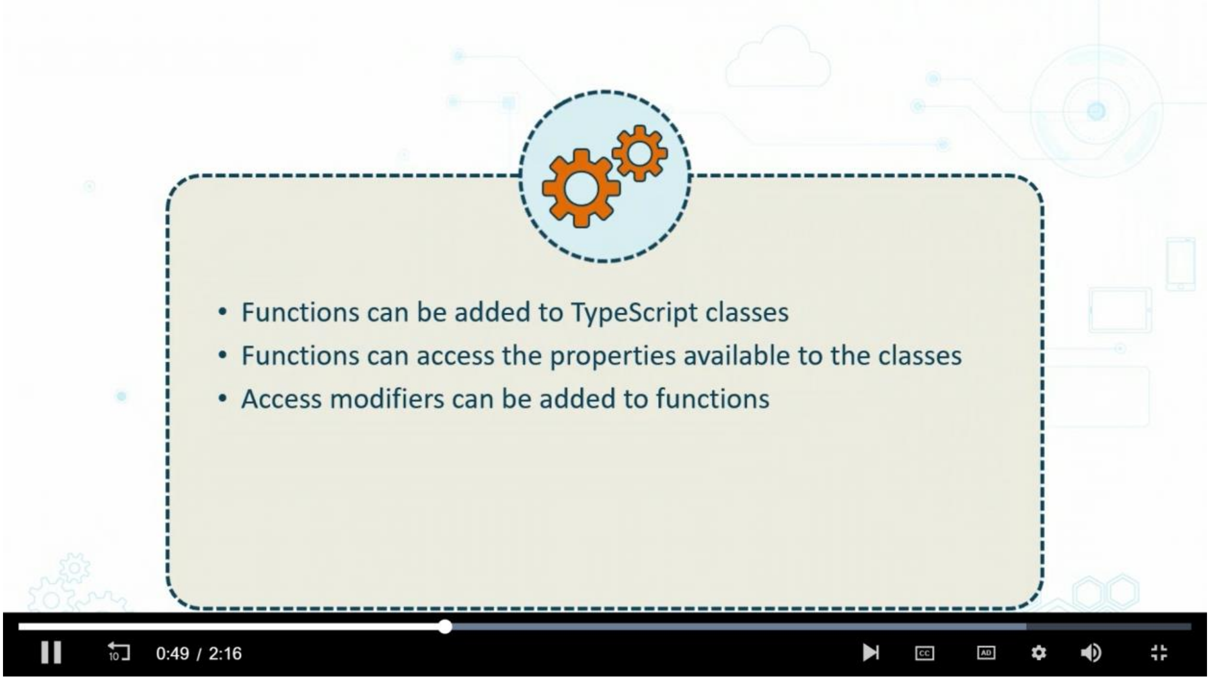
Private is encapsulated



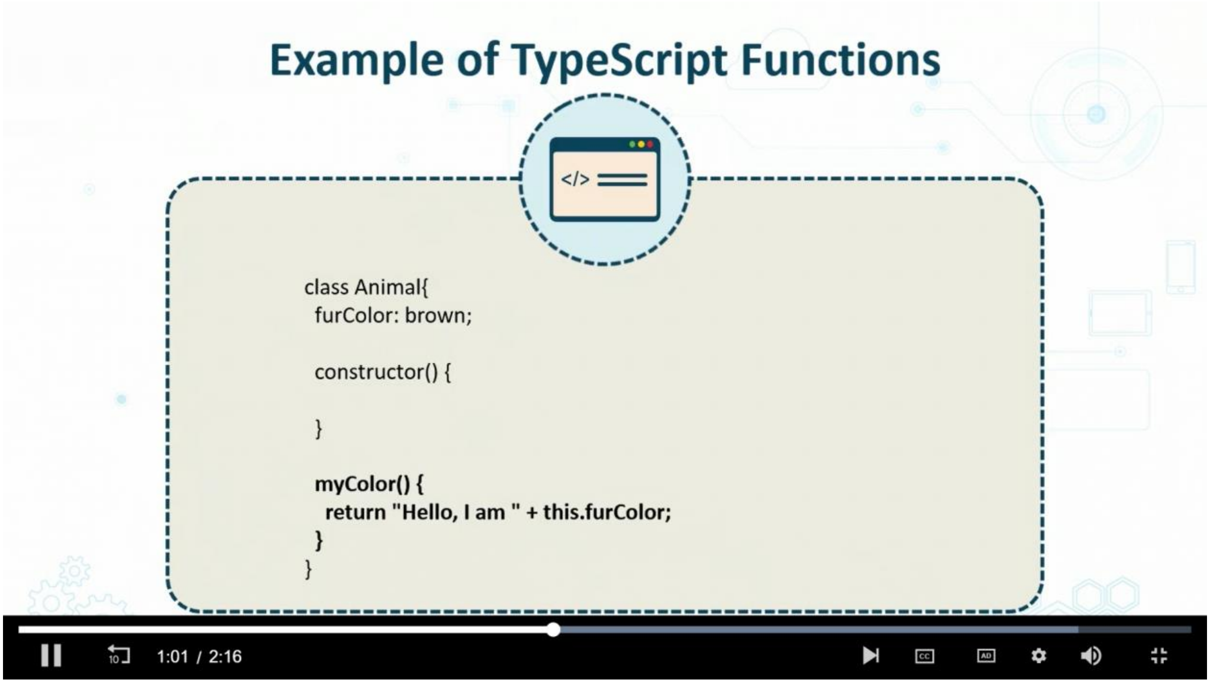
Example of TypeScript Properties

```
class Animal{
  furColor: brown;

  constructor() {

  }

  myColor() {
    return "Hello, I am " + this.furColor;
