

Obligatorio 1 – Información de estudiantes

Grupo: M1A- AN

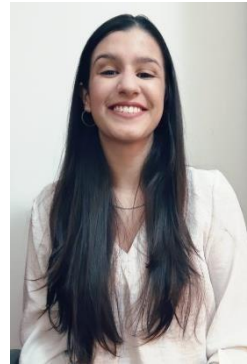
Felipe Rodríguez

Nro. De estudiante: 258250



Valentina Techera

Nro. De estudiante: 265947



Código:

HTML:

```
<!-- Felipe Rodriguez y Valentina Techera -->
```

```
<!doctype html>
```

```
<html lang="es">
```

```
  <head>
```

```
    <meta charset="UTF-8">
```

```
    <meta name="viewport" content="width=device-width, initial-  
scale=1.0">
```

```
    <title> Gestión de series </title>
```

```
    <link rel="stylesheet" href="CSS/Estilos.css">
```

```
    <script src="JavaScript/Clases.js"> </script>
```

```
    <script src="JavaScript/JS.js"> </script>
```

```
  </head>
```

```
  <body>
```

```

<header>
    <h1> Gestión de series </h1>
</header>
<nav>
    <ul id="lista">
        <li class="Series" id="idSeries"> <a
href="#ManejoDeSeries"> Series </a> </li>
        <li class="Opiniones" id="idOpiniones"> <a
href="#DatosOpiniones"> Opiniones </a> </li>
        <li class="Estadisticas" id="idEstadisticas"> <a
href="#InfoEstadisticas"> Estadísticas </a> </li>
    </ul>
</nav>
<section class="Series1">
    <a id="ManejoDeSeries"> </a>
    <h2> Manejo de series </h2>
    <form id="idFormSeries">
        <label for="idNombre"> Nombre </label>
        <br>
        <input type="text" id="idNombre" name="nombre"
maxlength="30" placeholder="Nombre" required>
        <br>
        <label for="idDescripcion"> Descripción </label>
        <br>
        <input type="text" id="idDescripcion"
name="descripcion" maxlength="30" placeholder="Descripción" required>
        <br>
        <label for="idTemporadas"> Cantidad de temporadas
</label>
        <br>
        <input type="number" id="idTemporadas"
name="temporadas" min="1" max="20" required>
        <br>
        <label for="idCapitulos"> Cantidad de capítulos
</label>
        <br>
        <input type="number" id="idCapitulos"
name="capitulos" min="1" max="30" required>

```

```

        </form>

        <br> <br>

        <button type="button" id="idAgregarSeries">
Agregar/Actualizar </button>

        <button type="button" id="idResetSeries"> Reset
</button>

        <button type="button" id="idPrevio"> Previo </button>

        <button type="button" id="idSiguiente"> Siguiente
</button>

        <br>

        <br>

        <a id="idImdb" href="https://www.imdb.com/"
target=blank>  </a>

    </section>

    <section class="Opiniones1">
        <a id="DatosOpiniones"> </a>
        <h2> Datos sobre opiniones de visualización </h2>
        <form id="FormOpiniones">
            <label for="idSerie"> Serie </label>
            <br>
            <select name="Serie" id="idSerie">

            </select>
            <br>
            <label for="idNumTemporada"> Temporada </label>
            <br>
            <input type="number" id="idNumTemporada"
name="NumTemporada" min="1" max="20" required>
            <br>
            <label for="idNumCapitulo"> Capítulo </label>
            <br>
            <input type="number" id="idNumCapitulo"
name="NumCapitulo" min="1" max="30" required>
            <br>
            <label for="idPuntaje"> Puntaje (1-10) </label>
            <br>
            <input type="range" id="idPuntaje" name="Puntaje">

```

```

        <br>
        <label for="idComentario"> Comentario </label>
        <br>
        <textarea id="idComentario" name="Comentario"
cols="50" rows="5" required></textarea>
    </form>
    <br>
    <br>
    <button id="idAgregarOpiniones" type="button">
Agregar/Actualizar </button>
    <button type="button" id="idResetOpiniones"> Reset
</button>

    <h2> Opiniones ingresadas de la serie </h2>
    <ul id="idLista">

    </ul>
</section>
<section class="Estadisticas1">
    <a id="InfoEstadisticas"> </a>
    <h2> Información estadística </h2>
    <p> Series sin opiniones </p>
    <ul id="idSinOpinion">

    </ul>
    <p> Top-3 Series con la máxima cantidad de
opiniones </p>
    <ul id="idTop3">

    </ul>
    <p> Información general </p>
    <label for="idOrden"> Orden: </label>
    <br>
    <input type="radio" id="idOrden" name="Orden"
value="Nombre de serie" checked>
    <label for="idOrden"> Nombre de serie </label>
    <input type="radio" id="idOrden1" name="Orden"
value="Cantidad de opiniones decreciente">

