## TABLE OF CONTENTS

**CHAPTER 1 : INTRODUCTION**

**1.1**  Introduction to Web Development

**1.2** Software Used

**CHAPTER 2: TRAINING WORK UNDERTAKEN**

**2.1.0 HTML**

**2.1.1** HTML Introduction

**2.1.2** HTML Page Structure

**2.1.3** Tags and Elements

**2.1.4** Text Formatting

**2.1.5** Hyperlink

**2.1.6 Cascading Style Sheets(CSS)**

**2.1.7** Box Model

**2.1.8** Fonts

**2.1.9** Navigation Bar

**2.1.10** Forms

**2.2.0 BOOTSTRAP**

**2.2.1** Breakpoints

**2.2.2** Jumbotrons and Glyphicons

**2.2.3** Thumbnails

**2.2.4** Forms

**2.3.0 STRUCTURED QUERY LANGUAGE (SQL)**

**2.3.1** SQL Introduction

**2.3.2** Introduction to DBMS

**2.3.3** Creating Database

**2.3.4** Data Definition Language(DDL)

**2.3.5** Alter and Drop

**2.3.6** Foreign Key & Truncate

**2.3.7** Data Manipulation Language(DML)

**2.4.0 PHP (Hypertext Preprocessor)**

**2.4.1** PHP Introduction

**2.4.2** Basics

**2.4.3** Variables, Data types and Operators

**2.4.4** Function, Array and Strings

**2.4.5** Loops

**CHAPTER** **3 PROJECT RESULT AND DISCUSSION**

**3.1** Coding

**3.2** Output

**3.3** Discussion

**CHAPTER 4**  **PROJECT CONCLUSION**

**4.1** Conclusion

**4.2** Future Scope

## CHAPTER 1: INTRODUCTION

### 1.1 WEB DEVELOPMENT

Web development is the work involved in developing a Web site for the Internet (World Wide Web) or an intranet (a private network). Web development can range from developing a simple single static page of plain text to complex Web-based

Internet applications (Web apps), electronic businesses, and social network services. A more comprehensive list of tasks to which Web development commonly refers, may include Web engineering, Web design, Web content development, client liaison, clientside/server-side scripting, Web server and network security configuration, and ecommerce development.

Among Web professionals, "Web development" usually refers to the main non-design aspects of building Web sites: writing markup and coding. Web development may use content management systems (CMS) to make content changes easier and available with basic technical skills.

For larger organizations and businesses, Web development teams can consist of hundreds of people (Web developers) and follow standard methods like Agile methodologies while developing Web sites. Smaller organizations may only require a single permanent or contracting developer, or secondary assignment to related job positions such as a graphic designer or information systems technician. Web development may be a collaborative effort between departments rather than the domain of a designated department. There are three kinds of Web developer specialization: front-end developer, back-end developer, and full-stack developer. Frontend developers are responsible for behavior and visuals that run in the user browser, while back-end developers deal with the servers.

#### 1.2 SOFTWARE USED

1. **NETBEANS** **:**

NetBeans is an integrated development environment (IDE) for Java. NetBeans allows applications to be developed from a set of modular software components called modules. NetBeans runs on Windows, macOS, Linux and Solaris. In addition to Java development, It has extensions for other languages like PHP, C, C++, HTML5, and JavaScript. Applications based on NetBeans, including the NetBeans IDE, can be extended by third party developers.

**BOOTSTRAP:**

Bootstrap is a free and open-source CSS framework directed at responsive,

Mobile-first front-end web development. It contains CSS- and (optionally)

JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components. Bootstrap is the seventh-most-starred project on GitHub, with more than 142,000 stars, behind freeCodeCamp (almost 312,000 stars) and marginally behind Vue.js framework.

**WAMPSERVER** **:**

WampServer refers to a solution stack for the Microsoft Windows operating system, created by Romain Bourdon and consisting of the Apache web server, OpenSSL for SSL support, MySQL database and PHP programming language.

## CHAPTER 2: TRAINING WORK UNDERTAKEN HTML & CSS , BOOTSTRAP, SQL, PHP

#### 2.1.1 HTML INTRODUCTION

HTML stands for HyperText Markup Language( **HTML** ) is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript.

