Funfit

```java
package com.funfit.bean;


public class Batch {
        private int batch_id;
        private String batch_type;
        private String batch_time;


        public Batch() {
                super();
        }
        public Batch(String batch_type, String batch_time) {
                super();
                this.batch_type = batch_type;
                this.batch_time = batch_time;
        }
        public Batch(int batch_id, String batch_type, String batch_time) {
                super();
                this.batch_id = batch_id;
                this.batch_type = batch_type;
                this.batch_time = batch_time;
        }


        public int getBatch_id() {
                return batch_id;
        }


        public void setBatch_id(int batch_id) {
                this.batch_id = batch_id;
        }

        public String getBatch_type() {
```

```java
                return batch_type;

        }


        public void setBatch_type(String batch_type) {

                this.batch_type = batch_type;

        }


        public String getBatch_time() {

                return batch_time;

        }


        public void setBatch_time(String batch_time) {

                this.batch_time = batch_time;

        }


}
```
///

Participant.java

```java
package com.funfit.bean;


public class Participant {

        private int Participant_id;

        private int batch_id;

        private String Participant_name;

        private int Participant_age;

        private String Participant_email;

        private String Participant_phn;


        public Participant() {

                super();

        }

        public Participant(int batch_id, String participant_name, int participant_age,
```

```java
                String participant_email, String participant_phn) {
        super();
        this.batch_id = batch_id;
        this.Participant_name = participant_name;
        this.Participant_age = participant_age;
        this.Participant_email = participant_email;
        this.Participant_phn = participant_phn;
    }
    public Participant(int participant_id, int batch_id, String participant_name, int participant_age,
                String participant_email, String participant_phn) {
        super();
        this.Participant_id = participant_id;
        this.batch_id = batch_id;
        this.Participant_name = participant_name;
        this.Participant_age = participant_age;
        this.Participant_email = participant_email;
        this.Participant_phn = participant_phn;
    }
    public int getParticipant_id() {
        return Participant_id;
    }
    public void setParticipant_id(int participant_id) {
        Participant_id = participant_id;
    }
    public int getBatch_id() {
        return batch_id;
    }
    public void setBatch_id(int batch_id) {
        this.batch_id = batch_id;
    }
    public String getParticipant_name() {
        return Participant_name;
    }
    public void setParticipant_name(String participant_name) {
```

```java
                Participant_name = participant_name;
        }
        public int getParticipant_age() {
                return Participant_age;
        }
        public void setParticipant_age(int participant_age) {
                Participant_age = participant_age;
        }
        public String getParticipant_email() {
                return Participant_email;
        }
        public void setParticipant_email(String participant_email) {
                Participant_email = participant_email;
        }
        public String getParticipant_phn() {
                return Participant_phn;
        }
        public void setParticipant_phn(String participant_phn) {
                Participant_phn = participant_phn;
        }
}
```

//

DBAccess.java

```java
package com.funfit.resource;


import java.sql.Connection;
import java.sql.DriverManager;


public class DBAccess {
        static String url="jdbc:mysql://localhost:3306/FunFit";
        static String user_id = "root";
        static String password= "root";
```

```java
        public static Connection getConnection() {

                try {

                        Class.forName("com.mysql.cj.jdbc.Driver");

                        Connection con = DriverManager.getConnection(url,user_id,password);

                        return con;

                }

                catch(Exception e) {

                        System.out.println(e);

                }

                return null;

        }

}
```

//

BatchDAO.java

```java
package com.funfit.dao;

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import java.util.ArrayList;

import java.util.List;

import com.funfit.bean.Batch;

import com.funfit.bean.Participant;

import com.funfit.resource.DBAccess;

import com.mysql.cj.jdbc.CallableStatement;
```

```java
public class BatchDAO {

        Connection con;

        Statement stmt;

        PreparedStatement pstmt;


        public BatchDAO() {

                con =  DBAccess.getConnection();

         }

        public int createBatch(Batch batch) {

                try {

                        String sql="CALL createBatch(?,?);";////

                        java.sql.CallableStatement callprocedure = con.prepareCall(sql);

                        callprocedure.setString(1, batch.getBatch_type());

                        callprocedure.setString(2, batch.getBatch_time());

                        callprocedure.executeQuery();

                        return 1;

                } catch (SQLException e) {

                        System.out.println(e);

                        return -1;

                }

        }


        public int updateBatch(Batch batch) {

                try {

                        String sql="update Batches set batch_type=?, batch_time=? where batch_id=?";

                        pstmt = con.prepareStatement(sql);

                        pstmt.setString(1, batch.getBatch_type());

                        pstmt.setString(2, batch.getBatch_time());

                        pstmt.setInt(3, batch.getBatch_id());

                        return pstmt.executeUpdate();

                }catch (SQLException e) {

                        System.out.println(e);

                        return -1;                }
```

```java
        }

        public int deleteBatch(int b_id) {
                try {

                        String sql="delete from Batches where batch_id=?";

                        pstmt = con.prepareStatement(sql);

                        pstmt.setInt(1,b_id);

                        return pstmt.executeUpdate();

                }

                catch (SQLException e) {

                        System.out.println(e.getMessage());

                        return -1;

                }

        }

        public Batch viewBatch(int b_id) {
                try {

                        Batch b = new Batch();

                        PreparedStatement pstmt = con.prepareStatement("select * from Batches where
batch_id=?");

                        pstmt.setInt(1,b_id);

                        ResultSet rs = pstmt.executeQuery();

                        while(rs.next()) {

                                b.setBatch_id(rs.getInt(1));

                                b.setBatch_type(rs.getString(2));

                                b.setBatch_time(rs.getString(3));

                        }


                        return b;

                }

                catch (SQLException e) {

                        System.out.println(e);

                        return null;

                }

        }
```

```java
public List<Participant> viewBatchParticipants(int b_id) {
    try {
        List<Participant> li = new ArrayList<Participant>();
        PreparedStatement pstmt = con.prepareStatement("select * from Participants where batch_id=?");
        pstmt.setInt(1,b_id);
        ResultSet rs = pstmt.executeQuery();
        while(rs.next()) {
            Participant p = new Participant();
            p.setParticipant_id(rs.getInt(1));
            p.setBatch_id(rs.getInt(2));
            p.setParticipant_name(rs.getString(3));
            p.setParticipant_age(rs.getInt(4));
            p.setParticipant_email(rs.getString(5));
            p.setParticipant_phn(rs.getString(6));
            li.add(p);
        }
        return li;
    }catch (SQLException e) {
        System.out.println(e);
        return null;
    }
}

public List<Batch> getBatches() {
    try {
        stmt = con.createStatement();
        List<Batch> li = new ArrayList<Batch>();
        String query = "select * from batches";
        ResultSet rs = stmt.executeQuery(query);
        while(rs.next()) {
            Batch b = new Batch();
            b.setBatch_id(rs.getInt(1));
```

```java
                    b.setBatch_type(rs.getString(2));

                    b.setBatch_time(rs.getString(3));

                    li.add(b);

            }

            return li;

        }catch (SQLException e) {

                System.out.println(e);

                return null;

        }

    }

}
```
///