  }
}
```

Adding functions in Typescript classes



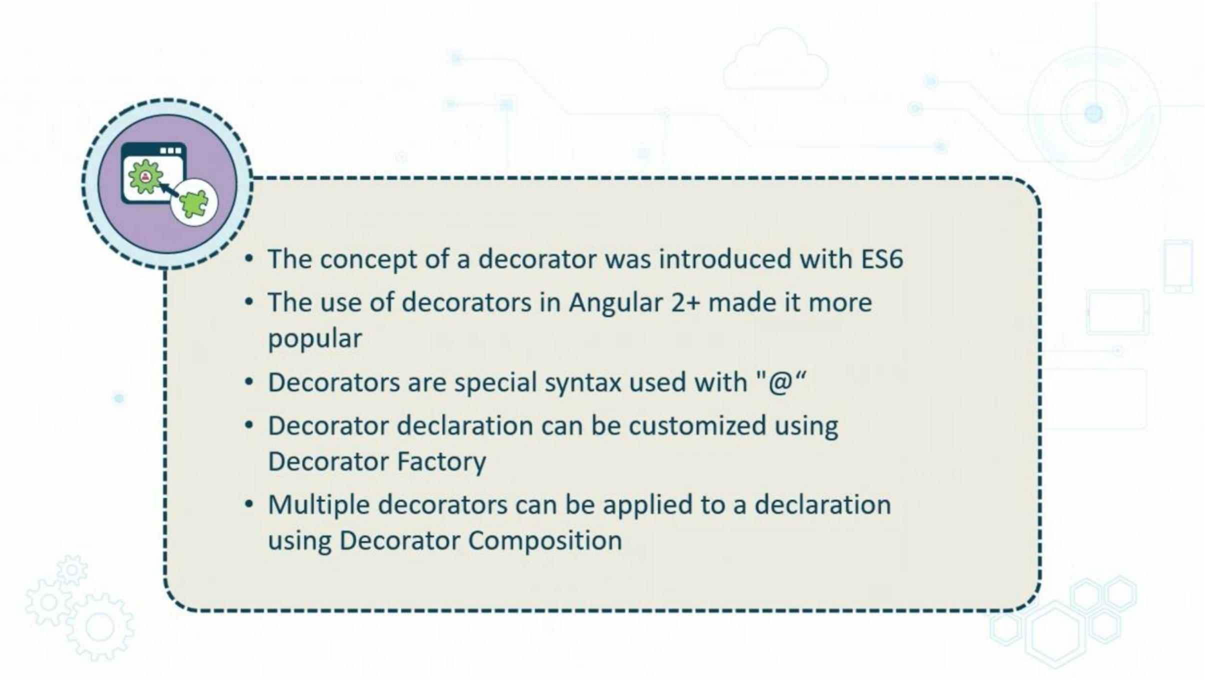
- Functions can be added to TypeScript classes
- Functions can access the properties available to the classes
- Access modifiers can be added to functions



