

# Ionic Study

Day 8

# 오늘 할 것들

- Angular 알아보기

# Angular

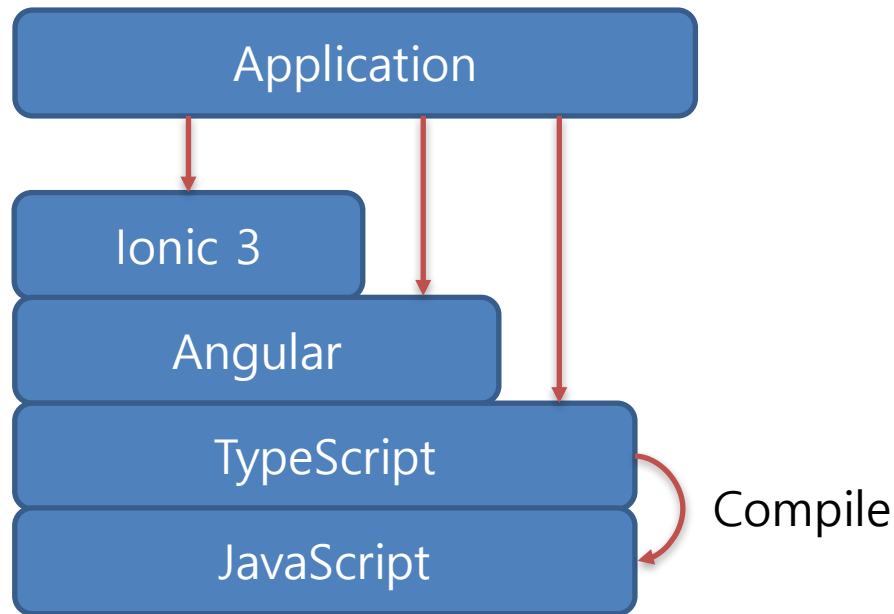
- Angular
  - 웹 응용프로그램 (웹앱)을 쉽게 만들기 위한 프론트엔드 플랫폼.
  - 컴포넌트 기반 구조 : 모든것이 컴포넌트
  - 뷰 (HTML+CSS) + 스크립트 (타입스크립트) => 웹앱

# Angular

- Angular의 대표적인 특징!
  - two way data binding
    - 모델(데이터를 조작하는 컴포넌트)과 뷰(브라우저에 표시하는 컴포넌트) 사이에 데이터가 자동으로 동기화.
  - interpolation
    - 컴포넌트의 속성값을 표시하기 위한 방법. 뷰 컴포넌트에서 {{ }}를 이용하여 속성의 바인딩이 가능.

# Angular

- 그렇다면 우리는 왜 이것 해야할까?



Ref) 반갑다 Ionic P78, Fig 3-1

- Ionic 앱은 Ionic 제공 API를 호출하면서 Angular의 기능을 사용하기 때문!

# Angular

- 컴포넌트 동적생성
  - html을 구성하기 위해서는 조건에 따라 Ionic 컴포넌트를 생성하거나 삭제할 필요가 있음.
  - structural directives (구조 디렉티브)
    - DOM(Document Object Model)의 요소(Element)를 추가, 삭제, 변경하여 뷰(html)의 레이아웃을 변경.
  - 요소의 속성명에 'ng'라는 접두사로 시작하는 경우가 많음.
    - ex: ngIf, ngSwitch...

# Angular - ngIf

- ngIf
  - 조건문을 만족하는 경우 해당 엘리먼트를 생성.
  - 요소의 속성에 \*ngIf="조건문"으로 조건 설정.
  - 조건문은 문법상 가능한 모든 문장의 사용이 가능.
    - 최종 형태가 true와 false로 나타나면 OK