ParticipantDAO.java

```java
package com.funfit.dao;

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.Statement;

import com.funfit.bean.Participant;

import com.funfit.resource.DBAccess;

public class ParticipantDAO {

        Connection con;

        Statement stmt;

        PreparedStatement pstmt;

        public ParticipantDAO() {
```

```java
            DBAccess dbcon = new DBAccess();

            con =  dbcon.getConnection();

    }

    public int createParticipant(Participant participant) {

            try {

                    String sql="CALL createParticipant(?,?,?,?,?)";//

                    pstmt = con.prepareStatement(sql);

                    pstmt.setInt(1, participant.getBatch_id());

                    pstmt.setString(2, participant.getParticipant_name());

                    pstmt.setInt(3, participant.getParticipant_age());

                    pstmt.setString(4, participant.getParticipant_email());

                    pstmt.setString(5, participant.getParticipant_phn());

                    return pstmt.executeUpdate();

            } catch (Exception e) {

                    System.out.println(e);

                    return -1;

            }

    }


    public int updateParticipant(Participant participant) {

            try {

                    String sql="update Participants set participant_name=?, participant_age=?,
participant_email=?, participant_phn=? where participant_id=?";

                    pstmt = con.prepareStatement(sql);

                    pstmt.setString(1, participant.getParticipant_name());

                    pstmt.setInt(2, participant.getParticipant_age());

                    pstmt.setString(3, participant.getParticipant_email());

                    pstmt.setString(4, participant.getParticipant_phn());

                    pstmt.setInt(5, participant.getParticipant_id());

                    return pstmt.executeUpdate();

            }catch (Exception e) {

                    System.out.println(e);

                    return -1;                    }

    }
```

```java
public int deleteParticipant(int p_id) {
    try {
        String sql="delete from Participants where participant_id=?";
        pstmt = con.prepareStatement(sql);
        pstmt.setInt(1,p_id);
        return pstmt.executeUpdate();
    }
    catch (Exception e) {
        System.out.println(e);
        return -1;
    }
}
public Participant viewParticipant(int p_id) {
    try {
        Participant p = new Participant();
        PreparedStatement pstmt = con.prepareStatement("select * from Participants where participant_id=?");
        pstmt.setInt(1,p_id);
        ResultSet rs = pstmt.executeQuery();
        while(rs.next()) {
            p.setParticipant_id(rs.getInt(1));
            p.setBatch_id(rs.getInt(2));
            p.setParticipant_name(rs.getString(3));
            p.setParticipant_age(rs.getInt(4));
            p.setParticipant_email(rs.getString(5));
            p.setParticipant_phn(rs.getString(6));
        }
        return p;
    }
    catch (Exception e) {
        System.out.println(e);
        return null;
    }
```

```
        }


}
/////////////////////////////////////////////////////////
```

BatchService.java

```java
package com.funfit.service;

import java.util.List;

import com.funfit.bean.Batch;
import com.funfit.bean.Participant;
import com.funfit.dao.BatchDAO;

public class BatchService {
        BatchDAO dao = new BatchDAO();
        public String createBatchService(Batch b){
                if(dao.createBatch(b)>0) {
                        return "Batch created successfully";
                }
                else {
                        return "Sorry! An error occured while creating batch. Please try again!!!";
                }
        }


        public String updateBatchService(Batch b){
                if(dao.updateBatch(b)>0) {
                        return "Batch updated successfully";
                }
                else {
                        return "Sorry! An error occured while updating the batch details. Please try again!!!";
                }
        }
```

```java
        public String deleteBatchService(int bid){

                if(dao.deleteBatch(bid)>0) {

                        return "Batch deleted successfully";

                }

                else {

                        return "Sorry! An error occured while deleting the batch. Please try again!!!";

                }

        }


        public Batch viewBatchService(int bid){

                Batch b = dao.viewBatch(bid);

                return b;

        }


        public List<Participant> viewBatchParticipantsService(int bid) {

                return dao.viewBatchParticipants(bid)!=null?dao.viewBatchParticipants(bid):null;

        }
}
```

//

ParticipantService.java

```java
package com.funfit.service;


import com.funfit.bean.Participant;

import com.funfit.dao.ParticipantDAO;


public class ParticipantService {

ParticipantDAO dao = new ParticipantDAO();

        public String createParticipantService(Participant p){

                if(dao.createParticipant(p)>0) {

                        return "Participant added successfully";

                }
```

```java
            else {

                    return "Sorry! An error occured while adding Participant. Please try again!!!";

            }

    }


    public String updateParticipantService(Participant p){

            if(dao.updateParticipant(p)>0) {

                    return "Participant details updated successfully";

            }

            else {

                    return "Sorry! An error occured while updating the participant details. Please try
again!!!";

            }

    }


    public String deleteParticipantService(int pid){

            if(dao.deleteParticipant(pid)>0) {

                    return "Participant deleted successfully";

            }

            else {

                    return "Sorry! An error occured while deleting the participant. Please try again!!!";

            }

    }


    public Participant viewParticipantsService(int pid) {

            return dao.viewParticipant(pid)!=null?dao.viewParticipant(pid):null;

    }
}
```

///

BatchAddController.java

```java
package com.funfit.controller;


import java.io.IOException;
```

```java
import javax.servlet.ServletException;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;


import com.funfit.bean.Batch;

import com.funfit.service.BatchService;


/**
 * Servlet implementation class BatchAddController
 */
@WebServlet("/BatchAddController")
public class BatchAddController extends HttpServlet {
	private static final long serialVersionUID = 1L;

  /**
   * @see HttpServlet#HttpServlet()
   */
  public BatchAddController() {
    super();
    // TODO Auto-generated constructor stub
  }

	/**
	 * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
	 */
	protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
		// TODO Auto-generated method stub
		response.getWriter().append("Served at: ").append(request.getContextPath());
	}

	/**
```

```java
 * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)
 */
protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        // TODO Auto-generated method stub
        //int bid = Integer.parseInt(request.getParameter("batch_id"));
        String btype = request.getParameter("type");
        String btime = request.getParameter("time");
        Batch b = new Batch(btype,btime);
        BatchService bs = new BatchService();
        String result=bs.createBatchService(b);
        response.getWriter().println(result);
    }

}
```

