

Promise

El objeto promise o promesa representa un valor que puede estar disponible ahora, en el futuro o nunca. (¡Igual que una promesa en la vida real!). Se utiliza cuando estamos frente a un código asíncrono. Ya que, valga la redundancia, este objeto "nos promete" que devolverá un valor en una línea de tiempo presente o futura y recuerda el contexto en donde se ejecuta, es decir, sabe con precisión en qué punto se ha de resolver un valor o lanzar un error.

Para la utilización de promesas vamos a poder utilizare diferentes formas, vamos a empezar viendo una primera forma, que nos resulte fácil de entender.

```
Para crear una promesa deberemos poner
variable= new Promise(nombre-función-1)
```

Creamos una promesa e indicamos el nombre de una función, que contiene el código que se va a ejecutar cuando se realiza la promesa (la acción que prometemos que vamos a desarrollar).

ejemplo, creamos una promesa, que cuando se realice se va a ejecutar la función tratar

```
001    let mipromesa=new Promise(tratar);
```

Crearemos una promesa en un función y dicha función devolver la promesa.

```
function crearPromesa(){
       let variable= new Promise(nombre-función-1);
       return variable
}
```

ejemplo, tenemos la función crear que crea una promesa, que cuando se realice se va a ejecutar la función tratar y que la función crear devuelve la promesa que hemos creado.

7	.)
001	<pre>function crear(){</pre>
002	<pre>let mipromesa=new Promise(tratar);</pre>
003	return mipromesa;
004	}

Deberemos crearnos una función que va a ser la encargada de prometer la promera, esto es, indicar que se realice la promesa.

```
function nombre-función(){
       crearPromesa().then(función-resuelve).catch(función-error);
}
       O bien
function nombre-función(){
       crearPromesa()
               .then(función-resuelve)
               .catch(función-error);
}
```

En el método then deberemos poner el nombre de una función, que está declarada más adelante y que se va a encargar de realizar las acciones prometidas.



En el método **catch** deberemos poner el nombre de una función, que está declarada más adelante y que se va a encargar de realizar las acciones que se van a realizar cuando se rechaza la promesa o hay un error en la misma.

001	<pre>function enviar(){</pre>
002	crear()
003	.then(correcto)
004	.catch(errores)
005	}

Vamos a declarar la función que se va a ejecutar cuando se realiza la promesa, lo que prometemos que vamos a hacer.

Esta función va a recibir dos parámetros:

- El primer parámetro se corresponde con el nombre simbólico de una función, que se va a mandar ejecutar cuando se acepta la promesa, no existe una función con ese nombre, solamente se utiliza en esta función para hacer la llamada a la función de resolución de la promesa.
- El segundo parámetro se corresponde con el nombre simbólico de una función, que se va a mandar ejecutar cuando se rechaza la promesa o cuando existe un error, no existe una función con ese nombre, solamente se utiliza en esta función para hacer la llamada a la función de rechazo o error de la promesa.

En el cuerpo de la función vamos a indicar que deseamos hacer y en un momento llamaremos a la función que resuelve la promesa, en este caso se la puede pasar varios parámetros, y en otro momento llamaremos a la función que rechaza la promesa.

Ejemplo: en este caso cogemos el valor de dos cajas de texto, si el valor de ambas es distinto de la cadena vacía llamamos a la función resolver con el valor de las dos cajas de texto y si no se llama a la función rechazar.

001	<pre>function tratar(resolver, rechazar){</pre>
002	<pre>let nom=document.getElementById("nombre").value.trim();</pre>
003	<pre>let ape=document.getElementById("apellidos").value.trim();</pre>
004	if(nom != "" && ape != "")
005	resolver(nom , ape)
006	else
007	rechazar();
800	1

A continuación nos declaramos la función que resuelve la promesa

Ejemplo: mostramos en una caja de texto la concatenación de los dos parámetros

001	<pre>function correcto(uno,dos){</pre>
002	<pre>document.getElementById("resultado").value=uno + " " + dos;</pre>
003	}

Y por último nos declaramos la función de rechazo de la promesa o de error de la promesa



Ejemplo: mostramos en una caja de texto un mensaje de error.