# nglf

- nglf 실습하기

- 예제 소스 Git 저장소에서 가져오기

- \$ git clone https://github.com/KimTibber/day8\_ngif\_sample

- \$ cd day8\_ngif\_sample

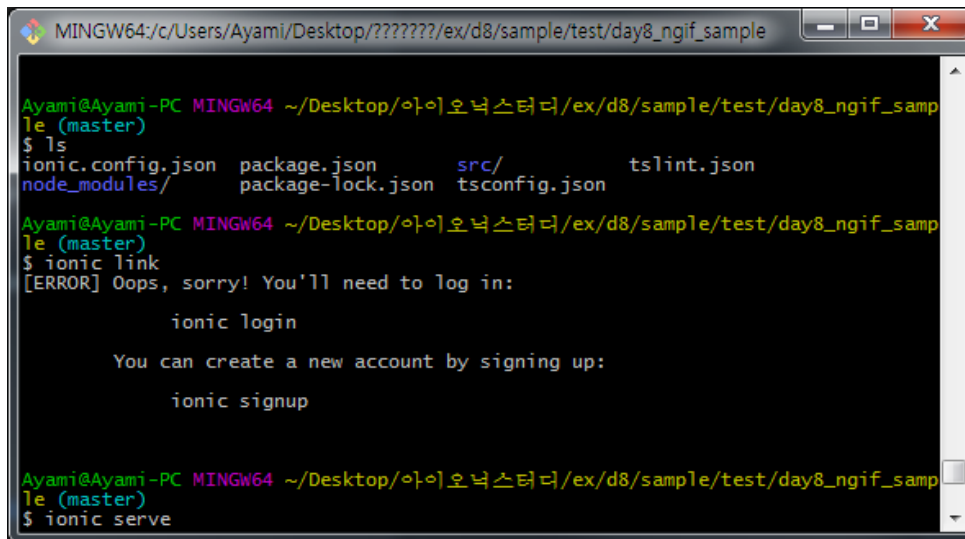
- \$ ionic link

- Y/n 질문시 Y 또는 그냥 엔터키 입력



# ngIf

- 이 에러는 상콤하게 무시해줍시다



```
MINGW64:/c/Users/Ayami/Desktop/???????/ex/d8/sample/test/day8_ngif_sample
Ayami@Ayami-PC MINGW64 ~/Desktop/아이오닉스터디/ex/d8/sample/test/day8_ngif_samp
le (master)
$ ls
ionic.config.json  package.json  src/  tslint.json
node_modules/     package-lock.json  tsconfig.json
Ayami@Ayami-PC MINGW64 ~/Desktop/아이오닉스터디/ex/d8/sample/test/day8_ngif_samp
le (master)
$ ionic link
[ERROR] Oops, sorry! You'll need to log in:

      ionic login

You can create a new account by signing up:

      ionic signup

Ayami@Ayami-PC MINGW64 ~/Desktop/아이오닉스터디/ex/d8/sample/test/day8_ngif_samp
le (master)
$ ionic serve
```

- ionic serve를 입력하여 프로젝트 확인

# ngIf

- **src/pages/login/login.html**
  - 입력받은 username과 password를 화면에 출력  
(button ion-button... 아래에 입력하세요)

```
18 .....<button ion-button (click)="login()">로그인!</button>
19 .....
20 .....<ion-list>
21 .....  <ion-item *ngIf="username != undefined && username.length >
           0">
22 .....    <p>사용자 이름 : {{username}}</p>
23 .....  </ion-item>
24 .....  <ion-item *ngIf="password != undefined && password.length >
           0">
25 .....    <p>비밀번호 : {{password}}</p>
26 .....  </ion-item>
27 .....</ion-list>
28 </ion-content>
```

- 실행해보면 입력하자마자 무언가가 돌아감... 이것이 바로 two way binding!

# Angular - ngSwitch

- ngSwitch
  - 자바스크립트의 switch 표현식과 동일한 역할을 수행.
  - 상위 요소에 ngSwitch 속성을 지정 : switch
  - 하위 요소에 ngSwitchCase 속성으로 case별 생성 조건을 정의 : case
  - ngSwitchCase에 모두 부합하지 않은 경우  
ngSwitchDefault 속성 지정 가능 (이 경우는 생략 가능) :  
default

# ngSwitch

- ngSwitch 실습하기

- tabs 템플릿으로 Ionic project 생성하기

- \$ ionic start day8\_ngswitch\_sample tabs

- 물어보는 대답엔 늘 그렇듯이 무심하게 엔터키를 입력...

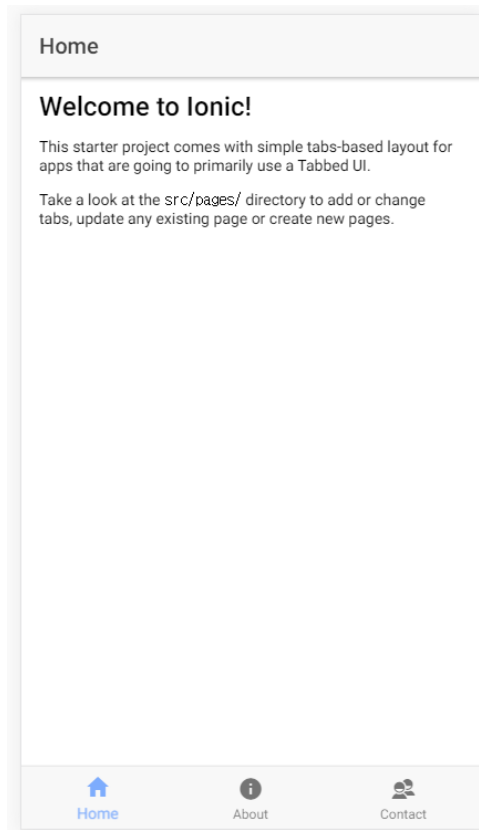
- (다만 Install the free Ionic Pro SDK and connect your app? 의 뉘시엔 넘어가지 말자)