///////////////////////////////////////

BatchUpdateController.java

```java
package com.funfit.controller;

import java.io.IOException;

import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import com.funfit.bean.Batch;
import com.funfit.bean.Participant;
import com.funfit.service.BatchService;
import com.funfit.service.ParticipantService;
```

```java
/**
 * Servlet implementation class BatchUpdateController
 */
@WebServlet("/BatchUpdateController")
public class BatchUpdateController extends HttpServlet {
        private static final long serialVersionUID = 1L;

  /**
   * @see HttpServlet#HttpServlet()
   */
  public BatchUpdateController() {
    super();
    // TODO Auto-generated constructor stub
  }


        /**
         * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
         */
        protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException,
IOException {
                // TODO Auto-generated method stub


        }


        /**
         * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)
         */
        protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException,
IOException {
                System.out.println(request.getParameter("_method"));

                if(request.getParameter("_method").equals("PUT")) {  // update ---> get other details of batch to
update using update.jsp

                        int pid = Integer.parseInt(request.getParameter("participant_id"));

                        ParticipantService ps = new ParticipantService();
```

```java
                Participant p = ps.viewParticipantsService(pid);

                RequestDispatcher dispatcher =
request.getRequestDispatcher("Participant/update_participant.jsp");

                request.setAttribute("participant", p);

                dispatcher.include(request,response);

        }
        else if(request.getParameter("_method").equals("POST")) {

                doPut(request, response); ///////UPDATE

        }
        else { ////DELETE

                doDelete(request, response);

        }

    }
    /**
     * @see HttpServlet#doPut(HttpServletRequest, HttpServletResponse)
     */
    protected void doPut(HttpServletRequest request, HttpServletResponse response) throws ServletException,
IOException {

            // update batch

            int bid = Integer.parseInt(request.getParameter("batch_id"));

            String btype = request.getParameter("type");

            String btime = request.getParameter("time");

            Batch b = new Batch(bid,btype,btime);

            BatchService bs = new BatchService();

            String result = bs.updateBatchService(b);

            response.getWriter().append(result);

    }


    /**
     * @see HttpServlet#doDelete(HttpServletRequest, HttpServletResponse)
     */
    protected void doDelete(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {

            //delete the batch

            int bid = Integer.parseInt(request.getParameter("batch_id"));
```

```java
                BatchService bs = new BatchService();

                String result = bs.deleteBatchService(bid);

                response.getWriter().append(result);

        }

}
```

/////////////////////////////////

BatchViewController.java

```java
package com.funfit.controller;

import java.io.IOException;

import java.util.List;


import javax.servlet.RequestDispatcher;

import javax.servlet.ServletException;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;


import com.funfit.bean.Batch;

import com.funfit.bean.Participant;

import com.funfit.service.BatchService;


/**
 * Servlet implementation class BatchViewController
 */
@WebServlet("/BatchViewController")
public class BatchViewController extends HttpServlet {

        private static final long serialVersionUID = 1L;
```

```java
    /**
     * @see HttpServlet#HttpServlet()
     */
    public BatchViewController() {
        super();
        // TODO Auto-generated constructor stub
    }


        /**
         * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
         */
        protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException,
IOException {
                // TODO Auto-generated method stub
                int bid = Integer.parseInt(request.getParameter("batch_id"));
                BatchService bs = new BatchService();
                Batch b = bs.viewBatchService(bid);
                List<Participant> trainee = bs.viewBatchParticipantsService(bid);
                RequestDispatcher dispatcher = request.getRequestDispatcher("Batch/batch_result.jsp");
                request.setAttribute("batch", b);
                request.setAttribute("trainee", trainee);
                dispatcher.include(request,response);
        }


        /**
         * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)
         */
        protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException,
IOException {
                // TODO Auto-generated method stub
                doGet(request, response);
        }

}
/////////////////////////////////////////////
```

ParticipantAddController.java

```java
package com.funfit.controller;

import java.io.IOException;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import com.funfit.bean.Batch;
import com.funfit.bean.Participant;
import com.funfit.service.ParticipantService;

/**
 * Servlet implementation class ParticipantAddController
 */
@WebServlet("/ParticipantAddController")
public class ParticipantAddController extends HttpServlet {
	private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public ParticipantAddController() {
        super();
        // TODO Auto-generated constructor stub
    }

	/**
	 * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
	 */
```

```java
	protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException,
IOException {
		// TODO Auto-generated method stub
		response.getWriter().append("Served at: ").append(request.getContextPath());
	}


	/**
	 * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)
	 */
	protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException,
IOException {
		// TODO Auto-generated method stub
		String p_name = request.getParameter("participant_name");
		int p_age = Integer.parseInt(request.getParameter("participant_age"));
		String p_email = request.getParameter("participant_email");
		String p_phn = request.getParameter("participant_phn");
		int p_batch_id = Integer.parseInt(request.getParameter("batch_id"));
		Participant p = new Participant(p_batch_id,p_name,p_age,p_email,p_phn);
		ParticipantService ps = new ParticipantService();
		String result=ps.createParticipantService(p);
		response.getWriter().println(result);
	}

}
```
///

ParticipantUpdateController.java

package com.funfit.controller;

import java.io.IOException;

import javax.servlet.RequestDispatcher;

import javax.servlet.ServletException;

import javax.servlet.annotation.WebServlet;

```java
import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;


import com.funfit.bean.Batch;

import com.funfit.bean.Participant;

import com.funfit.service.BatchService;

import com.funfit.service.ParticipantService;


/**
 * Servlet implementation class ParticipantUpdateController
 */
@WebServlet("/ParticipantUpdateController")
public class ParticipantUpdateController extends HttpServlet {
        private static final long serialVersionUID = 1L;

  /**
   * @see HttpServlet#HttpServlet()
   */
  public ParticipantUpdateController() {
    super();
    // TODO Auto-generated constructor stub
  }

        /**
         * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
         */
        protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException,
IOException {
                // TODO Auto-generated method stub
                response.getWriter().append("Served at: ").append(request.getContextPath());
        }

        /**
```

```java
    * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)
    */
    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException,
IOException {

            System.out.println(request.getParameter("_method"));

            if(request.getParameter("_method").equals("PUT")) {  // update ---> get other details of batch to
update using update.jsp

                    int pid = Integer.parseInt(request.getParameter("participant_id"));

                    ParticipantService ps = new ParticipantService();

                    Participant p = ps.viewParticipantsService(pid);

                    RequestDispatcher dispatcher =
request.getRequestDispatcher("Participant/update_participant.jsp");

                    request.setAttribute("participant", p);

                    dispatcher.include(request,response);


            }
            else if(request.getParameter("_method").equals("POST")) {

                    doPut(request, response);

            }
            else { ////DELETE

                    doDelete(request, response);

            }
    }


    /**
     * @see HttpServlet#doPut(HttpServletRequest, HttpServletResponse)
     */
    protected void doPut(HttpServletRequest request, HttpServletResponse response) throws ServletException,
IOException {

            int pid = Integer.parseInt(request.getParameter("participant_id"));

            String p_name = request.getParameter("participant_name");

            int p_age = Integer.parseInt(request.getParameter("participant_age"));

            String p_email = request.getParameter("participant_email");

            String p_phn = request.getParameter("participant_phn");

            int p_batch_id = Integer.parseInt(request.getParameter("batch_id"));

            Participant p = new Participant(pid,p_batch_id,p_name,p_age,p_email,p_phn);
```

```java
            ParticipantService ps = new ParticipantService();

            String result = ps.updateParticipantService(p);

            response.getWriter().append(result);

        }


        /**
         * @see HttpServlet#doDelete(HttpServletRequest, HttpServletResponse)
         */
        protected void doDelete(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {

                // delete participant

                int pid = Integer.parseInt(request.getParameter("participant_id"));

                ParticipantService ps = new ParticipantService();

                String result = ps.deleteParticipantService(pid);

                response.getWriter().append(result);

        }


}
```