001	<pre>function errores(){</pre>
002	<pre>document.getElementById("resultado").value="Al menos uno de los campos está</pre>
	vacío";
003	}

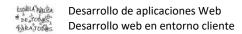
Ahora vamos a ver el programa completo

promise101.is

	ise101.js
001	<pre>if (document.addEventListener)</pre>
002	<pre>window.addEventListener("load",inicio)</pre>
003	else if (document.attachEvent)
004	<pre>window.attachEvent("onload",inicio);</pre>
005	function inicio() {
006	<pre>let boton=document.getElementById("poner");</pre>
007	<pre>if (document.addEventListener)</pre>
800	boton.addEventListener("click",enviar)
009	else if (document.attachEvent)
010	<pre>boton.attachEvent("onclick",enviar);</pre>
011	}
012	<pre>function enviar(){</pre>
013	crear()
014	.then(correcto)
015	.catch(errores)
016	}
017	<pre>function crear(){</pre>
018	<pre>let mipromesa= new Promise(tratar);</pre>
019	return mipromesa;
020	}
021	<pre>function tratar(resolver, rechazar){</pre>
022	<pre>let nom=document.getElementById("nombre").value.trim();</pre>
023	<pre>let ape=document.getElementById("apellidos").value.trim();</pre>
024	if (nom!="" && ape!="")
025	resolver(nom,ape)
026	else
027	rechazar();
028	}
029	function correcto(uno,dos) {
030	<pre>document.getElementById("resultado").value=uno + " " + dos;</pre>
031	}
032	function errores(){
033	document.getElementById("resultado").value="Al menos uno de los campos está
034	vacío";
034	I I

Vamos a ver varios ejemplos del mismo programa haciéndolo de otra forma promise102.js

001 if (document.addEventListener) window.addEventListener("load",inicio) 002 003 else if (document.attachEvent) window.attachEvent("onload",inicio); 004 005 function inicio(){ 006 let boton=document.getElementById("poner"); 007 if (document.addEventListener) 800 boton.addEventListener("cl ,enviar) 009 else if (document.attachEvent) 010 boton.attachEvent("onclick",enviar); 011 012 function enviar(){ 013 crear() 014 .then(correcto) 015 .catch (errores) 016 017 function crear(){ 018 let mipromesa= new Promise(019 function (resolver, rechazar){ 020 let nom=document.getElementById("nombre").value.trim(); let ape=document.getElementById("apellidos").value.trim();
if (nom!="" && ape!="") 021 022



7.- Utilización de mecanismos de comunicación asíncrona (AJAX)-Asynchronous Javascript and XML) Promise.



023	resolver(nom+ " " +ape) // solo se puede enviar un parámetro
024	else
025	rechazar();
026	}
027);
028	return mipromesa;
029	}
030	function correcto(uno) {
031	<pre>document.getElementById("resultado").value=uno;</pre>
032	}
033	<pre>function errores(){</pre>
034	document.getElementById("resultado").value="Al menos uno de los campos está
	vacío";
035	}

promise103.js

```
001 if (document.addEventListener)
002
         window.addEventListener("1
                                      ad",inicio)
003
     else if (document.attachEvent)
004
          window.attachEvent("onload",inicio);
005 function inicio(){
         let boton=document.getElementById("poner");
006
007
          if (document.addEventListener)
             boton.addEventListener("click",enviar)
800
009
         else if (document.attachEvent)
010
             boton.attachEvent("onclick",enviar);
011
012
     function enviar(){
013
          crear()
014
              .then(function (uno){
015
                  document.getElementById("resultado").value=uno;
016
              })
017
              .catch(function () {
018
                  document.getElementById("resultado").value="Al menos uno de los campos
     está vacío";
019
             })
020
021
     function crear(){
022
          let mipromesa= new Promise(
023
              function (resolver, rechazar) {
                  let nom=document.getElementById("nombre").value.trim();
024
025
                  let ape=document.getElementById("apellidos").value.trim();
026
                  if (nom!="" && ape!="")
027
                      resolver(nom+
                                         +ape) // solo se puede enviar un parámetro
028
029
                      rechazar();
030
031
032
         return mipromesa;
033
```