Example of TypeScript Functions

```
class Animal{
  furColor: brown;

  constructor() {

  }

  myColor() {
    return "Hello, I am " + this.furColor;
  }
}
```

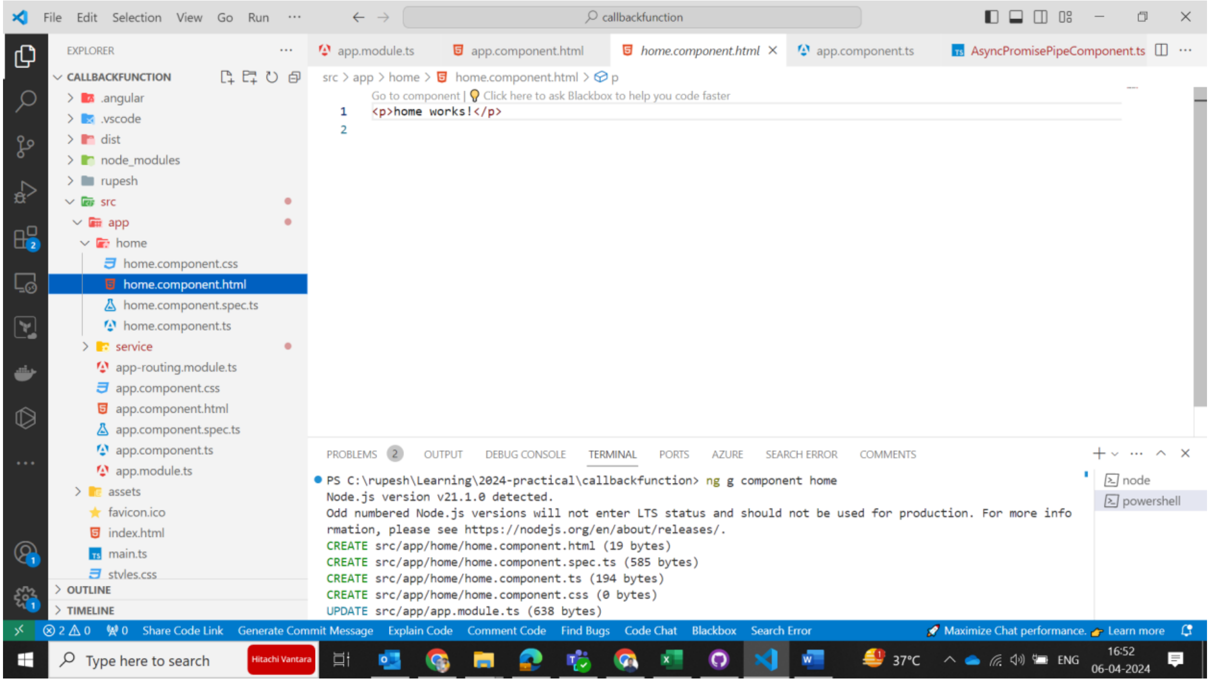
Defining Decorators in Typescript



- The concept of a decorator was introduced with ES6
- The use of decorators in Angular 2+ made it more popular
- Decorators are special syntax used with "@“
- Decorator declaration can be customized using Decorator Factory
- Multiple decorators can be applied to a declaration using Decorator Composition

