**REVIEW del progetto di Stefano Postal SE2\_2016\_174735\_P**

**In index.js:**

function ordineGiorno(giorno) { # i nomi delle funzioni non iniziano con un verbo come secondo la convenzione

var trovato = false;

var i;

for (i = 0; i < ordini.length && !trovato; i++) {

if (ordini[i].giorno.localeCompare(giorno) == 0) {

trovato = true;

}

}

return (trovato ? ordini[i-1] : "");

}

**In jsOrdina.js:**

var ordine = []; # indentazione scorretta

function add(name, i){

ordine.push({nome:name, id:i});

stampaOrdine();

var p1 = '<button type="submit" class="btn btn-danger plus" onclick="remove(' #1

var p2 = ')"><img src="img/minus.png" width="16" height="16" alt="-"></button>' #1

document.getElementById("dplus"+i).innerHTML = p1+"'"+name+"',"+i+p2;

}

function stampaOrdine(){

var s =""; #2

var sv = ""; #2

for(var i=0; i<ordine.length; i++){

sv += "<div><b>"+ordine[i].nome +"</b></div>"

}

document.getElementById("viewOrd").innerHTML = sv;

for(var i=0; i<ordine.length-1; i++){

s += ordine[i].nome+",";

}

if(ordine.length>0) s+= ordine[ordine.length-1].nome;

document.getElementById("ord").value = s;

}

#1 vengono dichiarate delle variabili usandole come costanti

#2 nomi poco intuitivi

**In ordinazione.tpl:**

<div class="row" id="menu">

<div class="col-xs-8" id="viewOrd">

</div>

<div class="col-xs-4 bottoni" >

<form action="/effettuaOrdine" method="post">

<input type="hidden" id="ord" name="ord">

<input type="hidden" name ="giorno" value="(:giorno:)">

<button type="submit" class="btn btn-success" id="b">ORDINA</button> #

</form>

<form action="/" method="post">

<button type="submit" class="btn btn-danger" id="b" onclick="Submit()">CANCELLA</button> #

</form>

</div>

</div>

# trovati due elementi con lo stesso id

# il nome di una funzione inizia con una lettera quindi non secondo convenzione

# indentazione un po’ approssimativa

>> Conclusioni:

-MVC è stato rispettato

-Commenti abbastanza presenti e codice abbastanza intuitivo

-Naming al 80%,