//

ParticipantViewController.java

```java
package com.funfit.controller;


import java.io.IOException;

import java.util.List;


import javax.servlet.RequestDispatcher;

import javax.servlet.ServletException;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;


import com.funfit.bean.Batch;
```

```java
import com.funfit.bean.Participant;

import com.funfit.service.BatchService;

import com.funfit.service.ParticipantService;


/**
 * Servlet implementation class ParticipantViewController
 */
@WebServlet("/ParticipantViewController")
public class ParticipantViewController extends HttpServlet {
        private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public ParticipantViewController() {
      super();
      // TODO Auto-generated constructor stub
    }


        /**
         * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
         */
        protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException,
IOException {
                int pid = Integer.parseInt(request.getParameter("participant_id"));

                ParticipantService ps = new ParticipantService();

                Participant p = ps.viewParticipantsService(pid);

                RequestDispatcher dispatcher = request.getRequestDispatcher("Participant/participant_result.jsp");

                request.setAttribute("participant", p);

                dispatcher.include(request,response);

        }


        /**
         * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)
```

```java
	 */
	protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException,
IOException {
		// TODO Auto-generated method stub
		doGet(request, response);
	}


}
```

///
/////////////////

add_batch.jsp

```jsp
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8" import="java.sql.*" %>
<%! int new_batch ;Connection con=null; Statement stmt; ResultSet rs =null; %>
<%
try{
	Class.forName("com.mysql.cj.jdbc.Driver");
}catch(Exception e){
		out.println("An Exception occured at Driver Connection!" +e.getMessage());
		}
try{
	con  = DriverManager.getConnection("jdbc:mysql://localhost:3306/funfit", "root", "root");
}catch(SQLException e){
	out.println("An Exception occured at Database Connection!" +e.getMessage());
}
try{
	String query ="select * from id; ";
	stmt = con.createStatement();
	rs = stmt.executeQuery(query);
	if(rs.next()){
		new_batch = rs.getInt(1) +1 ;
		System.out.println(new_batch);
	}
```

```
}catch(Exception e){

        out.println("An Exception occured at logic" +e.getMessage());

}
finally{

                try{

                        if (rs != null) rs.close();

                if (stmt != null) stmt.close();

                if(con!=null)con.close();

                    }catch(Exception e){

                        out.println("An Exception occured at logic" +e.getMessage());

                    }

        } %>


<!DOCTYPE html>
<html>
<head>
   <meta charset='utf-8'>
   <meta http-equiv='X-UA-Compatible' content='IE=edge'>
   <title>Adding Batch</title>
   <meta name='viewport' content='width=device-width, initial-scale=1'>
   <link rel='stylesheet' type='text/css' media='screen' href='../style2.css'>
</head>
<body>
   <div class="container">
      <img src="../Funfit_img/Zumba_funfit.png" alt="Background Image" class="background-image" />
      <div class="card">
         <div class="form-container">
               <form  id="batch_form"  action="../BatchAddController"  method="POST">
               <div><top><h2><a href="../index.html">Home</a></h2></top></hr><h1><center id="tit">Add
Batch</center></h1></div><br>
               <input type="hidden" name="_method" value="POST">
                 <label>Batch No:</label></br><input type="text" id="batch_id" name="batch_id" value="<%=
new_batch %>" disabled></br>
                   <label for="batch-type">Batch Type:</label></br><input type="radio" id ="batch-type"
name="type" value="morning"><label>Morning(a.m)</label></hr>
```

```html
                    <input type="radio" id="batch-type" name="type"
value="evening"><label>Evening(p.m)</label></br>
                        <label id="batch-select" for="batch-select">Batch Time:</label>
                    <select id = "batch-select" name="time">
                        <option value = "7">7</option>
                        <option value = "8">8</option>
                        <option value = "9">9</option>
                    </select><br>
                    <input type="submit" id="submit" value="submit"></hr>
                    <input type="reset" id="reset" value="reset"/></br>
                </form>
            </div>
        </div>
    </div>
</body>
</html>
```

/////////////////////////////////

batch_result.jsp

```jsp
<%@page import="org.apache.jasper.tagplugins.jstl.core.ForEach"%>

<%@page import="com.funfit.bean.Participant"%>

<%@page import="java.util.List"%>

<%@ page language="java" contentType="text/html; charset=UTF-8"

    pageEncoding="UTF-8" import ="java.util.List,com.funfit.bean.*"%>

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Batch Details</title>

<style>

.styled-table{

        border-collapse: collapse;

        margin: 25px 0;

        font-size: 0.9em;

        min-width: 500px;
```

```css
        border-radius: 5px 5px 0 0;

        overflow: hidden;

        box-shadow: 0 0 20px rgba(0,0,0,0.15);

}

.styled-table thead tr{

        background-color: #009879;

        color: #ffffff;

        text-align: left;

        font-weight: bold;

}


.styled-table th,td{

        padding: 12px 15px;

}


.styled-table tbody tr{

        border-bottom: 1px solid #dddddd;

}


.styled-table tbody tr nth:nth-of-type(even){

        background-color: #f3f3f3;

}


.styled-table tbody tr nth:last-of-type{

        border-bottom: 2px solid #009879;

}


.styled-table tbody tr.active-row{

        font-weight: bold;

        color:

}
</style>
</head>
<body>
```

```html
<div class="table-container">
        <table class="styled-table">
          <thead>
            <tr>
              <th>Batch id</th>
              <th>Batch Type</th>
              <th>Batch Time</th>
            </tr>
          </thead>
          <tbody>
            <%
    // Retrieve the list from the request
                Batch b = (Batch)request.getAttribute("batch");
    // Check if the list is not null and display it
    if (b != null) {
      out.println("<tr>");
      out.println("<td>" + b.getBatch_id() + "</td>");
      out.println("<td>" + b.getBatch_type() + "</td>");
      out.println("<td>" + b.getBatch_time()+ "</td>");
      out.println("</tr>");
    } else {
      out.println("<p>No items found.</p>");
    }
  %>
            </tbody>
         </table>
</div>
<div class="table-container">
        <table class="styled-table">
          <thead>
            <tr>
              <th>Participant id</th>
              <th>Name</th>
              <th>Email</th>
```

```
                <th>Phone Number </th>


        </tr>

      </thead>

      <tbody>

       <%

    // Retrieve the list from the request
            List<Participant> items = (List<Participant>)request.getAttribute("trainee");

    // Check if the list is not null and display it

    if(items != null) {


      for(Participant item : items){
        out.println("<tr>");
         out.println("<td>" + item.getParticipant_id() + "</td>");

         out.println("<td>" + item.getParticipant_name() + "</td>");

         out.println("<td>" + item.getParticipant_email() + "</td>");

         out.println("<td>" + item.getParticipant_phn() + "</td>");

         out.println("</tr>");

      }

    } else {

      out.println("<p>No items found.</p>");

    }

  %>

        </tbody>

      </table>

</div>

</body>

</html>
///////////////////////////////
```

