

RELAZIONE TPSIT

Creazione home.ejs

Come prima cosa creare all'interno della cartella *views* il file *home.ejs*, che conterrà il form in cui andremo a inserire i valori delle diverse stazioni.

```
<h3>Inserisci</h3>
  <form action="/salva" method="post">
    città: <select name="citta">
      <%for(var i=0;i<arrayCitta.length;i++) {%>
        <option value="<%=arrayCitta[i]%>"><%=arrayCitta[i]%></option>
      <% } %></select><br>
    Valore Massimo: <input type="text" name="massima"><br>
    Valore Minimo: <input type="text" name="minima"><br>
    <input type="submit" value="salva">
  </form>
  <input type="submit" value="riepilogo" href="/riepilogo">
```

Abbiamo realizzato la home del sito in *ejs* perché così potevamo generare dinamicamente partendo da un array i nomi delle stazioni.

Creazione delle classi

Creiamo le classi *Stazione*, al cui interno ci sarà anche la funzione *zona*, che ci permetterà di assegnare la zona (*NORD,CENTRO,SUD*) basandoci sulla variabile *citta*, e *Stazioni*, che utilizzeremo per creare un array di *Stazione*, e al cui interno ci sarà la funzione *media* che ci permetterà di calcolare la media della temperatura minima.

Classe Stazione

```
class Stazione{

  constructor(tempMax,tempMin,citta){
    this.tempMax=tempMax;
    this.tempMin=tempMin;
    this.citta=citta;
    this.data= Date.now().toString;
    this.zona = zona(citta);
  }

  get getCitta(){
    return this.citta;
  }

  get getMin(){
    return this.tempMin;
  }
}
```

```

    get getMax(){
        return this.tempMax;
    }

    get getZona(){
        return this.zona;
    }
}
module.exports={Stazione:Stazione}

function zona(citta){
    switch (citta){
        case "Verzuolo" :
            this.zona="NORD";
            break;
        case "Milano" :
            this.zona="NORD";
            break;
        case "Torino" :
            this.zona="NORD";
            break;
        case "Firenze" :
            this.zona="CENTRO";
            break;
        case "Bologna" :
            this.zona="CENTRO";
            break;
        case "Roma" :
            this.zona="CENTRO";
            break;
        case "Napoli" :
            this.zona="SUD";
            break;
        case "Bari" :
            this.zona="SUD";
            break;
        case "Messina" :
            this.zona="SUD";
            break;
    }
}

```

Classe Stazioni

```

var Stazione=require("../Stazione.js").Stazione;

class Stazioni{

    constructor(){
        this.listaStazioni=new Array();
    }

    add(tempMax,tempMin,citta){
        this.listaStazioni.push(new Stazione(tempMax,tempMin,citta));
    }
}

```

```

    }

    media(mediaV,mediaM,mediaT,mediaF,mediaB,mediaR,mediaN,mediaB2,mediaM2){
        for(var i=0;i<this.listaStazioni.length;i++){
            switch (this.listaStazioni[i].getcitta){
                case "Verzuolo":
                    mediaV=mediaV+this.listaStazioni[i].getMin;
                    break;
                case "Milano":
                    mediaM=mediaM+this.listaStazioni[i].getMin;
                    break;
                case "Torino":
                    mediaT=mediaT+this.listaStazioni[i].getMin;
                    break;
                case "Firenze":
                    mediaF=mediaF+this.listaStazioni[i].getMin;
                    break;
                case "Bologna":
                    mediaB=mediaB+this.listaStazioni[i].getMin;
                    break;
                case "Roma":
                    mediaR=mediaR+this.listaStazioni[i].getMin;
                    break;
                case "Napoli":
                    mediaN=mediaN+this.listaStazioni[i].getMin;
                    break;
                case "Bari":
                    mediaB2=mediaB+this.listaStazioni[i].getMin;
                    break;
                case "Messina":
                    mediaM=mediaM+this.listaStazioni[i].getMin;
                    break;
            }
        }
    }
}

module.exports={Stazioni:Stazioni}

```

Realizziamo l'index.js

Realizziamo il file *index.js* con cui da cui gestiremo le diverse funzioni. Per gestire le funzioni utilizziamo il framework *express*

```

var express =require("express");
var app=express();
var bodyParser=require("body-parser");
app.set('view engine', 'ejs');
var Stazioni=require("./Stazioni").Stazioni
app.use(bodyParser.json());
app.use(bodyParser.urlencoded({extended:true}));
var mieStazioni=new Stazioni();

let mediaV=0;
let mediaM=0;

