OUTPUTS

Suggest improvements

Tweet

If you want to bind to particular event, you can use the new Event syntax in Angular 2, but what if you need your own custom event?

To create a custom event, we can use the new `@Output` decorator. Take the following component:

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

@Component({
  selector: 'user-profile',
  template: '<div>Hi, my name is {{user.name}}</div>'
})
export class UserProfile {
  constructor() {}
}
```

Let's import `Output` and `EventEmitter` and create our new event

```
import { Component, Output, EventEmitter } from '@angular/

@Component({
  selector: 'user-profile',
  template: '<div>Hi, my name is {{user.name}}</div>'
})
export class UserProfile {
  @Output() userUpdated = new EventEmitter();

  constructor() {
    // Update user
    // ...
    this.userUpdated.emit(this.user);
  }
}
```

Now when we used this component elsewhere in our app, we can bind the event that `user-profile` emits

```
  <user-profile (userUpdated)="handleUserUpdated($event)">
```

```
export class SettingsPage {
  constructor(){}

  handleUserUpdated(user) {
    // Handle the event
  }
}
```

Built by the Ionic Team. Licensed under Apache 2.