HTML is used to design the structure of a web page. It is a set of instructions on how to display content on a web page. With HTML constructs, images and other objects such as Interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML documents are written in html editors. Since the html document is written in plain , we can use any text editor.

Ex : notepad, notepad++, sublime text, eclipse, netbeans etc.

#### 2.1.2 HTML PAGE STRUCTURE

<!DOCTYPE html>

<!-

Add your comments here

-->

<html>

<head>

<title> </title>

</head>

<body>

</body>

</html>

#### 2.1.3 TAGS AND ELEMENTS

**HTML TAGS :**

HTML tags are the hidden keyword within a web page that define how the browser must format and display the contents. Most of these tags have 2 parts , opening tag and closing tag.

<p> : Opening Tag </p> : Closing Tag

Opening Tag and Closing Tag together are called as a Container .

#### <html>

Tags are used to inform the browser how to format or display the text, except for the declaration of document type and comment. Comments can be included with html tags also.

#### <body>

It contains all the visible contents of the page. It may include texts, links and tables, or videos. <**div>**

It is used to create different sections in a web page.

#### 2.1.4 TEXT FORMATTING

There are six different heading tags.

<h1> I am Heading </h1>

<h2>I am Heading</h2>

Number defines the size of the text.

<i> : Tag to make text look italic.

<strong> : Tag to make text look bold.

<em>: Tag to make text look italic.

These tags help the browser know about the special importance of text , not just styling. It also helps in SEO purposes .

<u> : Tag element is used to underline text.

<mark> : Tag element is used to highlight text.

#### 2.1.5 HYPERLINKS

Any content , image or text can be linked to a new page. The text linked is called anchor text. Link is created using html <a>click here</a>link.

href=”hypertext reference” <a href = “ “ >click here</a> target=”\_blank”

<a href=”link” target=”\_blank”>

This will open in another tab.

target=”\_self”

<a href=”link” target=”\_self” >

This will open in the same tab.

#### 2.1.6 CASCADING STYLE SHEETS (CSS)

Cascading Style Sheets is a style sheet language used for describing the presentation of a document written in a markup language such as HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.

There are three ways of Styling :

1) Inline 2) Internal 3) External

#### 2.1.7 BOX MODEL

#### 

* Each and every element can be considered as a box.
* Innermost box called content contains content which could be image , text.
* Padding clears the area around the content. It is always transparent .
* Padding is surrounded by a border whose colour can be chosen.
* Margin clears the area outside the border. Default margin is 0.Margin is also transparently applied.

#### 2.1.8 FONTS

Properties :

1) Font Style 2) Font Variant 3) Font weight 4) Font Size 5) Line height 6) Font family Font size and Font family are mandatory.

For Ex : h4 { font: italic small-caps bolder 28px arial, sans-serif ; }

Comma is used in font family if there is unavailability of any font.

#### 2.1.9 NAVIGATION BAR

Navigation Bar and logo are together often referred to as a header. Navigation bar links different sections within a page.

Navigation links are added using the <nav> tag. Using the <ul> tag inside the <nav> tag we can create a navigation bar.

To create logo in navigation bar ;

<div id=”header”>

<a href=”#” class=”logo”>

<img src=”./img/logo.jpg” alt=”msd”

</a>

#### 2.1.10 FORMS

It collects information stored in databases or sent to the server. Using the <form> element to create form.

<form>

<input type=”text” placeholder =”Enter first Name” name=”First name”><br><br>

<input type=”submit”>

## BOOTSTRAP

### 2.2.1 BOOTSTRAP INTRODUCTION ( FRONT END DEVELOPMENT)

Bootstrap is a free and open-source CSS framework directed at responsive, mobile first front-end web development. It contains CSS and JavaScript-based design templates for typography, forms, buttons, navigation, and other components.

Bootstrap is one of the simplest to implement and use in the market. Its implementation is as simple as importing a CSS and using the classes available. Bootstrap’s responsiveness makes it all much simpler. It can intelligently sense the device’s resolution and screen width and adjust the content accordingly. Bootstrap is supported by the huge open source community present on GitHub. Any bugs or issues are resolved in no time for the releases.