```

```

decreciente </label>
<label for="idOrden1"> Cantidad de opiniones
<br>
<br>
<table id="idTabla">

</table>
<br>
</section>
<footer>
<p> Felipe Rodríguez 258250 Valentina Techera 265947
</p>
</footer>
</body>
</html>

```

JavaScript:

JS.JS:

```
// Felipe Rodriguez y Valentina Techera
```

```
// Felipe Rodriguez y Valentina Techera
```

```
window.addEventListener("load", inicio);
```

```
let series = new Series();
```

```
let opiniones = new Opiniones();
```

```
let sistema = new Sistema();
```

```
function inicio (){
```

```
    document.getElementById("idAgregarSeries").addEventListener("click",
    agregar);
```

```
    document.getElementById("idSerie").addEventListener("change", lista);
```

```
    document.getElementById("idResetSeries").addEventListener("click",
    resetearSeries);
```

```
    document.getElementById("idResetOpiniones").addEventListener("click",
    resetearOpiniones);
```

```
    document.getElementById("idAgregarOpiniones").addEventListener("click",
    agregarOpinion);
```

```
    document.getElementById("idImdb").addEventListener("click", imdb);
```

```

        document.getElementById("idSeries").addEventListener("click", verSeries);
        document.getElementById("idOpiniones").addEventListener("click",
verOpiniones);
        document.getElementById("idEstadisticas").addEventListener("click",
verEstadisticas);
        document.getElementById("idPrevio").addEventListener("click", previo);
        document.getElementById("idSiguiente").addEventListener("click",
siguiente);
        document.getElementById("idOrden").addEventListener("click",
agregarTabla);
        document.getElementById("idOrden1").addEventListener("click",
agregarTabla);
        mostrar("Series1");
        ocultar("Opiniones1");
        ocultar("Estadisticas1");
        agregarTabla();
    }

```

```

function lista (){
    let serie = document.getElementById("idSerie").value;
    agregarALista(serie);
}

```

```

function previo (){
    let nombre = document.getElementById("idNombre");
    let descripcion = document.getElementById("idDescripcion");
    let temporadas = document.getElementById("idTemporadas");
    let capitulos = document.getElementById("idCapitulos");
    nombre.value=series.previo(nombre.value);
    descripcion.value=series.descripcion(nombre.value);
    temporadas.value=series.temporadas(nombre.value);
    capitulos.value=series.capitulos(nombre.value);
}

```

```

function siguiente(){
    let nombre = document.getElementById("idNombre");
    let descripcion = document.getElementById("idDescripcion");

```

```
    let temporadas = document.getElementById("idTemporadas");  
    let capitulos = document.getElementById("idCapitulos");  
    nombre.value=series.siguiente(nombre.value);  
    descripcion.value=series.descripcion(nombre.value);  
    temporadas.value=series.temporadas(nombre.value);  
    capitulos.value=series.capitulos(nombre.value);  
}
```

```
function resetearSeries(){  
    document.getElementById("idFormSeries").reset();  
}
```

```
function resetearOpiniones(){  
    document.getElementById("FormOpiniones").reset();  
}
```

```
function verSeries(){  
    mostrar("Series1");  
    ocultar("Opiniones1");  
    ocultar("Estadisticas1");  
}
```

```
function verOpiniones(){  
    ocultar("Series1");  
    mostrar("Opiniones1");  
    ocultar("Estadisticas1");  
}
```

```
function verEstadisticas(){  
    ocultar("Series1");  
    ocultar("Opiniones1");  
    mostrar("Estadisticas1");  
}
```

```

function agregar(){
    if (document.getElementById("idFormSeries").reportValidity()){
        let nombre = document.getElementById("idNombre").value;
        let descripcion = document.getElementById("idDescripcion").value;
        let temporadas = document.getElementById("idTemporadas").value;
        let capitulos = document.getElementById("idCapitulos").value;
        let cantOpiniones=0;
        let sumaPuntaje=0;
        let promedio=0;
        if (series.repetida(nombre)==false){
            series.agregar(new características(nombre, descripcion,
temporadas, capitulos));
            sistema.agregar(new características1(nombre,descripcion,
temporadas, capitulos, cantOpiniones, sumaPuntaje, promedio))
            agregarSelect();
            sinOpinion();
        } else {
            series.modificarDescripcion(nombre, descripcion);
            series.modificarTempradas(nombre, temporadas);
            series.modificarCapitulos(nombre, capitulos);
            sistema.modificarDescripcion(nombre, descripcion);
            sistema.modificarTempradas(nombre, temporadas);
            sistema.modificarCapitulos(nombre, capitulos);
            sinOpinion();
            top3();
        }
    }
    agregarTabla();
}