delete_batch.jsp
```
<%@ page language="java" contentType="text/html; charset=UTF-8"
   pageEncoding="UTF-8"  %>
```

```html
<!DOCTYPE html>
<html>
<head>
  <meta charset='utf-8'>
  <meta http-equiv='X-UA-Compatible' content='IE=edge'>
  <title>Delete a Batch</title>
  <meta name='viewport' content='width=device-width, initial-scale=1'>
  <link rel='stylesheet' type='text/css' media='screen' href='../style2.css'>
</head>
<body>
  <div class="container">
    <img src="../Funfit_img/Zumba_funfit.png" alt="Background Image" class="background-image" />
    <div class="card">
      <div class="form-container">
            <form action="../BatchUpdateController" method="POST">
            <div><top><h2><a href="../index.html">Home</a></h2></top></hr><h1 ><center>Delete
Batch</center></h1></div><br>
              <input type="hidden" name="_method" value="DELETE">
              <label>Batch No:</label></br><input type="text" name="batch_id" ></br>
              <input type="submit" id="submit" value="submit"></hr>
              <input type="reset" id="reset" value="reset"/></br>
            </form>
      </div>
    </div>
  </div>
</body>
</html>
```

/////////////////////////////

get_batch.jsp
```jsp
<%@ page language="java" contentType="text/html; charset=UTF-8"
  pageEncoding="UTF-8"  %>


<!DOCTYPE html>
<html>
```

```html
<head>

    <meta charset='utf-8'>

    <meta http-equiv='X-UA-Compatible' content='IE=edge'>

    <title>Selecting a batch</title>

    <meta name='viewport' content='width=device-width, initial-scale=1'>

    <link rel='stylesheet' type='text/css' media='screen' href='../style2.css'>

</head>

<body>

    <div class="container">

        <img src="../Funfit_img/Zumba_funfit.png" alt="Background Image" class="background-image" />

        <div class="card">

            <div class="form-container">

                    <form action="../BatchUpdateController" method="POST">

                    <div><top><h2><a href="../index.html">Home</a></h2></top></hr><h1><center>Choose a
Batch</center></h1></div><br>

                        <input type="hidden" name="_method" value="PUT">

                        <label>Batch No:</label></br><input type="text" name="batch_id" ></br>

                        <input type="submit" id="submit" value="submit"></hr>

                        <input type="reset" id="reset" value="reset"/></br>

                    </form>

            </div>

        </div>

    </div>

</body>

</html>
```

/////////////////////////////

update_batch.jsp
```jsp
<%@ page language="java" contentType="text/html; charset=UTF-8"

    pageEncoding="UTF-8" import="java.sql.*" %>

<%@ page import="com.funfit.bean.Batch" %>

<%

Batch b =(Batch)request.getAttribute("batch");

%>

<!DOCTYPE html>
```

```html
<html>
<head>
    <meta charset='utf-8'>
    <meta http-equiv='X-UA-Compatible' content='IE=edge'>
    <title>Updating the Batch details</title>
    <meta name='viewport' content='width=device-width, initial-scale=1'>
    <link rel='stylesheet' type='text/css' media='screen' href='style2.css'>


</head>
<body>
    <div class="container">
        <img src="Funfit_img/Zumba_funfit.png" alt="Background Image" class="background-image" />
        <div class="card">
            <div class="form-container">
                <form action="/FunFit/BatchUpdateController" method="POST">
                <div><top><h2><a href="FunFit/index.html">Home</a></h2></top></hr><h1><center
id="tit">Update Batch</center></h1></div><br>
                <input type="hidden" name="_method" value="POST">
                    <label>Batch No:</label></br><input type="text" id="batch_id" name="batch_id" value="<%=
b.getBatch_id() %>" readonly="readonly"></br>
                    <label for="batch-type">Batch Type:</label></br><input type="radio" id ="batch-type"
name="type" value="morning"><label>Morning(a.m)</label></hr>
                        <input type="radio" id="batch-type" name="type"
value="evening"><label>Evening(p.m)</label></br>
                        <label id="batch-select" for="batch-select">Batch Time:</label>
                    <select id = "batch-select" name="time">
                      <option value = "7">7</option>
                      <option value = "8">8</option>
                      <option value = "9">9</option>
                    </select><br>
                  <input type="submit" id="submit" value="submit"></hr>
                  <input type="reset" id="reset" value="reset"/></br>
                </form>
            </div>
        </div>
    </div>
```

```
<script>

document.getElementById('batch_id').innerHTML = <%= b.getBatch_id() %>;

<%-- document.getElementById('batch-select').innerHTML = <%= b.getBatch_time() %> ; --%>

</script>

</body>

</html>
```

////////////////////////////////

view_batch.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"

    pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

    <meta charset='utf-8'>

    <meta http-equiv='X-UA-Compatible' content='IE=edge'>

    <title>Delete a Batch</title>

    <meta name='viewport' content='width=device-width, initial-scale=1'>

    <link rel='stylesheet' type='text/css' media='screen' href='../style2.css'>

</head>

<body>

    <div class="container">

        <img src="../Funfit_img/Zumba_funfit.png" alt="Background Image" class="background-image" />

        <div class="card">

            <div class="form-container">

                    <form action="../BatchViewController" method="GET">

                    <div>

                            <h2><a href="../index.html">Home</a></h2>

                            <h1><center>Choose a Batch to delete</center></h1>

                    </div>

                    <div id ="view_container">

                        <label>Batch No:</label></br><input type="text" name="batch_id" placeholder="Enter the batch
id:"></br>