promise104.is

promi	56104.55
001	<pre>if (document.addEventListener)</pre>
002	<pre>window.addEventListener("load",inicio)</pre>
003	else if (document.attachEvent)
004	<pre>window.attachEvent("onload",inicio);</pre>
005	function inicio(){
006	<pre>let boton=document.getElementById("poner");</pre>
007	<pre>if (document.addEventListener)</pre>
800	boton.addEventListener("click",enviar)
009	else if (document.attachEvent)
010	boton.attachEvent("onclick",enviar);
011	}
012	<pre>function enviar(){</pre>
013	crear()
014	.then((uno)=>{
015	<pre>document.getElementById("resultado").value=uno;</pre>
016	})
017	.catch(()=>{
018	<pre>document.getElementById("resultado").value="Al menos uno de los campos</pre>
	está vacío";
019	})



020	}
021	<pre>function crear(){</pre>
022	let mipromesa= new Promise(
023	(resolver, rechazar) => {
024	<pre>let nom=document.getElementById("nombre").value.trim();</pre>
025	<pre>let ape=document.getElementById("apellidos").value.trim();</pre>
026	if (nom!="" && ape!="")
027	resolver(nom+ " " +ape) // solo se puede enviar un parámetro
028	else
029	rechazar();
030	}
031);
032	return mipromesa;
033	}

promise105.is

ргони	ise105.js
001	if (document.addEventListener)
002	<pre>window.addEventListener("load",inicio)</pre>
003	else if (document.attachEvent)
004	<pre>window.attachEvent("onload",inicio);</pre>
005	function inicio(){
006	<pre>let boton=document.getElementById("poner");</pre>
007	<pre>if (document.addEventListener)</pre>
800	boton.addEventListener("click",enviar)
009	else if (document.attachEvent)
010	<pre>boton.attachEvent("onclick",enviar);</pre>
011	}
012	<pre>function enviar(){</pre>
013	crear()
014	.then(uno=>{
015	<pre>document.getElementById("resultado").value=uno;</pre>
016	})
017	.catch(() =>{
018	<pre>document.getElementById("resultado").value="Al menos uno de los campos</pre>
010	está vacío";
019	})
020	1
021	<pre>function crear(){</pre>
022	<pre>let mipromesa= new Promise(</pre>
023	(resolver, rechazar) => {
024	<pre>let nom=document.getElementById("nombre").value.trim();</pre>
025	<pre>let ape=document.getElementById("apellidos").value.trim();</pre>
026	if (nom!="" && ape!="")
027	resolver(nom+ " " +ape) // solo se puede enviar un parámetro
028	else
029	rechazar();
030	}
031);
032	return mipromesa;
033	}

Una segunda forma de crear una promesa es:

Para crear una promesa deberemos poner variable= new Promise(nombre-función-1)

Creamos una promesa e indicamos el nombre de una función, que contiene el código que se va a ejecutar cuando se realiza la promesa (la acción que prometemos que vamos a desarrollar).

```
ejemplo, creamos una promesa, que cuando se realice se va a ejecutar la función tratar

O01 let mipromesa=new Promise (tratar);
```

A continuación, a la variable de la promesa le vamos a aplicar los métodos then y catch.

variable



55 III (54)

En el método **then** deberemos poner el nombre de una función, que está declarada más adelante y que se va a encargar de realizar las acciones prometidas.

En el método **catch** deberemos poner el nombre de una función, que está declarada más adelante y que se va a encargar de realizar las acciones que se van a realizar cuando se rechaza la promesa o hay un error en la misma.