```

```

function agregarTabla(){
    let tabla=document.getElementById("idTabla");
    let ordenNombre=document.getElementById("idOrden");
    tabla.innerHTML="";
    let caption= tabla.createCaption();
    caption.innerHTML="Información detallada"

```



```

let cabezal = tabla.createTHead();
let filaTit= cabezal.insertRow();
let celTit1 = filaTit.insertCell(0);
celTit1.innerHTML ="Serie";
let celTit2 = filaTit.insertCell(1);
celTit2.innerHTML = "Cantidad de opiniones";
let celTit3 = filaTit.insertCell(2);
celTit3.innerHTML = "Promedio General";
if (ordenNombre.checked){
    let datos=sistema.darOrdenLetra();
    for (let elem of datos){
        let fila = tabla.insertRow();
        let celda1 = fila.insertCell(0);
        celda1.innerHTML= elem.nombre;
        let celda2 = fila.insertCell(1);
        celda2.innerHTML= elem.cantOpiniones;
        let celda3 = fila.insertCell(2);
        celda3.innerHTML= parseInt(elem.promedio);
    }
} else {
    let datos1=sistema.darOrdenNumeros();
    for (let elem of datos1){
        let fila = tabla.insertRow();
        let celda1 = fila.insertCell(0);
        celda1.innerHTML= elem.nombre;
        let celda2 = fila.insertCell(1);
        celda2.innerHTML= elem.cantOpiniones;
        let celda3 = fila.insertCell(2);
        celda3.innerHTML= parseInt(elem.promedio);
    }
}
}

```

```

function imdb (){
    let serie=document.getElementById("idNombre").value;

```

```

    if (serie==""){
        document.getElementById("idImdb").href="https://www.imdb.com/";
    } else {

        document.getElementById("idImdb").href="https://www.imdb.com/find?q="
+ serie + "&ref_=nv_sr_sm";
    }
}

```

```

function agregarSelect(){
    let combo = document.getElementById("idSerie");
    combo.innerHTML = "";
    let datos = series.darTodos();
    for (let elemento of datos){
        let nodoC = document.createElement("option");
        let nodoTextoC = document.createTextNode(elemento);
        nodoC.appendChild(nodoTextoC);
        combo.appendChild(nodoC);
    }
}

```

```

function ocultar(clase){
    var x = document.getElementsByClassName(clase);
    var i;
    for (i = 0; i < x.length; i++) {
        x[i].style.display = "none";
    }
}

```

```

function mostrar(clase){
    var x = document.getElementsByClassName(clase);
    var i;
    for (i = 0; i < x.length; i++) {
        x[i].style.display = "block";
    }
}

```

```

function agregarOpinion() {
    if (document.getElementById("FormOpiniones").reportValidity()){
        let nombre= document.getElementById("idSerie").value;
        let comentario =
document.getElementById("idComentario").value;
        let temporada =
document.getElementById("idNumTemporada").value;
        let capitulo =
document.getElementById("idNumCapitulo").value;
        let puntaje =
parseInt(document.getElementById("idPuntaje").value/10);
        if (repetido(nombre, temporada, capitulo)==false){
            opiniones.agregar(new Opinion(nombre, comentario,
temporada, capitulo, puntaje));
            agregarALista(nombre);
            sistema.cantidadOpiniones(nombre);
            sistema.puntajePromedio(nombre, puntaje);
            agregarTabla();
            top3();
            sinOpinion();
        } else {
            duplicada(nombre, temporada, capitulo, comentario,
puntaje);
            sistema.puntajePromedio(nombre, puntaje);
            agregarALista(nombre);
            agregarTabla();
        }
    }
}

```

```

function repetido(nombre, temporada, capitulo){
    let respuesta=false;
    if (opiniones.repetida(nombre)==true){
        if (opiniones.repetidaTemporada(temporada)==true){
            if (opiniones.repetidoCapitulo(capitulo)==true){
                respuesta=true;
            }
        }
    }
}