Linking Javascript File(Jquery) :

Javascript is used to make the page more dynamic, to add effects , animation etc. Javascript files in the JS folder of bootstrap are used to add predefined javascript functions in the html code. Script element in html is used to embed script within an html document.

<script type=”text/javascript src=”bootstrap/js/bootstrap.min.js”></script>

★ Second method to link Bootstrap file :

“https://max.edn.bootstrapedn.com/bootstrap/3.3.7/css/bootstrap.min.js”>

“ https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery/1.12.4/jquery.min.js”>

“ https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js”

#### 2.2.2 BREAKPOINTS

**MULTIPLE CLASSES :**

It is one of the best ways of defining complex web pages layout and design. At times we have to make minor changes in an element while keeping everything else just the same. This can be achieved using multiple classes on the element we plan to make changes to. Use of multiple classes on an element is a core part of designing web pages using BOOTSTRAP , Bootstrap has a number of predefined classes.

**BREAKPOINTS IN BOOTSTRAP :**

DEVICES BREAKPOINT

Large Devices >=1200px

Medium Devices 992px to 1199px

Small Devices 768px to 991px

Extra Small Devices <768px

#### 2.2.3 JUMBOTRON AND GLYPHICONS

Jumbotron is a big box for calling extra attention to some special attention content or information. Class Jumbotron is used to create jumbotrons.

It is represented in grey box and round colors. It also enlarges the text inside it.

Jumbotron takes the full width of the element.

<body>

<div class =”container”>

<h1>This is a jumbotron</h1>

<div class=”jumbotron”

We can also insert tables, content, text etc inside a jumbotron.

GLYPHICONS : It is the icon displayed on any web page. <body>

<div class=”container”>

<h3>search</h3>

<h3 search <span class=”glyphicon glyphicon\_search”></span></h3>

#### 2.2.4 THUMBNAILS

Thumbnails are reduced versions of audio and videos.

Thumbnails take the available width.

<div class=”container”>

<h1>Thumbnails<h1>

<div class=”col-sm-4”>

<a href=”#”><img src=”img path” alt=”Responsive image”></a> Adding <a> tag to make thumbnails clickable.

Thumbnails are treated as links using anchor tags.

**.** thumbnail { padding:4px; border:1px solid #ddd; borderradius:4px; }

To add content in thumbnail ;

<div class=”content”>

<h3> </h3>

<p> </p>

</div>

#### 2.2.5 FORMS

FORM STYLING :

<div class=”container”>

<div class=”row”

<div class=”col-xs-4”>

<h1>Styling forms</h1>

By default form-control class can take the entire width for the element.

<form>

<div class=”form-group”>

<input type=”text” class=”form-control” name=”first\_name placeholder=”first\_name”>

</div>

.form-group { margin-bottom:5px;

}

To stop users from entering data add disabled attributes.

To make the first name and last name appear above the input field use <label> tag.

<div class=”form-group”>

<label=”first-name”>First Name</label>

<input type=”text” class=”form-control” name=”first\_name”>

**STYLING CHECKBOXES AND RADIO BUTTONS** :

<div class=”checkbox”> I want to order :

<label>

<input type=”checkbox” name=”food” value=”coffee” checked>Coffee

</label>

.label { padding-left:20px; marginbottom:20px; }

## STRUCTURED QUERY LANGUAGE((SQL)

### 2.3.1 SQL INTRODUCTION ( BACK END DEVELOPMENT)

SQL is a domain-specific language used in programming and designed for managing data held in a relational database management system, or for stream processing in a relational data stream management system. A database server is a computer program that provides database services to other programs or computers, as defined by the client-server model.

.

### 2.3.2 INTRODUCTION TO DBMS

A system application for creating and managing your databases , access and can modify your Data . DBMS works in an efficient and secured manner.