                        <input type="submit" id="submit" value="submit"></hr>
```

```html
                <input type="reset" id="reset" value="reset"/></br>
            </div>
        </form>
    </div>
  </div>
</div>
</body>
</html>
```

//////////////////////////////

add_participant.jsp

```jsp
<%@page import="com.funfit.dao.BatchDAO"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8" import="java.sql.*,com.funfit.bean.*,java.util.*" %>
<%!  int new_id ;Connection con=null; Statement stmt = null; ResultSet rs =null; List<Batch> listOfBatches;  %>
<%
try{
        Class.forName("com.mysql.cj.jdbc.Driver");
}catch(Exception e){
                out.println("An Exception occured at Driver Connection!" +e.getMessage());
                }
try{
        con  = DriverManager.getConnection("jdbc:mysql://localhost:3306/funfit", "root", "root");
}catch(SQLException e){
        out.println("An Exception occured at Database Connection!" +e.getMessage());
}

if(con!=null){
        try{
                String query1 = "select * from participant_id; ";
                stmt = con.createStatement();
                rs = stmt.executeQuery(query1);
                if(rs.next()){
```

```java
                new_id = rs.getInt(1) +1 ;

                System.out.println(new_id);

        }

        BatchDAO bs = new BatchDAO();

        listOfBatches = bs.getBatches();

}


finally{

        try{

                if (rs != null) {

        try {

           rs.close();

        } catch (SQLException e) {

           out.println("Failed to close ResultSet! " + e.getMessage());

        }

        }

        if (stmt != null) {

          try {

            stmt.close();

          } catch (SQLException e) {

            out.println("Failed to close Statement! " + e.getMessage());

          }

        }

        if (con != null) {

          try {

            con.close();

          } catch (SQLException e) {

            out.println("Failed to close Connection! " + e.getMessage());

          }

        }

          }catch(Exception e){

                out.println("An Exception occured at logic2" +e.getMessage());

          }

}
```

```
    }


 %>
<!DOCTYPE html>
<html>
<head>
  <meta charset='utf-8'>
  <meta http-equiv='X-UA-Compatible' content='IE=edge'>
  <title>Page Title</title>
  <meta name='viewport' content='width=device-width, initial-scale=1'>
  <link rel='stylesheet' type='text/css' media='screen' href='../style2.css'>
</head>
<body>
  <div class="container">
    <img src="../Funfit_img/Zumba_funfit.png" alt="Background Image" class="background-image" />
    <div class="card" >
      <div class="form-container" >
            <form  id="batch_form"  action="../ParticipantAddController"  method="POST">
            <div>
            <div><h2><a href="../index.html">Home</a></h2></hr><h1><center id="tit">Add
Participant</center></h1></div><br>
            <input type="hidden" name="_method" value="POST">



            <label>Participant Id:</label></br><input type="text" id="participant_id" name="participant_id"
value="<%= new_id %>" disabled></br>
            <label>Participant Name:</label></br><input type="text" id="participant_name"
name="participant_name" placeholder="Enter the Participant name"></br>
            <label>Participant Age:</label></br><input type="number" id="participant_age"
name="participant_age"></br>
            <label>Participant Email:</label></br><input type="email" id="participant_email"
name="participant_email" placeholder="email@example.com"></br>
            <label>Participant phone number :</label></br><input type="text" id="participant_phn"
name="participant_phn" placeholder="Enter the phone number"></br>
            <label id="batch_id_label" for="batch_id">Batch Id:</label>
            <select id = "batch-id" name="batch_id">
```

```
            <%
    if(listOfBatches != null) {

        for(Batch item : listOfBatches){

          out.println("<option value =");

            out.println("\""+item.getBatch_id()+"\">Batch: " +item.getBatch_id()+"     Time: "+item.getBatch_time()+"
Type: "+item.getBatch_type()+"</option>");

        }

    } else {

        out.println("<p>No batchess found.</p>");

    }

  %>

                </select><br>

                <input type="submit" id="submit" value="submit"></hr>

                <input type="reset" id="reset" value="reset"/></br>

                  </div>

                </form>

          </div>

        </div>

      </div>

</body>

</html>

//////////////////////////////////
```

delete_participant.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"  %>


<!DOCTYPE html>
<html>
<head>
  <meta charset='utf-8'>
  <meta http-equiv='X-UA-Compatible' content='IE=edge'>
  <title>Page Title</title>
  <meta name='viewport' content='width=device-width, initial-scale=1'>
```

```html
    <link rel='stylesheet' type='text/css' media='screen' href='../style2.css'>
</head>
<body>
   <div class="container">
      <img src="../Funfit_img/Zumba_funfit.png" alt="Background Image" class="background-image" />
      <div class="card">
         <div class="form-container">
               <form action="../ParticipantUpdateController" method="POST">
               <div><top><h2><a href="../index.html">Home</a></h2></top></hr><h1><center>Choose participant</center></h1></div><br>
                  <input type="hidden" name="_method" value="DELETE">
                   <label>Batch No:</label></br><input type="text" name="participant_id" placeholder="Enter the participant id:"></br>
                  <input type="submit" id="submit" value="submit"></hr>
                  <input type="reset" id="reset" value="reset"/></br>
               </form>
         </div>
      </div>
   </div>
</body>
</html>
```

get_participant.jsp

```jsp
<%@ page language="java" contentType="text/html; charset=UTF-8"
   pageEncoding="UTF-8"  %>

<!DOCTYPE html>
<html>
<head>
   <meta charset='utf-8'>
   <meta http-equiv='X-UA-Compatible' content='IE=edge'>
   <title>Page Title</title>
   <meta name='viewport' content='width=device-width, initial-scale=1'>
```

```html
        <link rel='stylesheet' type='text/css' media='screen' href='../style2.css'>
</head>
<body>
    <div class="container">
        <img src="../Funfit_img/Zumba_funfit.png" alt="Background Image" class="background-image" />
        <div class="card">
            <div class="form-container">
                    <form action="../ParticipantUpdateController" method="POST">
                    <div><top><h2><a href="../index.html">Home</a></h2></top></hr><h1><center>Choose a participant</center></h1></div><br>

                        <input type="hidden" name="_method" value="PUT">

                        <label>Batch No:</label></br><input type="text" name="participant_id" placeholder="Enter the participant id:"></br>

                        <input type="submit" id="submit" value="submit"></hr>

                        <input type="reset" id="reset" value="reset"/></br>

                    </form>

        </div>

      </div>

    </div>
</body>
</html>
/////////////////////////////////
```