```

```

        }
    }
}
return respuesta;
}

```

```

function duplicada(nombre, temporada, capitulo, comentario, puntaje){
    eliminar(nombre, temporada, capitulo);
    opiniones.modificarComentario(nombre, temporada, capitulo, comentario);
    opiniones.modificarPuntaje(nombre, temporada, capitulo, puntaje);
}

```

```

function eliminar(nombre, temporada, capitulo){
    let puntaje = opiniones.buscarPuntaje(nombre, temporada, capitulo);
    sistema.eleiminarPuntaje(nombre, puntaje);
}

```

```

function agregarALista(nombre) {
    let lista = document.getElementById("idLista");
    lista.innerHTML = "";
    let datos = opiniones.darTodos();
    for (let elem of datos) {
        if (elem.nombre===nombre){
            let nodo = document.createElement("li");
            let nodoTexto = document.createTextNode(elem);
            nodo.appendChild(nodoTexto);
            lista.appendChild(nodo);
        }
    }
}

```

```

function top3 (){
    let lista = document.getElementById("idTop3");
    lista.innerHTML = "";
    let top1=0;

```

```

let top2=0;
let top3=0;
let top1Serie="";
let top2Serie="";
let top3Serie="";
let top1Nombre="";
let top2Nombre="";
let datos=sistema.darConOpiniones();

for (let elem of datos){
    if (top1<elem.cantOpiniones){
        top1=elem.cantOpiniones
        top1Serie=elem
        top1Nombre=elem.nombre;
    }
}

for (let elem of datos){
    if (top2<elem.cantOpiniones && elem.nombre!=top1Nombre){
        top2=elem.cantOpiniones
        top2Serie=elem
        top2Nombre=elem.nombre
    }
}

for (let elem of datos){
    if (top3<elem.cantOpiniones && elem.nombre!=top1Nombre &&
elem.nombre!=top2Nombre){
        top3=elem.cantOpiniones
        top3Serie=elem
    }
}

if (datos.length==1){
    solo1(lista, top1Serie);
} else {
    if (datos.length==2){

```

```

        solo1(lista, top1Serie);
        solo2(lista, top2Serie);
    } else {
        solo1(lista, top1Serie);
        solo2(lista, top2Serie);
        solo3(lista, top3Serie);
    }
}
}

```

```

function solo1(lista, top1Serie){
    let nodo = document.createElement("li");
    let nodoTexto = document.createTextNode(top1Serie);
    nodo.appendChild(nodoTexto);
    lista.appendChild(nodo);
}

```

```

function solo2(lista, top2Serie){
    let nodo = document.createElement("li");
    let nodoTexto = document.createTextNode(top2Serie);
    nodo.appendChild(nodoTexto);
    lista.appendChild(nodo);
}

```

```

function solo3(lista, top3Serie){
    let nodo = document.createElement("li");
    let nodoTexto = document.createTextNode(top3Serie);
    nodo.appendChild(nodoTexto);
    lista.appendChild(nodo);
}

```

```

function sinOpinion (){
    let lista = document.getElementById("idSinOpinion");
    lista.innerHTML = "";
    let datos=sistema.darTodos();
}

```

```

for (let elem of datos){
    if (elem.cantOpiniones==0){
        let nodo = document.createElement("li");
        let nodoTexto = document.createTextNode(elem);
        nodo.appendChild(nodoTexto);
        lista.appendChild(nodo);
    }
}

```

Clases.JS:

// Felipe Rodriguez y Valentina Techera

```

class Series {
    constructor(){
        this.lista=[];
    }
    agregar(elem){
        this.lista.push(elem);
        this.lista.sort(function (a,b){
                                return a.ordenLetras(b)
                            })
    }
    darTodos(){
        return this.lista;
    }
    cantidadOpiniones(serie){
        let lista=this.lista;
        for (let elem of lista){
            if (elem.nombre==serie){
                elem.cantOpiniones++;
            }
        }
    }
    repetida(nombre){
        let repetido=false;