- \$ cd day8\_ngswitch\_sample

- \$ ionic serve

# ngSwitch

- 탭 템플릿



- 탭 네비게이션을 가지는 템플릿
- 탭 네비게이션에 위치한 탭 버튼을 이용해 대응하는 세부 탭 페이지로 이동이 가능.
- 콘텐츠 영역을 탭으로 분할해서 각각 노출하게 되므로 멀티 콘텐츠를 지향하거나 구현하게 되는 세부기능의 명확한 분리가 가능한 프로젝트에 적합.

# ngSwitch

- **src/home/home.html**
  - 전체 내용을 아래와 같이 변경 후 저장

```
1  <ion-header>
2  .. <ion-navbar>
3  ... <ion-title>ngSwitch예제 {{appName}}</ion-title>
4  .. </ion-navbar>
5  </ion-header>
6
7  <ion-content padding>
8  .. <div padding>
9  ... <ion-segment [(ngModel)]="weather">
10 ... <ion-segment-button value="rainy">
11 ...   rainy
12 ... </ion-segment-button>
13 ... <ion-segment-button value="sunny">
14 ...   sunny
15 ... </ion-segment-button>
16 ... <ion-segment-button value="cloudy">
17 ...   cloudy
18 ... </ion-segment-button>
19 ... </ion-segment>
20 .. </div>
21 .. <div [ngSwitch]="weather">
22 ... <p *ngSwitchCase="'rainy'">우산을 준비해 주세요</p>
23 ... <p *ngSwitchCase="'sunny'">양산을 준비해 주세요</p>
24 ... <p *ngSwitchCase="'cloudy'">가볍게 외출하세요</p>
25 .. </div>
26 </ion-content>
27
```

# ngSwitch

- **src/home/home.ts**

- export Class HomePage... 아래에 해당 내용(붉은색 테두리) 추가 후 저장

```
1  import { Component } from '@angular/core';
2  import { NavController } from 'ionic-angular';
3
4  @Component({
5    selector: 'page-home',
6    templateUrl: 'home.html'
7  })
8  export class HomePage {
9    weather:string="sunny";
10
11    constructor(public navCtrl: NavController) {
12
13    }
14
15  }
```

# Angular - ngFor

- ngFor
  - 반복적인 요소 추가에 사용
  - 부모 요소에 `*ngFor="let 변수명 or 배열명; let i=인덱스변수;"` 속성을 주면 배열을 순회하여 자식 요소를 반복 추가.
  - 자식 요소 내부에 `{{ 변수명 or 인덱스변수 }}`를 이용하여 순회하는 값들에 접근이 가능.



# ngFor

- **src/about/about.html**
  - 전체 내용을 아래와 같이 변경 후 저장

```
1  <ion-header>
2    <ion-navbar>
3      <ion-title>
4        ngFor 예제
5      </ion-title>
6    </ion-navbar>
7  </ion-header>
8
9  <ion-content padding>
10    <div padding>
11      <h3>ngFor</h3>
12      <ion-list>
13        <ion-item *ngFor="let month of months; let i=index;">
14          <p>{{i}}: {{month}}</p>
15        </ion-item>
16      </ion-list>
17    </div>
18
19  </ion-content>
20
```

# ngFor

- **src/about/about.ts**

- export Class AboutPage... 아래에 해당 내용(붉은색 테두리) 추가 후 저장

```
1  import { Component } from '@angular/core';
2  import { NavController } from 'ionic-angular';
3
4  @Component({
5    selector: 'page-about',
6    templateUrl: 'about.html'
7  })
8  export class AboutPage {
9    months=[];
10
11    constructor(public navCtrl: NavController) {
12      this.months=["Jan", "Feb", "Mar", "April", "May", "Jun", "Jul", "Aug",
13        "Sep", "Oct", "Nov", "Dec"];
14    }
15  }
```

# Angular – 스타일 동적 변경

- html 요소의 스타일을 사용자의 입력에 따라 동적으로 변경하는 것이 가능.
- ngStyle
  - 한 요소에 여러 스타일 속성을 적용하는 경우 사용
    - 스타일이라 하면 당연히 css 스타일시트...
  - ngStyle="스타일 변수명"
    - 스타일 변수명은 ts파일 내부에서 object 형태로 정의.