participant_result.jsp

```jsp
<%@page import="org.apache.jasper.tagplugins.jstl.core.ForEach"%>
<%@page import="com.funfit.bean.Participant"%>
<%@page import="java.util.List"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8" import ="java.util.List,com.funfit.bean.*"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
```

```css
<style>
.styled-table{

        border-collapse: collapse;

        margin: 25px 0;

        font-size: 0.9em;

        min-width: 500px;

        border-radius: 5px 5px 0 0;

        overflow: hidden;

        box-shadow: 0 0 20px rgba(0,0,0,0.15);

}
.styled-table thead tr{

        background-color: #009879;

        color: #ffffff;

        text-align: left;

        font-weight: bold;

}


.styled-table th,td{

        padding: 12px 15px;

}


.styled-table tbody tr{

        border-bottom: 1px solid #dddddd;

}


.styled-table tbody tr nth:nth-of-type(even){

        background-color: #f3f3f3;

}


.styled-table tbody tr nth:last-of-type{

        border-bottom: 2px solid #009879;

}


.styled-table tbody tr.active-row{
```

```html
        font-weight: bold;

        color:

}

</style>

</head>

<body>

<div class="table-container">

        <table class="styled-table">

          <thead>

            <tr>

                <th>Participant id</th>

                <th>Batch id</th>

                <th>Name</th>

                <th>Age</th>

                <th>Email</th>

                <th>Phone Number</th>

            </tr>

          </thead>

          <tbody>

            <%
// Retrieve the list from the request
            Participant p = (Participant)request.getAttribute("participant");
// Check if the list is not null and display it
if (p != null) {
    out.println("<tr>");
    out.println("<td>" + p.getParticipant_id() + "</td>");
    out.println("<td>" + p.getBatch_id() + "</td>");
    out.println("<td>" + p.getParticipant_name() + "</td>");
    out.println("<td>" + p.getParticipant_age() + "</td>");
    out.println("<td>" + p.getParticipant_email() + "</td>");
    out.println("<td>" + p.getParticipant_phn() + "</td>");
    out.println("</tr>");
} else {
    out.println("<p>No items found.</p>");
```

```
        }
    %>
            </tbody>
        </table>
</div>
</body>
</html>
//////////////////////////////
```

update_participant.jsp

```
<%@page import="com.funfit.bean.Participant"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"  %>
<%@ page import="com.funfit.bean.*,java.util.*, com.funfit.dao.*" %>
<%
Participant p  =(Participant)request.getAttribute("participant");
%>
<%!List<Batch> listOfBatches;  %>
<%
try{
            BatchDAO bs = new BatchDAO();
            listOfBatches = bs.getBatches();
    }
    catch(Exception e){
                out.println("An Exception occured at logic2" +e.getMessage());
        }
%>


<!DOCTYPE html>
<html>
<head>
  <meta charset='utf-8'>
  <meta http-equiv='X-UA-Compatible' content='IE=edge'>
```

```html
    <title>Page Title</title>
    <meta name='viewport' content='width=device-width, initial-scale=1'>
    <link rel='stylesheet' type='text/css' media='screen' href='style2.css'>


</head>
<body>
    <div class="container">
        <img src="Funfit_img/Zumba_funfit.png" alt="Background Image" class="background-image" />
        <div class="card">
            <div class="form-container">
                    <form id="batch_form" action="/FunFit/ParticipantUpdateController" method="POST">
                    <div>
                    <div><top><h2><a href="index.html">Home</a></h2></top></hr><h1><center id="tit">Update
Participant</center></h1></div><br>
                    <input type="hidden" name="_method" value="POST">
                       <%-- <label>Batch No:</label></br><input type="text" id="batch_id" name="batch_id" value="<%=
b.getBatch_id() %>" readonly="readonly"></br> --%>
                        <label>Participant Id:</label></br><input type="text" id="participant_id" name="participant_id"
value="<%= p.getParticipant_id() %>" readonly="readonly"></br>
                    <label>Participant Name:</label></br><input type="text" id="participant_name"
name="participant_name" placeholder="Enter the Participant name"></br>
                    <label>Participant Age:</label></br><input type="number" id="participant_age"
name="participant_age"></br>
                    <label>Participant Email:</label></br><input type="email" id="participant_email"
name="participant_email" placeholder="email@example.com"></br>
                    <label>Participant phone number :</label></br><input type="text" id="participant_phn"
name="participant_phn" placeholder="Enter the phone number"></br>
                    <label id="batch_id_label" for="batch_id">Batch Id:</label>
                    <select id = "batch-id" name="batch_id">
                        <%
    if(listOfBatches != null) {
      for(Batch item : listOfBatches){
        out.println("<option value =");
         out.println("\""+item.getBatch_id()+"\">Batch: " +item.getBatch_id()+"    Time: "+item.getBatch_time()+"
Type: "+item.getBatch_type()+"</option>");
      }
    } else {
```

```
            out.println("<p>No batchess found.</p>");
        }
    %>

                </select><br>

                <input type="submit" id="submit" value="submit"></hr>

                <input type="reset" id="reset" value="reset"/></br>

                  </div>

                </form>

        </div>

      </div>

    </div>

    <script>


    document.getElementById('participant_id').innerHTML = <%= p.getParticipant_id() %>;

    <%-- document.getElementById('batch-select').innerHTML = <%= b.getBatch_time() %> ; --%>

</script>

</body>

</html>

///////////////////////////
```

view_participant.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"  %>


<!DOCTYPE html>

<html>

<head>

    <meta charset='utf-8'>

    <meta http-equiv='X-UA-Compatible' content='IE=edge'>

    <title>Delete a Participant</title>

    <meta name='viewport' content='width=device-width, initial-scale=1'>

    <link rel='stylesheet' type='text/css' media='screen' href='../style2.css'>

</head>
```

```html
<body>
  <div class="container">
    <img src="../Funfit_img/Zumba_funfit.png" alt="Background Image" class="background-image" />
    <div class="card">
      <div class="form-container">
        <form action="../ParticipantViewController" method="GET">
          <div>
            <h2><a href="../index.html">Home</a></h2>
            <h1><center>Choose a Participant to view Details</center></h1>
          </div>
          <input type="hidden" name="_method" value="PUT">
          <label>Batch No:</label></br><input type="text" name="participant_id" placeholder="Enter the participant id:"></br>
          <input type="submit" id="submit" value="submit"></hr>
          <input type="reset" id="reset" value="reset"/></br>
        </form>
      </div>
    </div>
  </div>
</body>
</html>
```

/////////////////////////////

index.html

```html
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<title>Funfit</title>
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css" rel="stylesheet">
<link rel="stylesheet" type="text/css" href="style.css">
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js"></script>
</head>
<body>
```

```html
<div class="m-4">

	<nav class="navbar navbar-expand-sm navbar-light bg-light">

		<div class="container-fluid">

			<img id="logo" src="Funfit_img/Logo funfit.png"/>

			<a href="#" class="navbar-brand">Funfit</a>

			<div id="navbarCollapse" class="collapse navbar-collapse">

				<ul class="nav navbar-nav ms-auto">

					<li class="nav-item dropdown" >


						<a class="nav-link dropdown-toggle" href="#"
id="navbarDropdown" role="button" data-toggle="dropdown">