DBMS Types :

1. Flat file database
2. Relational database : 1)hierarchical database

2)RDBMS

1. No SQL : 1)Key value 2)column oriented 3)document oriented 4)Graph DB

### 2.3.3 CREATING DATABASE

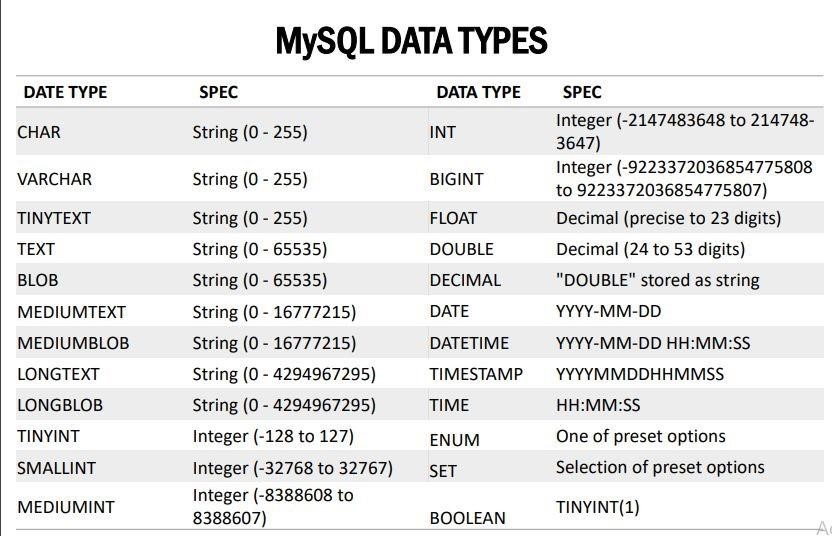
In technical terms; database is a set of multiple tables. Tables may be connected to each other by the concept of foreign keys.

INTRODUCTION TO SQL

Structured Query Language :

1. It is used to communicate with the database.
2. My SQL is an open source dbms.
3. My SQLi is an advanced version of My SQL.

**MYSQL DATABASES** :



### 3.3.4 DATA DEFINITION LANGUAGE(DDL)

1) Commands in SQL

Create command is used to create databases and tables.

For ex : CREATE database temp\_db;

$ always write database names in your query

Data contains actual values corresponding to each column of the table.

**PRIMARY KEY:**

Primary key is a number uniquely associated with each row or record.

For ex: CREATE TABLE temp\_db\_products(id INT NOT NULL , name

VARCHAR(30), category

INT(30), PRIMARY KEY(id));

### 3.3.5 ALTER AND DROP

We can alter the structure of the existing table.

Four keywords used with alter command are :

1)add 2)drop

3)change 4)modify

Ex: To add a column to an existing table.

ALTER table database-name.table-name ADD column-new data-type (length constraints) To add a column from an existing table.

ALTER table database-name.table-name DROP column-name.

**DROP COMMAND :**

Syntax :

DROP database-name.table-name

DROP database temp\_db;

### 3.3.6 FOREIGN KEY AND TRUNCATE

Mapping between two tables is done using foreign key.

A foreign key is a key used to link two tables together. This is sometimes also called a referencing key .

A Foreign Key is a column or a combination of columns whose values match a Primary Key in a different table.

To view relation views add storage to innodb operations.

Relation views provide functionality to link columns of one table to columns of another table.

**TRUNCATE :**

Truncate command deletes all the records inside the table.

Used when we only want to keep the structure of the table and delete the data. PhpMyadmin allows users to execute truncate commands without using the SQL command.

Data Definition Language(DDL)

1. Primary key
2. Alter command 3) Drop command
3. Foreign key
4. Truncate

### 3.3.7 DATA MANIPULATION LANGUAGE(DML)

A collection of those commands that help to INSERT, DELETE, MODIFY or pull out data from the database.

**INSERT :**

This command is used to insert data into the tables.

For ex : INSERT INTO database-name (name, category) values(‘ ‘,’ ‘)

**UPDATE :**

This command is used to update the existing data in tables.

For ex : UPDATE database-name SET first\_name =’ ‘ WHERE users.id= 1 ;

**DELETE :**

This command is used to delete an entire row from a table. To delete a row ; make sure it does not have a primary key or foreign key

For ex : DELETE FROM users WHERE id=’ ‘ ;

**SELECT :**

Select command is used to pull out the existing data from tables.

SELECT \* FROM ‘ ‘

Use \* to Display all the columns.

### 3.3.8 ASSIGNMENT

★TO CREATE THE DATABASE AND THE TABLES TO STORE THE USER DATA.

