

## 1. Introduction

- Show Power Point "Themes inside Framework Controls"
- This sample is based on the "TcHmiToggleStateControl" and extend it
- Explain different theme layers with the graphic
  - Themes in Framework Controls are based on the control layer

## 2. Add new theme inside the Framework Control

- Navigate to Themes folder and add a new folder "Custom"
- Copy Style.css of the base theme in this folder
- Change CSS properties for demo
  - Set the button background-color to "lightgreen"
  - Set the button font color to "black"

## 3. Open Description.json and add a new theme entry

- Scroll to themes
- Copy base entry and rename to custom entry:

c.

```
70  "Custom": {  
71    "resources": [  
72      {  
73        "name": "Themes/Custom/Style.css",  
74        "type": "Stylesheet",  
75        "description": "Custom theme style"  
76      }  
77    ]  
78  }
```

- Explain type of the resource

## 4. Build project and reload Desktop.view

## 5. Select project node and change the theme to Custom --> theme was detected automatically

## 6. Switch theme during runtime:

- Add "Custom" theme inside project (then it's available in the set theme function)
- Add two TcHmiButtons
- Button 1 loads the base theme and button two loads the custom theme (available under functions)
- Open live view and show functionality

## 7. Add CSS control theme for the Framework Control in the custom theme --> the files are automatically copied