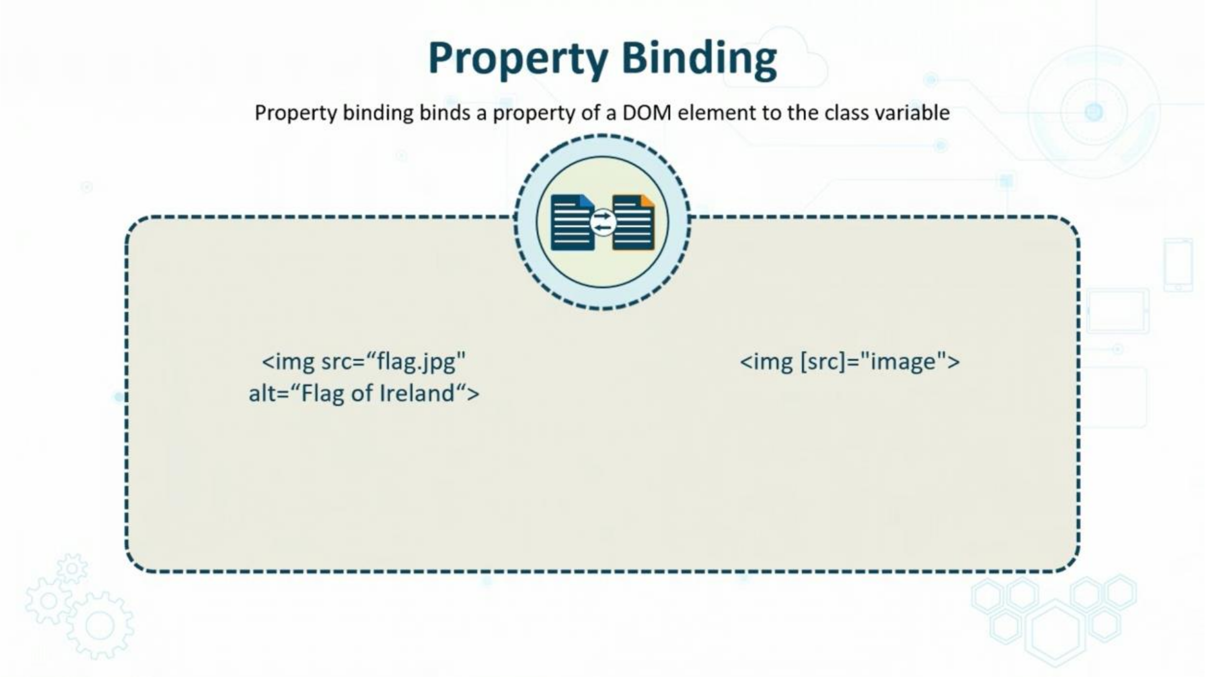
Creating components

Ng g component home



```html
<p>home works!</p>
```

```
PS C:\rupesh\Learning\2024-practical\callbackfunction> ng g component home
Node.js version v21.1.0 detected.
Odd numbered Node.js versions will not enter LTS status and should not be used for production. For more info
rmation, please see https://nodejs.org/en/about/releases/.
CREATE src/app/home/home.component.html (19 bytes)
CREATE src/app/home/home.component.spec.ts (585 bytes)
CREATE src/app/home/home.component.ts (194 bytes)
CREATE src/app/home/home.component.css (0 bytes)
UPDATE src/app/app.module.ts (638 bytes)
```