001	mipromesa
002	.then(correcto)
003	.catch(errores);

Vamos a declarar la función que se va a ejecutar cuando se realiza la promesa, lo que prometemos que vamos a hacer.

Esta función va a recibir dos parámetros:

- El primer parámetro se corresponde con el nombre simbólico de una función, que se va a mandar ejecutar cuando se acepta la promesa, no existe una función con ese nombre, solamente se utiliza en esta función para hacer la llamada a la función de resolución de la promesa.
- El segundo parámetro se corresponde con el nombre simbólico de una función, que se va a mandar ejecutar cuando se rechaza la promesa o cuando existe un error, no existe una función con ese nombre, solamente se utiliza en esta función para hacer la llamada a la función de rechazo o error de la promesa.

En el cuerpo de la función vamos a indicar que deseamos hacer y en un momento llamaremos a la función que resuelve la promesa, en este caso se la puede pasar varios parámetros, y en otro momento llamaremos a la función que rechaza la promesa.

Ejemplo: en este caso cogemos el valor de dos cajas de texto, si el valor de ambas es distinto de la cadena vacía llamamos a la función resolver con el valor de las dos cajas de texto y si no se llama a la función rechazar.

Humui	namamos a la junción resolver con el valor de las dos cajas de texto y si no se nama a la junción rechazar.	
001	<pre>function tratar(resolver, rechazar){</pre>	
002	<pre>let nom=document.getElementById("nombre").value.trim();</pre>	
003	<pre>let ape=document.getElementById("apellidos").value.trim();</pre>	
004	if(nom != "" && ape != "")	
005	resolver(nom , ape)	
006	else	
007	rechazar();	
800	}	

A continuación nos declaramos la función que resuelve la promesa

Ejemplo: mostramos en una caja de texto la concatenación de los dos parámetros

Bjempio. Mosti amos en ana caja de texto la concatenación de los dos parametros		
001	<pre>function correcto(uno,dos){</pre>	
002	<pre>document.getElementById("resultado").value=uno + " " + dos;</pre>	
003	}	



Y por último nos declaramos la función de rechazo de la promesa o de error de la promesa

Ejemplo: mostramos en una caja de texto un mensaje de error.

001	<pre>function errores(){</pre>
002	document.getElementById("resultado").value="Al menos uno de los campos está
	vacío";
003	1