```

```

let mediaT=0;
let mediaF=0;
let mediaB=0;
let mediaR=0;
let mediaN=0;
let mediaB2=0;
let mediaM2=0;

app.listen(1345, function(){

app.get("/", function(req, res){
    var arrayCitta= new
    Array("Verzuolo", "Milano", "Torino", "Firenze", "Bologna", "Roma", "Napoli", "B
    ari", "Messina");
    res.render("home.ejs", {arrayCitta:arrayCitta});
});

});

```

Realizziamo il file salva.html e aggiungiamo la funzione salva

Aggiungiamo al file *index.js* la funzione *salva*, che ci permetterà di salvare in un array tutti i valori che l'utente inserirà nel form

```

app.post("/salva", function(req, res){
    var max=req.body.massima;
    var min=req.body.minima;
    var citta=req.body.citta;
    mieStazioni.add(max, min, citta);
mieStazioni.media(mediaV, mediaM, mediaT, mediaF, mediaB, mediaR, mediaN, mediaB2, media
M2);
    res.sendFile(__dirname+"/sito/salva.html");
});

```

Successivamente creiamo la cartella *sito* ed al suo interno realizziamo il file *salva.html*, che darà la “veste grafica” alla funzione.

```

<html>
  <head></head>
  <body>
    <h3>Inserimento Effettuato</h3>
    <p>Inserimnto dati effettuato</p>
  </body>
</html>

```

Realizziamo il file riepilogo.ejs e aggiungiamo la funzione riepilogo

Aggiungiamo al file *index.js* la funzione *riepilogo*, successivamente, dentro la cartella *ejs*, aggiungiamo il file *riepilogo.ejs* a cui passare l'array contenente il nome delle città, la variabile *Stazioni*, e l'array contenente le medie delle temperature minime

index.js

```
app.post("/riepilogo", function(req, res){
    var arrayMedia = new
    Array(mediaV, mediaM, mediaT, mediaF, mediaB, mediaR, mediaN, mediaM2);
    res.render("riepilogo.ejs" {arrayMedia:arrayMedia, arrayCitta:arrayCitta, sta
    zioni:Stazioni});
});
```

riepilogo.ejs

```
<%num = 0%>
<%for(var i=0;i<arrayCitta.length;i++) {%>
    <b><%arrayCitta[i]%></b>:<br>
    <%var massima = 0%>
    <%for(var j = 0; j < stazioni.listaStazioni.length; i++){%>
        <%if(stazioni.listaStazioni[j].getMax() > massima){%>
            <%massima=stazioni.listaStazioni[j].getMax()%>
        <%}%>
        <%for(var j = 0; j < stazioni.listaStazioni.length; i++){%>
            <%if(stazioni.listaStazioni[j].getCitta() > arrayCitta[i]){%>
                <%num = num + 1 %>
            <%}%>
        <%}%>
        <p>massima: <%massima%> </p>
        <p>minima: <%num%> </p>
    <% } %>
    <br>
    <%var massimaNord = 0%>
    <%var minimaNord = 0%>
    <%var massimaCentro = 0%>
    <%var minimaCentro = 0%>
    <%var massimaSud = 0%>
    <%var minimaSud = 0%>
    <%for(var i=0;i<stazioni.listaStazioni.length;i++) {%>
        <%if(stazioni.listaStazioni[i].getZona() == NORD){%>
            <%massimaNord = massimaNord + stazioni.listaStazioni[i].getMax()%>
            <%minimaNord = minimaNord + stazioni.listaStazioni[i].getMin()%>
        <%}if(stazioni.listaStazioni[i].getZona() == CENTRO){%>
            <%massimaCentro = massimaCentro + stazioni.listaStazioni[i].getMax()%>
            <%minimaCentro = minimaCentro + stazioni.listaStazioni[i].getMin()%>
        <%}else{%>
            <%massimaSud = massimaSud + stazioni.listaStazioni[i].getMax()%>
            <%minimaSud = minimaSud + stazioni.listaStazioni[i].getMin()%>
        <%}%>
    <% } %>
    <b>NORD</b>:<br>
    <p>massima: <%massimaNord%></p>
    <p>minima: <%minimaNord%></p><br>
    <b>CENTRO</b>:<br>
    <p>massima: <%massimaCentro%></p>
    <p>minima: <%minimaCentro%></p><br>
    <b>SUD</b>:<br>
    <p>massima: <%massimaSud%></p>
    <p>minima: <%minimaSud%></p><br>
```