Items:

CREATE TABLE `items` (

`id` int(11) NOT NULL AUTO\_INCREMENT,

`name` varchar(255) NOT NULL,

`price` int(11) NOT NULL,

PRIMARY KEY (`id`)

); Users:

CREATE TABLE `users` (

`id` int(11) NOT NULL AUTO\_INCREMENT,

`name` varchar(255) NOT NULL,

`email` varchar(255) NOT NULL,

`password` varchar(255) NOT NULL,

`contact` varchar(255) NOT NULL,

`city` varchar(255) NOT NULL,

`address` varchar(255) NOT NULL,

PRIMARY KEY (`id`) ) ;Users\_items:

CREATE TABLE `users\_items` (

`id` int(11) NOT NULL AUTO\_INCREMENT,

`user\_id` int(11) NOT NULL,

`item\_id` int(11) NOT NULL,

`status` enum('Added to cart','Confirmed') NOT NULL,

PRIMARY KEY (`id`),

KEY `item\_id` (`item\_id`),

KEY `user\_id` (`user\_id`),

CONSTRAINT `users\_items\_ibfk\_1` FOREIGN KEY (`user\_id`) REFERENCES `users` (`id`),

CONSTRAINT `users\_items\_ibfk\_2` FOREIGN KEY (`item\_id`) REFERENCES `items` (`id`)

);

**1)LOGIN PAGE :**



**2)WORKING AREA :**



## PHP(HYPERTEXT PREPROCESSOR)

### 2.4.1 PHP INTRODUCTION ( BACK END DEVELOPMENT )

PHP is a server side scripting language which is used to perform operations on databases. In building dynamic pages and websites functionality. PHP code is usually processed on a web server by a PHP interpreter implemented as a module, a daemon or as a Common Gateway Interface (CGI) executable. On a web server, the result of the interpreted and executed PHP code which may be any type of data, such as generated HTML or binary image data – would form the whole or part of a HTTP response.

### 2.4.2 BASICS

PHP STRUCTURE :

<!DOCTYPE html>

<html>

<head>

<title><?php echo” “?> </title>

</head>

<body>

<?php

Echo” “?>

</body>

</html>

### 2.4.3 VARIABLES , DATA TYPES AND OPERATORS

To write variable name in php

$variable\_name

<body>

<?php

$var1=18;

$var2=12;

Echo gettype($var1);

?>

</body>

To perform addition :

<?php

$var1=18;

$var2=12;

$sum= $var1 + $var2 ;

Echo $sum;

?>

**CONCATENATION :**

Concatenation is an addition of combining two strings to make it one string.

<?php

$var1=”Hello”;

$var2=”Internshala”;

Echo $var1.$var2;

?>

### 2.4.4 FUNCTIONS , ARRAYS AND STRINGS

FUNCTIONS:

A function is an independent code that performs a particular task.

<?php

Function sum ($parameter1 , $parameter2)

{ $addition = $parameter1 + $parameter2;

return addition ;

}

?>

<body>

$var=1;

$var=2;

$sum=sum($var1, $var2);

Echo “ Sum of two variables is “. $sum.” “;

?>

If passed with function call, it will have that value and if not passed , it will have that value and if not passed it will use the default value.

**ARRAYS:**

An array is a special variable, which can hold more than one value at a time.

Stores integers, strings and other arrays etc.

<body>

<?php

$numbers = array(18,12)

$sum= $numbers[0] + $numbers[1]; echo”Sum of two variables is “.sum. “ “;

Echo “Length of the array is “. sizeof ($numbers)”; Array can be 1D, 2D and 3D.

**STRINGS :**

A string is a set of characters that can contain spaces and numbers.

<?php

$variable1=18; echo”The

value of variable1 is “$variable1”

To calculate string length ;

<?php

$string1= “ “;

$length\_of\_string . strlen($string1);

Echo $length\_of\_string

Double quotes inside single quotes will be considered as strings. Single quotes inside Quotes will be considered as strings.

### 2.4.5 LOOPS

A loop consists of two parts, a body of a loop and a control statement. The control statement is a combination of some conditions that direct the body of the loop to execute until the specified condition becomes false. The purpose of the loop is to repeat the same code a number of times.