```

```

        for (let elem of this.lista){
            if (elem.nombre.toUpperCase() == nombre.toUpperCase()){
                repetido=true;
            }
        }
        return repetido;
    }

    modificarDescripcion(nombre, descripcion){
        for (let elem of this.lista){
            if (elem.nombre.toUpperCase() == nombre.toUpperCase()){
                elem.descripcion=descripcion
            }
        }
    }

    modificarTemporadas(nombre, temporadas){
        for (let elem of this.lista){
            if (elem.nombre.toUpperCase() == nombre.toUpperCase()){
                elem.temporadas=temporadas
            }
        }
    }

    modificarCapitulos(nombre, capitulos){
        for (let elem of this.lista){
            if (elem.nombre.toUpperCase()== nombre.toUpperCase()){
                elem.capitulos=capitulos
            }
        }
    }

    previo(nombre){
        let lugar = 0;
        for (let i = 0; i < this.lista.length; i++){
            if (this.lista[i].nombre == nombre){
                lugar = i-1
            }
            if (lugar<0){

```



```

        lugar = this.lista.length-1
    }
}
return this.lista[lugar].nombre;
}
siguiente(nombre){
    let lugar = 0;
    for (let i = 0; i < this.lista.length; i++){
        if (this.lista[i].nombre == nombre){
            lugar = i+1
        }
        if (lugar>this.lista.length-1){
            lugar = 0
        }
    }
    return this.lista[lugar].nombre;
}
descripcion(nombre){
    let respuesta="";
    for (let elem of this.lista){
        if (elem.nombre == nombre){
            respuesta = elem.descripcion
        }
    }
    return respuesta;
}
temporadas(nombre){
    let respuesta="";
    for (let elem of this.lista){
        if (elem.nombre == nombre){
            respuesta = elem.temporadas
        }
    }
    return respuesta;
}

```

```

    capitulos(nombre){
        let respuesta="";
        for (let elem of this.lista){
            if (elem.nombre == nombre){
                respuesta = elem.capitulos
            }
        }
        return respuesta;
    }
}

```

```

class características {
    constructor(nombre, descripcion, temporadas, capitulos){
        this.nombre = nombre;
        this.descripcion = descripcion;
        this.temporadas = temporadas;
        this.capitulos = capitulos;
    }
    toString(){
        return this.nombre;
    }
    ordenLetras(b){
        return this.nombre.localeCompare(b.nombre)
    }
}

```

```

class Opinion {
    constructor(nombre, comentario, temporada, capitulo, puntaje) {
        this.nombre = nombre;
        this.comentario = comentario;
        this.temporada = temporada;
        this.capitulo = capitulo;
        this.puntaje = puntaje;
    }
}

```

```
toString() {  
    return "Temp: " + this.temporada + " " + "Cap: " + this.capitulo + " " +  
    "Puntaje: " + this.puntaje + " " + this.comentario;  
}  
}
```

```
class Opiniones {  
    constructor(){  
        this.lista=[];  
    }  
    agregar(elem) {  
        this.lista.push(elem);  
    }  
  
    darTodos() {  
        return this.lista;  
    }  
  
    buscarPuntaje(nombre, temporada, capitulo){  
        let respuesta=0;  
        for (let elem of this.lista){  
            if (elem.nombre===nombre){  
                if (elem.temporada===temporada){  
                    if (elem.capitulo===capitulo){  
                        respuesta=elem.puntaje  
                    }  
                }  
            }  
        }  
        return respuesta;  
    }  
  
    repetida(nombre){  
        let repetido=false;  
        for (let elem of this.lista){  
            if (elem.nombre == nombre){
```

```

        repetido=true;
    }
}
return repetido;
}

repetidaTemporada(temporada){
    let repetido=false;
    for (let elem of this.lista){
        if (elem.temporada == temporada){
            repetido=true;
        }
    }
    return repetido;
}

repetidoCapitulo(capitulo){
    let repetido=false;
    for (let elem of this.lista){
        if (elem.capitulo == capitulo){
            repetido=true;
        }
    }
    return repetido;
}

modificarComentario(nombre, temporada, capitulo, comentario){
    for (let elem of this.lista){
        if (elem.nombre == nombre && elem.temporada==temporada
&& elem.capitulo==capitulo){
            elem.comentario=comentario
        }
    }
}

modificarPuntaje(nombre, temporada, capitulo, puntaje){
    for (let elem of this.lista){
        if (elem.nombre == nombre && elem.temporada==temporada
&& elem.capitulo==capitulo){
            elem.puntaje=puntaje

