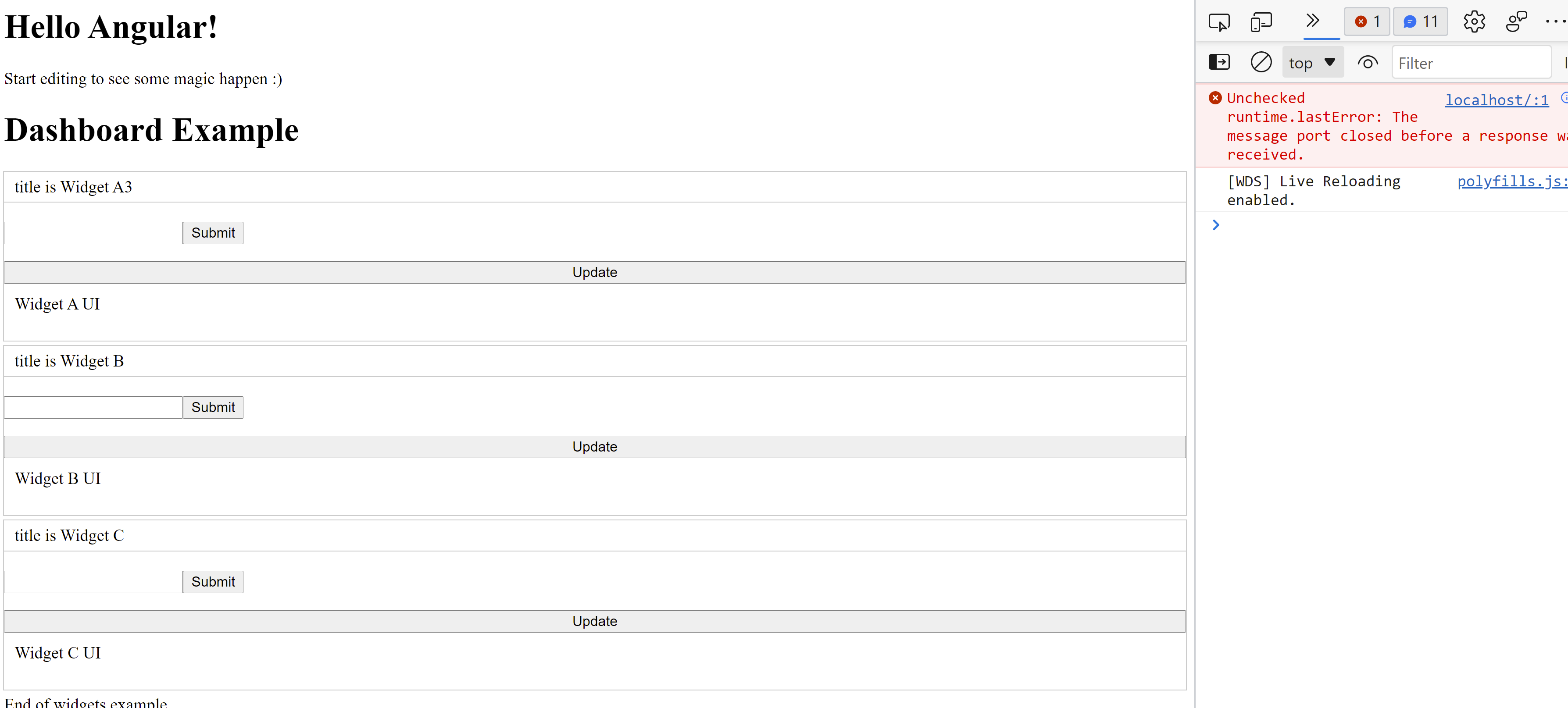
**Parent Child Input Output**



**HelloComponent**

import { Component, Input } from '@angular/core';

@Component({

  selector: 'hello',

  template: `<h1>Hello {{name}}!</h1>`,

  styles: [`h1 { font-family: Lato; }`]

})

export class HelloComponent  {

  @Input() name: string;

}

**AppModule**

import { NgModule } from '@angular/core';

import { BrowserModule } from '@angular/platform-browser';

import { FormsModule } from '@angular/forms';

import { AppComponent } from './app.component';

import { WidgetAComponent } from './widgets/widgetA.component';

import { WidgetBComponent } from './widgets/widgetB.component';

import { WidgetCComponent } from './widgets/widgetC.component';

import { WidgetComponent } from './widget/widget.component';

import { HelloComponent } from './hello.component';

@NgModule({

  imports:      [ BrowserModule, FormsModule ],

  declarations: [ AppComponent, HelloComponent, WidgetComponent, WidgetAComponent, WidgetBComponent, WidgetCComponent ],

  bootstrap:    [ AppComponent ]

})

export class AppModule { }

**App.Component.html**

  <hello name="{{ name }}"></hello>

<p>

  Start editing to see some magic happen :)