THREE TYPES OF LOOPS:

1)FOR LOOP 2)WHILE LOOP 3)DO WHILE LOOP

**TO DO LIST**

## INTRODUCTION

To do list is a to-do-list and tasks manager for professionals, students and small business combining tasks, projects, comments, attachments, notification and more.

To do list lets users streamline their personal and team productivity efficiently.

Between the internet, cell phones, and all sorts of other technology, business today is entirely different than it has been. Projects are bigger, results are faster (even for the competition), and the corresponding workload associated with a successful company is larger than ever. To be sure, even relatively simple tasks require ample attentiveness and multi-faceted plans to complete.

That’s where to-do lists, as well as the general idea of breaking tasks down into smaller and easier-to-handle chunks, come into perspective.

# HTML CODE

|  |
| --- |
| <!DOCTYPE html>  <html lang="en">    <head>  <meta charset="UTF-8">  <meta name="viewport" content=  "width=device-width, initial-scale=1.0">    <link rel="stylesheet" href=  "https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css" integrity=  "sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T" crossorigin="anonymous">    <script src="app.js"></script>    <title>To Do List</title>  </head>    <body>  <header class="bg-success text-white p-5">  <div class="container">  <div class="row">  <div class="col-lg-12 col-md-12 col-sm-12">  <font face="Comic sans MS" size="11" color="black">  <strong>ToDo List</strong>  </font>    </div>  </div>  </div>  </header>    <div class="container mt-3">  <h2>Add Items</h2>    <label id="lblsuccess" class="text-success" style="display: none;">  </label>    <form id="addForm"> |
| <div class="row">  <div class="col-lg-7 col-md-7 col-sm-7">  <input type="text" onkeyup= "toggleButton(this, 'submit')" class="form-control" id="item"> </div>    <div class="col-lg-5 col-md-5 col-sm-5"> <input type="submit" class="btn btn-dark" id="submit" value="Submit" disabled>  </div>  </div>  </form>    <h3 class="mt-4">Tasks</h3>    <form id="addForm">  <ul class="list-group" id="items"></ul>  </form>  </div>  </body>  </html> |

# JAVA SCRIPT CODE

|  |
| --- |
| window.onload = () => { const form1 = document.querySelector("#addForm");    let items = document.getElementById("items"); let submit = document.getElementById("submit");  let editItem = null;    form1.addEventListener("submit", addItem); items.addEventListener("click", removeItem);  }; function addItem(e) {  e.preventDefault(); if (submit.value != "Submit") { console.log("Hello"); |

|  |
| --- |
| editItem.target.parentNode.childNodes[0].data = document.getElementById("item").value;  submit.value = "Submit"; document.getElementById("item").value = "";  document.getElementById("lblsuccess").innerHTML = "Text edited successfully";  document.getElementById("lblsuccess")  .style.display = "block";  setTimeout(function() { document.getElementById("lblsuccess")  .style.display = "none";  }, 3000);  return false;  } let newItem = document.getElementById("item").value; if (newItem.trim() == "" || newItem.trim() == null) return false; else document.getElementById("item").value = "";  let li = document.createElement("li"); li.className = "list-group-item";  let deleteButton = document.createElement("button");    deleteButton.className =  "btn-danger btn btn-sm float-right delete";  deleteButton.appendChild(document.createTextNode("Delete"));    let editButton = document.createElement("button");  editButton.className =  "btn-success btn btn-sm float-right edit";  editButton.appendChild(document.createTextNode("Edit"));    li.appendChild(document.createTextNode(newItem)); li.appendChild(deleteButton); li.appendChild(editButton);  items.appendChild(li);  } |
| function removeItem(e) {  e.preventDefault(); if (e.target.classList.contains("delete")) { if (confirm("Are you Sure?")) { let li = e.target.parentNode; items.removeChild(li); document.getElementById("lblsuccess").innerHTML = "Text deleted successfully";  document.getElementById("lblsuccess") .style.display = "block";  setTimeout(function() { document.getElementById("lblsuccess")  .style.display = "none";  }, 3000);  } } if (e.target.classList.contains("edit")) { document.getElementById("item").value =  e.target.parentNode.childNodes[0].data; submit.value = "EDIT"; editItem = e;  }  } function toggleButton(ref, btnID) { document.getElementById(btnID).disabled = false;  } |

**OUTPUT**