```

```

    }
  }
}

```

```

class Sistema {
  constructor(){
    this.lista=[];
  }
  agregar(elem){
    this.lista.push(elem);
    this.lista.sort(function (a,b){
      return a.ordenLetras(b)
    })
  }

  darTodos(){
    return this.lista;
  }

  darConOpiniones(){
    let respuesta=[];
    for (let elem of this.lista){
      if (elem.cantOpiniones>0){
        respuesta.push(elem)
      }
    }
    return respuesta;
  }

  darOrdenLetra(){
    let respuesta = this.lista.sort(function (a,b){
      return a.ordenLetras(b)
    })

    return respuesta;
  }
}

```

```

    }
    eleiminarPuntaje(nombre, puntaje){
        for (let elem of this.lista){
            if (elem.nombre===nombre){
                elem.sumaPuntaje=elem.sumaPuntaje-puntaje;
            }
        }
    }
    puntajePromedio(nombre, puntaje){
        for (let elem of this.lista){
            if (elem.nombre===nombre){
                elem.sumaPuntaje=elem.sumaPuntaje+puntaje;

                elem.promedio=elem.sumaPuntaje/elem.cantOpiniones;
            }
        }
    }
    cantidadOpiniones(serie){
        let lista=this.lista;
        for (let elem of lista){
            if (elem.nombre===serie){
                elem.cantOpiniones++;
            }
        }
    }
    darOrdenNumeros(){
        let respuesta = this.lista.sort(function (a,b){
            return a.ordenNumero(b)
        })

        return respuesta;
    }
    modificarDescripcion(nombre, descripcion){
        for (let elem of this.lista){
            if (elem.nombre.toUpperCase() == nombre.toUpperCase()){
                elem.descripcion=descripcion
            }
        }
    }

```

```

        }
    }
}
modificarTemporadas(nombre, temporadas){
    for (let elem of this.lista){
        if (elem.nombre.toUpperCase() == nombre.toUpperCase()){
            elem.temporadas=temporadas
        }
    }
}
modificarCapitulos(nombre, capitulos){
    for (let elem of this.lista){
        if (elem.nombre.toUpperCase()== nombre.toUpperCase()){
            elem.capitulos=capitulos
        }
    }
}
}

```

```

class caracteristicas1{
    constructor(nombre, descripcion, temporadas, capitulos, cantOpiniones,
sumaPuntaje, promedio){
        this.nombre=nombre;
        this.descripcion=descripcion;
        this.temporadas=temporadas;
        this.capitulos=capitulos;
        this.cantOpiniones=cantOpiniones;
        this.sumaPuntaje=sumaPuntaje;
        this.promedio=promedio;
    }

    ordenNumero(b){
        return b.cantOpiniones-this.cantOpiniones;
    }
}

```

```
ordenLetras(b){  
    return this.nombre.localeCompare(b.nombre);  
}
```

```
toString() {  
    return this.nombre + "-" + this.descripcion+ " (Temporadas:" +  
    this.temporadas + ", Capítulos: " + this.capitulos + ")";  
}  
}
```

CSS:

```
/* Felipe Rodriguez y Valentina Techera */
```

```
header{  
    background-color: #900c37;  
    color: white;  
    text-align: center;  
    border-radius: 30px 0px 30px 0px;  
    padding: 10px;  
  
}
```

```
nav{  
    float: left;  
    width: 24%;  
    margin-right: 1%;  
  
}
```

```
nav ul li {  
    margin: 5px auto;  
    list-style: none;  
    text-align: center;  
}
```

```
a:link{
```



```
        text-decoration:none;
    }
```

```
section {
        width:75%;
        float:right;

    }
```

```
.Series, .Series1{
        background: linear-gradient(90deg, rgba(0,0,0,0) 1%, rgba(242,213,71,1)
99%);
        border-radius: 0px 0px 50px 0px;
    }
```

```
.Opiniones, .Opiniones1{
        background: linear-gradient(90deg, rgba(0,0,0,0) 1%, rgba(235,97,75,1)
99%);
        border-radius: 0px 0px 30px 0px;

    }
```

```
.Estadisticas, .Estadisticas1{
        background: linear-gradient(90deg, rgba(0,0,0,0) 1%, rgba(198,47,84,1)
99%);
        border-radius: 0px 0px 30px 0px;
    }
```

```
table, th, td{
        text-align: center;
        border: 1px solid black;
    }
```

```
#idTabla{
        width: 700px;
        height:100px;
```

```
}
```

```
footer{  
    background-color: #900c37;  
    color: white;  
    text-align: center;  
    border-radius: 30px 0px 30px 0px;  
    padding: 10px;  
    clear:both;  
}
```

```
@media screen and (max-width:600px)  
{  
    nav{width: 100%;}  
}
```