Ahora vamos a ver el programa completo

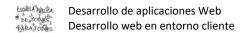
promise111.js

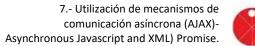
	, ,
001	<pre>if (document.addEventListener)</pre>
002	<pre>window.addEventListener("load",inicio)</pre>
003	else if (document.attachEvent)
004	<pre>window.attachEvent("onload",inicio);</pre>
005	function inicio(){
006	<pre>let boton=document.getElementById("poner");</pre>
007	<pre>if (document.addEventListener)</pre>
800	boton.addEventListener("click",crear)
009	else if (document.attachEvent)
010	<pre>boton.attachEvent("onclick",crear);</pre>
011	}
012	<pre>function crear(){</pre>
013	<pre>let mipromesa= new Promise(tratar);</pre>
014	mipromesa
015	.then(correcto)
016	<pre>.catch(errores);</pre>
017	}
017 018	<pre>function tratar(resolver, rechazar){</pre>
	•
018	<pre>function tratar(resolver, rechazar){</pre>
018 019	<pre>function tratar(resolver, rechazar) { let nom=document.getElementById("nombre").value.trim();</pre>
018 019 020	<pre>function tratar(resolver, rechazar) { let nom=document.getElementById("nombre").value.trim(); let ape=document.getElementById("apellidos").value.trim();</pre>
018 019 020 021	<pre>function tratar(resolver, rechazar){ let nom=document.getElementById("nombre").value.trim(); let ape=document.getElementById("apellidos").value.trim(); if (nom!="" && ape!="")</pre>
018 019 020 021 022	<pre>function tratar(resolver, rechazar){ let nom=document.getElementById("nombre").value.trim(); let ape=document.getElementById("apellidos").value.trim(); if (nom!="" && ape!="") resolver(nom + " " + ape)</pre>
018 019 020 021 022 023	<pre>function tratar(resolver, rechazar){ let nom=document.getElementById("nombre").value.trim(); let ape=document.getElementById("apellidos").value.trim(); if (nom!="" && ape!="") resolver(nom + " " + ape) else</pre>
018 019 020 021 022 023 024	<pre>function tratar(resolver, rechazar){ let nom=document.getElementById("nombre").value.trim(); let ape=document.getElementById("apellidos").value.trim(); if (nom!="" && ape!="") resolver(nom + " " + ape) else rechazar(); } function correcto(uno,dos){</pre>
018 019 020 021 022 023 024 025	<pre>function tratar(resolver, rechazar){ let nom=document.getElementById("nombre").value.trim(); let ape=document.getElementById("apellidos").value.trim(); if (nom!="" && ape!="") resolver(nom + " " + ape) else rechazar(); }</pre>
018 019 020 021 022 023 024 025	<pre>function tratar(resolver, rechazar){ let nom=document.getElementById("nombre").value.trim(); let ape=document.getElementById("apellidos").value.trim(); if (nom!="" && ape!="") resolver(nom + " " + ape) else rechazar(); } function correcto(uno,dos){</pre>
018 019 020 021 022 023 024 025 026	<pre>function tratar(resolver, rechazar){ let nom=document.getElementById("nombre").value.trim(); let ape=document.getElementById("apellidos").value.trim(); if (nom!="" && ape!="") resolver(nom + " " + ape) else rechazar(); } function correcto(uno,dos){ document.getElementById("resultado").value=uno; } function errores(){</pre>
018 019 020 021 022 023 024 025 026 027	<pre>function tratar(resolver, rechazar){ let nom=document.getElementById("nombre").value.trim(); let ape=document.getElementById("apellidos").value.trim(); if (nom!="" && ape!="") resolver(nom + " " + ape) else rechazar(); } function correcto(uno,dos){ document.getElementById("resultado").value=uno; }</pre>
018 019 020 021 022 023 024 025 026 027 028 029	<pre>function tratar(resolver, rechazar){ let nom=document.getElementById("nombre").value.trim(); let ape=document.getElementById("apellidos").value.trim(); if (nom!="" && ape!="") resolver(nom + " " + ape) else rechazar(); } function correcto(uno,dos){ document.getElementById("resultado").value=uno; } function errores(){</pre>
018 019 020 021 022 023 024 025 026 027 028	<pre>function tratar(resolver, rechazar){ let nom=document.getElementById("nombre").value.trim(); let ape=document.getElementById("apellidos").value.trim(); if (nom!="" && ape!="") resolver(nom + " " + ape) else rechazar(); } function correcto(uno,dos){ document.getElementById("resultado").value=uno; } function errores(){ document.getElementById("resultado").value="Al menos uno de los campos está</pre>

Vamos a ver varios ejemplos del mismo programa haciéndolo de otra forma

promise112.js

001	<pre>if (document.addEventListener)</pre>
002	<pre>window.addEventListener("load",inicio)</pre>
003	else if (document.attachEvent)
004	<pre>window.attachEvent("onload",inicio);</pre>
005	function inicio() {
006	<pre>let boton=document.getElementById("poner");</pre>
007	<pre>if (document.addEventListener)</pre>
800	boton.addEventListener("click",crear)
009	else if (document.attachEvent)
010	<pre>boton.attachEvent("onclick",crear);</pre>
011	}
012	<pre>function crear(){</pre>
013	<pre>let mipromesa= new Promise(</pre>
014	<pre>function (resolver, rechazar) {</pre>
015	<pre>let nom=document.getElementById("nombre").value.trim();</pre>
016	<pre>let ape=document.getElementById("apellidos").value.trim();</pre>
017	if (nom!="" && ape!="")
018	resolver(nom + " " + ape)
019	else







```
020
                      rechazar();
021
              }
022
         ) :
023
         mipromesa
024
              .then(function (uno ){
025
                 document.getElementById("resultado").value=uno;
026
027
              .catch(function () {
                 document.getElementById("resultado").value="Al menos uno de los campos
028
029
             });
030
```