							Home

						</a>

					</li>

					<li class="nav-item dropdown" >

						<a href="#" class="nav-link dropdown-toggle" data-bs-toggle
="dropdown">Batches</a>

							<div class="dropdown-menu dropdown-menu-end">

								<a href="Batch/add_batch.jsp" class="dropdown-
item">Add Batch</a>

								<a href="Batch/get_batch.jsp"  class="dropdown-
item">Update Batch</a>

								<a href="Batch/delete_batch.jsp"
class="dropdown-item">Delete Batch</a>

								<a href="Batch/view_batch.jsp" id="view_batch"
class="dropdown-item">View Batch</a>

							</div>

					</li>

					<li class="nav-item dropdown" >

						<a href="#" class="nav-link dropdown-toggle" data-bs-
toggle="dropdown">Participants</a>

							<div class="dropdown-menu dropdown-menu-end">

								<a href="Participant/add_participant.jsp"
class="dropdown-item">Add Participant</a>

								<a href="Participant/get_participant.jsp"
class="dropdown-item">Update Participant</a>

								<a href="Participant/delete_participant.jsp"
class="dropdown-item">Delete Participant</a>
```

```html
                                                    <a href="Participant/view_participant.jsp"
class="dropdown-item">View Participant</a>


                                                    </div>
                                        </li>
                                </ul>
                        </div>
                </div>
        </nav>
        <div class="mask1"><img src="Funfit_img/Zumba_funfit.png" alt="Zumba"/></div>
</div>
</body>
</html>
```

//

style.css

```css
@charset "UTF-8";
.m-4{
    margin: 0rem !important;
}
#logo{
    height: 70px;
    border-radius: 75%;
    width: auto;
}
.navbar-brand{
    font-size: xx-large;
    font-family: cursive;
    font-style: oblique;
    font-weight: bold;
}
.nav #navbarDropdown{

    color: #f73434de !important;
```

```css
    font-family: Verdana, sans-serif !important;

    font-weight: bolder;

}
.nav #navbarDropdown::after {

    display: none; /* Hides the default caret */

    color: #f73434de !important;

}
.nav .nav-link{

    color: #aaa9b1!important;

    font-family: Verdana, sans-serif  !important;

    font-weight: bolder;

}
.nav .nav-link:hover{

    color: #f73434de !important;

}
.dropdown-menu .dropdown-item:hover{

    color: #f73434de !important;

    font-weight: bolder;

}
img{

    width: 100vw;

    height: 100dvh;

}
////////////////////////////////////////////////////////////
```

style2.css

```css
img{

    width: 100vw;

    height: fit-content;

}
.card {

    position: fixed;

    bottom: 0;
```

```css
    left: 0;
    width: 100%;
    height: 100%;
    background-color: rgb(100, 104, 100);
    opacity: 0.8;
  }

.form-container{
  background-color: #ffdfcc;
  width:75%;
  height:70%;
  margin: 10%;
  padding-left: 10v;
  padding-right: 10v;
  padding-top: 10v;
  padding-bottom: 10v;
  display: inline-flex;
  align-items: center;
  justify-content: center;
  flex-direction:column;
  row-gap: 20px;
  overflow-y: auto;
}

#batch_form{
        height: inherit;
}

label{
        border:3px;
        padding-top: 20px;
        padding-bottom: 20px;
        font-size: large;
        font-family: Verdana, sans-serif;
```

```css
}

input[type="text"],input[type="email"]{
        color: #370e45 !important;

        padding-top: 20px;

        padding-bottom: 20px;

        display: block;              /* Block display for consistent alignment */

    width: 100%;                 /* Full width or adjust as needed */

    max-width: 400px;            /* Optional max width for larger screens */

    margin: 0;               /* Removes any default margin */

    border: none;               /* Removes all borders */

    border-bottom: 2px solid black; /* Adds a line at the bottom */

    outline: none;              /* Removes the default focus outline */

    background: none;              /* Removes background color */

    box-sizing: border-box;        /* Includes padding in width calculation */

    transition: border-color 0.3;

    font-size: large;

    font-style: italic;

}


input[type="radio"] {
  padding: 5px ;

  appearance: none;        /* Removes the default radio appearance */

  width: 20px;            /* Set width */

  height: 20px;           /* Set height */

  border: 2px solid #370e45;    /* Default border */

  border-radius: 50%;       /* Makes it circular */

  outline: none;           /* Removes the default focus outline */

  cursor: pointer;          /* Changes cursor to pointer */

  position: relative;       /* Allows positioning of custom elements */

}


input[type="radio"]:checked::before {
  content: '';
```

```css
    display: block;

    width: 10px;            /* Size of the inner dot */

    height: 10px;           /* Size of the inner dot */

    background-color: #370e45;    /* Color of the inner dot */

    border-radius: 50%;        /* Makes the dot circular */

    position: absolute;        /* Position it relative to the radio */

    top: 4px;              /* Adjust position as needed */

    left: 4px;             /* Adjust position as needed */
}


#batch-select, #batch-id{

        padding-top: 20px;

        padding-bottom:20px;
}


input[type="submit"],input[type="reset"]{

  padding: 20px;

  border: 0;

  border-radius: 100px;

  box-sizing: border-box;

  color: #ffffff;

  cursor: pointer;

  font-family: -apple-system, system-ui, system-ui, "Segoe UI", Roboto, "Helvetica Neue", "Fira Sans", Ubuntu,
Oxygen, "Oxygen Sans", Cantarell, "Droid Sans", "Apple Color Emoji", "Segoe UI Emoji", "Segoe UI Symbol", "Lucida
Grande", Helvetica, Arial, sans-serif;

  font-size: 16px;

  font-weight: 600;

  justify-content: center;

  line-height: 20px;

  max-width: 480px;

  min-height: 40px;

  min-width: 0px;

  overflow: hidden;

  padding: 0px;

  padding-left: 20px;
```

```css
  padding-right: 20px;

  text-align: center;

  touch-action: manipulation;

  transition: background-color 0.167s cubic-bezier(0.4, 0, 0.2, 1) 0s, box-shadow 0.167s cubic-bezier(0.4, 0, 0.2, 1) 0s,
color 0.167s cubic-bezier(0.4, 0, 0.2, 1) 0s;

  user-select: none;

  -webkit-user-select: none;

}

#submit{

        background-color: #0A66C2;

        margin: 30px;

}

#submit:focus{

        background-color: #16437E;

}

#submit:active{

        background: #09223b;

        color: rgb(255, 255, 255, .7);

}

#reset{

        background-color: #13aa52;

        margin: 30px;

}

#reset:focus{

        background-color: #16437E;

}

#reset:active{

        background: #09223b;

        color: rgb(255, 255, 255, .7);

}
```

///