# ngStyle

- src/contact/contact.html
  - 전체 내용을 아래와 같이 변경 후 저장

```
<ion-header>
  <ion-navbar>
    <ion-title>
      스타일 동적변경
    </ion-title>
  </ion-navbar>
</ion-header>

<ion-content>
  <div [ngStyle]="currentStyles">
    현재 적용되어있는 스타일
  </div>
  <ion-list>
    <ion-item>
      <ion-label>굵게</ion-label>
      <ion-checkbox [(ngModel)]="bold" (ionChange)="setCurrentStyles()"
        color="dark" checked="false"></ion-checkbox>
    </ion-item>
    <ion-item>
      <ion-label>이탤릭</ion-label>
      <ion-checkbox [(ngModel)]="italic" (ionChange)="setCurrentStyles()"
        color="dark" checked="false"></ion-checkbox>
    </ion-item>
    <ion-item>
      <ion-label>크게</ion-label>
      <ion-checkbox [(ngModel)]="large" (ionChange)="setCurrentStyles()"
        color="dark" checked="false"></ion-checkbox>
    </ion-item>
  </ion-list>
</ion-content>
```

# ngStyle

- **src/contact/contact.ts**
  - ContactPage 클래스 내부의 내용을 아래와 같이 변경 후 저장

```
export class ContactPage {  
  bold:boolean=false;  
  italic:boolean=false;  
  large:boolean=false;  
  
  currentStyles = {  
    'font-style': 'normal',  
    'font-weight': 'normal',  
    'font-size': '12px'  
  };  
  
  constructor(public navCtrl: NavController) {  
  
  }  
  
  setCurrentStyles() {  
    this.currentStyles = {  
      'font-style': (this.italic ? 'italic' : 'normal'),  
      'font-weight': (this.bold ? 'bold' : 'normal'),  
      'font-size': (this.large ? '24px' : '12px')  
    };  
  }  
}
```

# Angular – 스타일 동적 변경

- 인라인 속성 변경
  - 요소에 `style.attribute="value"` 속성을 추가하여 html 내에서 스타일 변경이 가능.
  - ex) `<div [style.font-size]="12px">`
- 요소 숨기기
  - 컴포넌트 생성 후, 보여지거나 숨겨지게 할 수 있는 지시자.
  - 요소에 `hidden="boolean"` 속성을 추가하여 제어 가능
  - ex) `<hr [hidden]="(a > b)">`
    - 지정한 조건문이 true일 경우 해당 요소는 숨겨짐. 반대로 false일 경우 해당 요소는 나타남

# 인라인 스타일

- **src/contact/contact.html**
  - `</ion-list>` 밑에 추가

```
<br><br>
<div [style.font-size]="fontSize">인라인 스타일!</div>
<ion-segment [(ngModel)]="fontSize" (ionChange)="change()">
  <ion-segment-button value="x-large">x-large</ion-segment-button>
  <ion-segment-button value="smaller">smaller</ion-segment-button>
</ion-segment>
```

# 인라인 스타일

- **src/contact/contact.ts**

- 멤버 변수 신규 추가

```
fontSize:string='x-large';
```

- 멤버 함수 신규 추가

```
change() {  
  console.log('fontSize='+this.fontSize);  
}
```



# hidden

- **src/contact/contact.html**
  - `</ion-segment>` 밑에 추가

```
<br><br>
<div [hidden]="hiddenFlag">
  이거 보이나요?
</div>
<div>
  <button ion-button (click)="hideClick()" [hidden]="hiddenFlag">숨기기</button>
  <button ion-button (click)="showClick()" [hidden]="!hiddenFlag">나타내기</button>
</div>
```

# 인라인 스타일

- **src/contact/contact.ts**

- 멤버 변수 신규 추가

```
hiddenFlag:boolean=false;
```

- 멤버 함수 신규 추가

```
hideClick() {  
    this.hiddenFlag = true;  
}  
  
showClick() {  
    this.hiddenFlag = false;  
}
```

오늘은 여기까지~

See you next day!