**Data Binding**

Photo by [Khara Woods](https://unsplash.com/@kharaoke?utm_source=medium&utm_medium=referral) on [Unsplash](https://unsplash.com?utm_source=medium&utm_medium=referral" \t "_blank)

**What is Databinding?**

Template (HTML) နဲ့ Business Logic (TypeScript Code) တို့အကြား အပြန်အလှန် ဆက်သွယ်နိုင်တဲ့ နည်းလမ်းဖြစ်တယ်။

**Databinding လေးမျိုး**

**String Interpolation**

String Interpolation ဆိုတာ output ထုတ်ပြချင်တဲ့ data ကို string အဖြစ်သို့ convert လုပ်လိုက်ပြီး HTML မှာပြသနိုင်အောင် လုပ်ဆောင်ပေးတာပဲ ဖြစ်ပါတယ်။ အလွယ်ပြောရရင် TypeScript က data ကို HTML မှာ ထုတ်ပြချင်ရင် သုံးတယ်။

***TypeScript***  
name: string = "Mg Mg";

***HTML***  
<h1>{{name}}</h1>  
<h1> {{'Hello ' + name}} </h1>

ဒီလိုမျိုး variable တွေကို ထုတ်ပြနိုင်ရုံသာမက expression တွေ method call တွေပါရေးနိုင်ပါတယ်။ ဥပမာ ternary expression နဲ့အခုလို ရေးနိုင်တယ်။

***TypeScript***  
status: string = "Online";  
show: boolean = true;

***HTML***  
<h1>{{show ? status : ''}} </h1>

Method call လုပ်ပြီး return ပြန်ပေးတဲ့ data ကိုပြချင်ရင် ဒီလိုရေးနိုင်တယ်။

***TypeScript***  
greet(): string {  
 return "Welcome";  
}

***HTML***  
<h1> {{greet()}} </h1>

**Property Binding**

TypeScript က data ကို HTML attribute မှာ bind လုပ်ချင်ရင် သုံးတယ်။ data တွေကို dynamic reload လုပ်ပေးနိုင်တယ်။ HTML attribute တွေအပြင် directive တွေ၊ custom component တွေကို data bind လုပ်တဲ့နေရာမှာလည်း အသုံးပြုပါတယ်။

***TypeScript***  
name: string = "Mg Mg";

***HTML***  
<input type="text" [value]="name" />

ဥပမာစမ်းသပ်ဖို့ servers.component.html မှာအခုလိုရေးပါ။

<button class="btn btn-primary"   
 **[disabled]="!allowNewServer"**>Add Server</button>  
<app-server></app-server>  
<app-server></app-server>

disabledဆိုတာ HTML attribute တစ်ခုပါ။ ဒီ attribute ကို data bind လုပ်ဖို့ [ ] ထဲထည့်လိုက်ပြီး ညာဘက်ခြမ်းမှာ variable ကို ထည့်လိုက်တာ ဖြစ်ပါတယ်။

servers.component.ts ထဲမှာတော့အခုလိုရေးပါ။

**allowNewServer: boolean = false;**  
 constructor() {  
 setTimeout(() => {  
 **this.allowNewServer = true;**  
 }, **3000**);  
 }

allowNewServer ဆိုတဲ့ boolean တစ်ခုကိုကြေငြာပြီး falseထည့်ထားပါတယ်။ constructor ထဲမှာ setTimeout()နဲ့ ၃စက္ကန့်ကြာရင် boolean ထဲကို true ထည့်ပေးဖို့ရေးထားပါတယ်။ run ကြည့်လိုက်ရင်တော့ button သည် run ပြီး ၃စက္ကန့်ကြာမှ နှိပ်လို့ရသွားမှာကို တွေ့ရမှာပါ။

dynamic updating template

ဒီနည်းနဲ့ TypeScript ကနေ template က data တွေကို လွယ်ကူစွာ manipulate လုပ်နိုင်မှာဖြစ်ပါတယ်။

**Event Binding**

HTML က user event တွေကို TypeScript ကကိုင်တွယ်ဖို့အတွက်သုံးတယ်။

***HTML***  
<button (click)="alert()">Click Me</button>

***TypeScript***  
alert() {  
 ...  
}

စမ်းသပ်ဖို့အတွက် servers.component.html မှာအခုလိုရေးပါ။

<button  
 class="btn btn-primary"  
 [disabled]="!allowNewServer"  
 **(click)="onCreateServer()"**  
>  
 Add Server  
</button>  
<p>{{ **serverCreationStatus** }}</p>  
<app-server></app-server>  
<app-server></app-server>

Add Server button ကိုနိုပ်ရင် onCreateServer() method ကို ခေါ်ပေးစေဖို့အတွက် (click) ဆိုတဲ့ event binding ကိုသုံးပြီး ရေးထားတာကို တွေ့ရမှာပါ။ <p> ထဲမှာ serverCreationStatusvariable ကို string interpolation သုံးပြီး ထုတ်ထားပါတယ်။ ပြီးရင်တော့ servers.component.ts မှာအခုလိုရေးလိုက်ပါ။

serverCreationStatus: string = 'No server was created!';

onCreateServer() {  
 this.serverCreationStatus = 'Server was created!';  
 }

serverCreationStatusvariable ကိုကြေငြာပြီး onCreateSever()method ကိုခေါ်တဲ့ အခါ variable ထဲက တန်ဖိုးကို ပြောင်းလဲလိုက်တာ ဖြစ်ပါတယ်။ ပြီးရင်တော့ run ကြည့်လိုက်ပါ။ button enable ဖြစ်သွားလို့ click နိုပ်လိုက်တဲ့အခါ <p> မှာထုတ်ထားတဲ့ စာက ပြောင်းလဲသွားတာ တွေ့ရမှာ ဖြစ်ပါတယ်။

Event Binding

**Two-way Databinding**

TypeScript နဲ့ HTML နှစ်ဘက်လုံးစလုံး data အပြန်အလှန် ချိတ်ဆက် လုပ်ဆောင်ဖို့အတွက် သုံးတယ်။ အသုံးပြုဖို့အတွက် FormModulesကို import လုပ်ပေးထားဖို့လိုတယ်။

***HTML***  
<input type="text" [(ngModel)]="name" />

***TypeScript***  
name: string = "Mg Mg";

name variable ကို HTML ဘက်ကပြင်လိုက်ရင် TypeScript ဘက်မှာ update လုပ်ပေးသလိုပဲ TypeScript ဘက်ကပြင်လိုက်ရင်လည်း HTML ဘက်မှာ update လုပ်ပေးမှာဖြစ်ပါတယ်။

**Practice**

* Server Name ကို input box နဲ့တောင်းယူပြီး variable တစ်ခုထဲမှာသိမ်းထားပါ။
* Add Server ကိုနှိပ်လိုက်ရင် server create လုပ်ပြီးကြောင်း server name နဲ့တကွဖော်ပြပါ။

**Assignment**

* Username ကိုတောင်းယူပြီး output ပြန်ထုတ်ပြပါ။
* reset button တစ်ခုကို ထည့်ပါ။ button ကို input box ထဲမှာ data ရှိမှ နှိပ်လို့ရအောင်လုပ်ပါ။
* button ကိုနှိပ်ရင် username ကို reset လုပ်လိုက်ပါ။

**Previous:** [**Component Deep Dive**](https://medium.com/learn-ng/component-deep-dive-7d93da97086a)

**Next:**