MONASH University

# FIT2095 e-Business software technologies S2 2019



| | |
|---|---|
| **Started on** | Friday, 20 September 2019, 12:21 PM |
| **State** | Finished |
| **Completed on** | Friday, 20 September 2019, 12:57 PM |
| **Time taken** | 36 mins 2 secs |
| **Grade** | **9.53** out of 10.00 (**95**%) |

Print friendly format

Question **1**

Complete

Mark 9.53 out of 10.00

Question 1

- When creating a big application, or a product in which we want the structure to the system, the TypeScript is a better choice since this makes the project easy to maintain and creates it with as few bugs as possible. On the other hand, JavaScript would have the following immediate problems like lack of Object-Oriented Programming, lack of abstraction to provide base structure to a project, etc.

- TypeScript uses type hinting and typecasting so that a variable is given a certain number. Then the language will show the error while developing and resolve it even before it's deployed. This feature is not there in JavaScript.

2.

**Property Binding** simply means passing data from the component class (e.g. app.component.ts) and setting the value of a given element in the view (e.g. app.component.html). Property binding is one way. Eg:

Welcome to {{title}}!!!

The text between the braces (title) is often the name of a component property. Angular replace that name with the string value of the corresponding component property. In the example above, Angular evaluates the title the application title on the HTML.

**Event Binding** simply means passing data from an element to a component. Event binding is also one way like property binding. Some example of event binding are when users usually enter text into input boxes, pick items from lists and/or click buttons. The only way to know about a user action is to listen for certain events and that's where the event binding comes in the role.

Comment:
Q1=100% Q2=87% Q3=100%

Question **2**

Complete

Not graded

Question 2

<label for="width"> Width </label>

<input id="width" [(ngmodel)] = "width" type="number">

<label for="Height"> Width </label>

<input id="Height" [(ngmodel)] = "Height" type="number">

Area of rectange is: {{width}}*{{Height}}

Question **3**

Complete

Not graded

Question 3

**app.component.html**

Items:

<input [(ngModel)]="item" />

<br>

<button (click)="newItem()">Add Item</button>

<button (click)="clearItems()">Clear Items</button>

<br>

<ul *ngFor="let item of data">

<li>{{item}}</li>

</ul>

**app.component.ts**

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

@Component({

selector:'app-root',

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

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

})

exportclass AppComponent {

data =[];

item ="";

newItem(){

this.data.push(this.item);

}

clearItems(){

this.data=[];

}

}

Question **4**

Not answered

Not graded

Question 4

Question **5**

Not answered

Not graded

## Question 5

Question **6**

Not answered

Not graded

## Question 6

Question **7**

Not answered

Not graded

## Question 7

Question **8**

Not answered

Not graded

## Question 8

◀ Week 8 Workshop Quiz-Clean

Jump to...

Week 8 Pre-Reading Quiz ▶