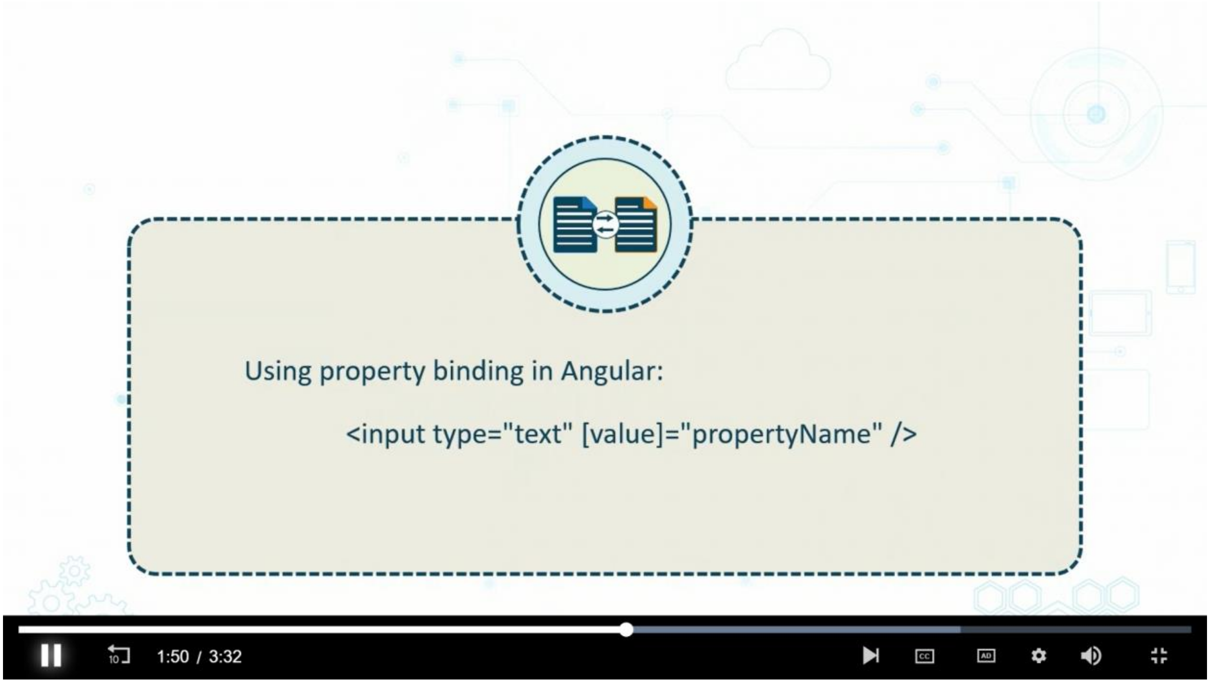


Property Binding

Property binding binds a property of a DOM element to the class variable

<img src="flag.jpg"
alt="Flag of Ireland">

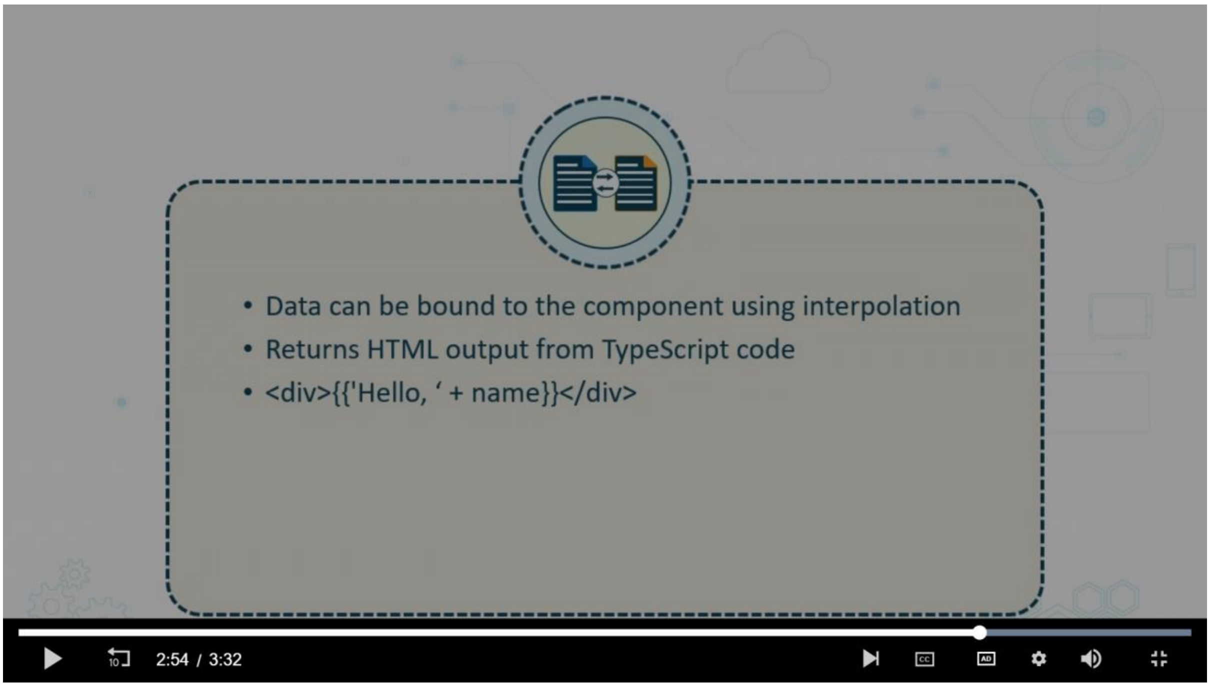
<img [src]="image">



Using property binding in Angular:

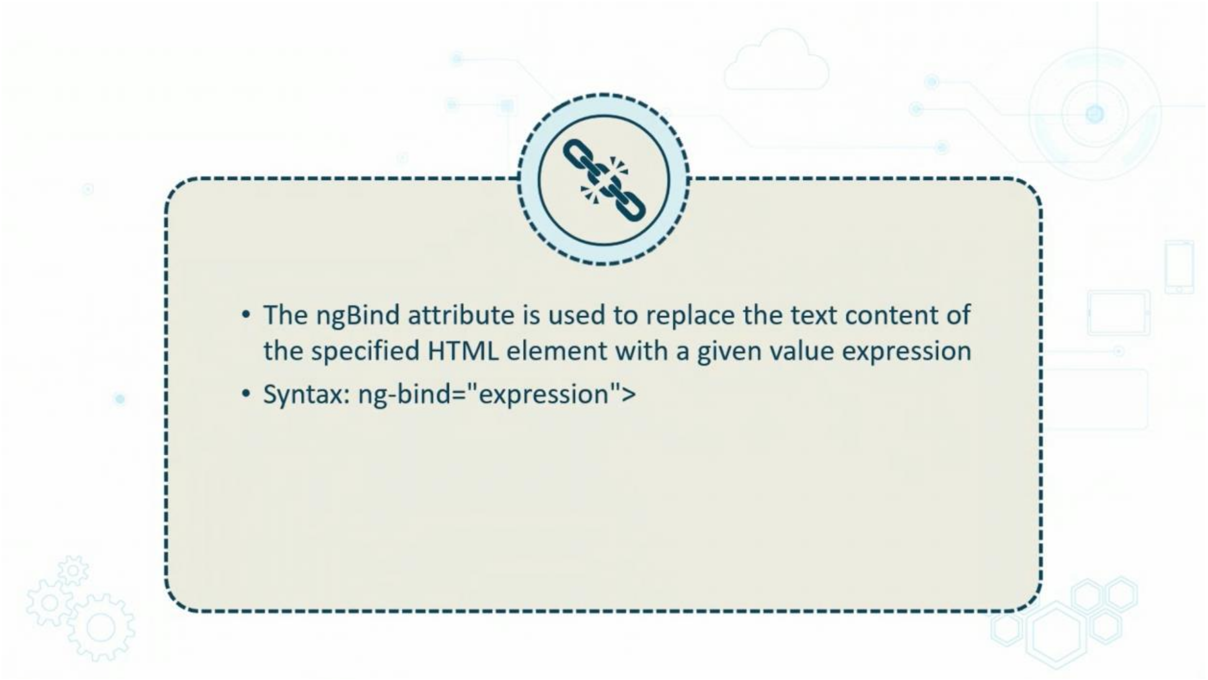
```
<input type="text" [value]="propertyName" />
```

Interpolatin examples



- Data can be bound to the component using interpolation
- Returns HTML output from TypeScript code
- <div>{{'Hello, ' + name}}</div>

Ng-bind



- The ngBind attribute is used to replace the text content of the specified HTML element with a given value expression
- Syntax: ng-bind="expression">