</p>

<h1>Dashboard Example</h1>

<app-widget [title]="titleA">

    <app-widget-a (titleEventA)="setATitle($event)"></app-widget-a>

</app-widget>

<app-widget [title]="titleB">

    <app-widget-b (titleEventB)="setBTitle($event)"></app-widget-b>

</app-widget>

<app-widget [title]="titleC">

    <app-widget-c (titleEventC)="setCTitle($event)"></app-widget-c>

</app-widget>

End of widgets example

**App.Component.ts**

import { Component } from '@angular/core';

@Component({

  selector: 'my-app',

  templateUrl: './app.component.html',

  styleUrls: [ './app.component.css' ]

})

export class AppComponent  {

  name = 'Angular';

  titleA = '';

  titleB = '';

  titleC = '';

  setTitle(evTitle: string){

      alert('A title is ' + evTitle);

  }

  setATitle(evTitle: string){

      alert('A title in Widget is ' + evTitle);

      this.titleA = evTitle;

  }

  setBTitle(evTitle: string){

      alert('B title in Widget is ' + evTitle);

      this.titleB = evTitle;

  }

  setCTitle(evTitle: string){

      alert('C title in Widget is ' + evTitle);

      this.titleC = evTitle;

  }

}

**WidgetComponent**

<header>

      <div>title is {{ title }}</div>

</header>

<form #it="ngForm" (ngSubmit)="onSubmit(it)" novalidate>

    <br/><input name="newT" #it="ngModel" [(ngModel)]="newTitle">

    <button type="submit">Submit</button>

</form>

<button (click)="updateWidget()">Update</button>

<section>

  <ng-content [select]="WA"></ng-content>

  <ng-content [select]="WB"></ng-content>

  <ng-content [select]="WC"></ng-content>

</section>

import { Component, Input} from '@angular/core';

import { NgForm } from '@angular/forms';

@Component({

  selector: 'app-widget',

  templateUrl: './widget.component.html',

  styleUrls: ['./widget.component.css']

})

export class WidgetComponent {

    @Input() title = 'Hello world';

    newTitle='';

    constructor()

    {

    }

    updateWidget() {

        alert('update() method in widget component ' + this.title + 'ok');

        if (this.newTitle !== ''){

             this.title = this.newTitle;

        }

        return this.title;

    }

    onSubmit(fm: NgForm){

        console.log(fm.value);

    }

  }

**WidgetAComponent**

import { Component, Output, EventEmitter, ViewChild, AfterContentInit } from '@angular/core';

import { Observable, of } from 'rxjs';

import { Widget } from '../widget/widget';

@Component({

   selector: 'app-widget-a',

   template: `Widget A UI

   <br/><p WA="x"></p>

   `

})

export class WidgetAComponent implements Widget, AfterContentInit {

  @Output() titleEventA:EventEmitter<string> = new EventEmitter<string>();

  constructor(){  }

  getTitle(): string {

      return 'Widget A3';

  }

  update(): Observable<string> {

      let titleA = this.getTitle();

      alert('A title is ' + titleA);

      this.titleEventA.emit(titleA);

      return of(titleA);

  }

  ngAfterContentInit(){

      this.update();

  }

}

**WidgetBComponent**

import { Component, Output, EventEmitter, AfterContentInit } from '@angular/core';

import { Observable, of } from 'rxjs';

import { Widget } from '../widget/widget';

@Component({

  selector: 'app-widget-b',

  template: `Widget B UI

        <br/><p WB="x"></p>

  `,

})

export class WidgetBComponent implements Widget, AfterContentInit {

  @Output() titleEventB:EventEmitter<string> = new EventEmitter<string>();

  constructor(){}

  getTitle(): string {

      return 'Widget B';

  }

  update(): Observable<string> {

      alert('Updating Widget B...');

      let titleB = this.getTitle();

      this.titleEventB.emit(titleB);

      return of(titleB);

  }

  ngAfterContentInit(){

      this.update();

  }

}

**WidgetCComponent**

import { Component, Output, EventEmitter, AfterContentInit } from '@angular/core';

import { Observable, of } from 'rxjs';

import { Widget } from '../widget/widget';

@Component({

  selector: 'app-widget-c',

  template: `Widget C UI

  <br/><p WC="x"></p>

  `

})

export class WidgetCComponent implements Widget, AfterContentInit {

  @Output() titleEventC:EventEmitter<string> = new EventEmitter<string>();

  constructor(){}

  getTitle(): string {

      return 'Widget C';

  }

  update(): Observable<string> {

      alert('Updating Widget C...');

      let titleC = this.getTitle();

      this.titleEventC.emit(titleC);

      return of(titleC);

  }

  ngAfterContentInit(){

      this.update();

  }

}

After updating the input for A B and C