promise113.js

```
001 if (document.addEventListener)
002
          window.addEventListener("load",inicio)
003
      else if (document.attachEvent)
004
          window.attachEvent("onload", inicio);
005
     function inicio(){
006
          let boton=document.getElementById("poner");
007
          if (document.addEventListener)
800
              boton.addEventListener("click",crear)
009
          else if (document.attachEvent)
010
              boton.attachEvent("onclick'
                                            crear);
011 }
012
      function crear(){
          let mipromesa= new Promise( (resolver, rechazar)=>{
    let nom=document.getElementById("nombre").value.trim();
013
014
015
                   let ape=document.getElementById("apellidos").value.trim();
                   if (nom!="" && ape!="")
016
                       resolver(nom + "
017
018
                   else
019
                       rechazar();
020
021
          );
022
          mipromesa
023
               .then( (uno )=> {
024
                   document.getElementById("resultado").value=uno;
025
               })
026
               .catch(()=>{
027
                   document.getElementById("resultado").value="Al menos uno de los campos
      está vacío";
028
              });
029
```

nromise114 is

ргош	ISE114.JS
001	<pre>if (document.addEventListener)</pre>
002	<pre>window.addEventListener("load",inicio)</pre>
003	else if (document.attachEvent)
004	<pre>window.attachEvent("onload",inicio);</pre>
005	function inicio(){
006	<pre>let boton=document.getElementById("poner");</pre>
007	<pre>if (document.addEventListener)</pre>
800	boton.addEventListener("click",crear)
009	else if (document.attachEvent)
010	<pre>boton.attachEvent("onclick",crear);</pre>
011	}
012	<pre>function crear(){</pre>
013	<pre>let mipromesa= new Promise((resolver, rechazar)=>{</pre>
014	<pre>let nom=document.getElementById("nombre").value.trim();</pre>
015	<pre>let ape=document.getElementById("apellidos").value.trim();</pre>
016	if (nom!="" && ape!="")
017	resolver(nom + " " + ape)
018	else
019	rechazar();
020	}
021);
022	mipromesa
023	.then(uno => {
024	<pre>document.getElementById("resultado").value=uno;</pre>
025	1)
026	.catch(()=>{



027	<pre>document.getElementById("resultado").value="Al menos uno de los campos</pre>
- 30	está vacío";
028));
029	}

Una tercera forma de crear una promesa es:

En una función devolvemos una promesa que creamos en ese preciso momento.

```
function nombre-función(){
     return new Promise(nombre-función-1);
}
```

Creamos una promesa e indicamos el nombre de una función, que contiene el código que se va a ejecutar cuando se realiza la promesa (la acción que prometemos que vamos a desarrollar) y devolvemos la promesa creada en la misma línea.

001	function cr	ear(){
002	return	<pre>new Promise(tratar);</pre>
003	}	

Deberemos crearnos una función que va a ser la encargada de prometer la promera, esto es, indicar que se realice la promesa.

En el método **then** deberemos poner el nombre de una función, que está declarada más adelante y que se va a encargar de realizar las acciones prometidas.

En el método **catch** deberemos poner el nombre de una función, que está declarada más adelante y que se va a encargar de realizar las acciones que se van a realizar cuando se rechaza la promesa o hay un error en la misma.

001	<pre>function enviar(){</pre>	
002	crear()	
003	.then(correcto)	
004	.catch(errores)	
005	}	

Vamos a declarar la función que se va a ejecutar cuando se realiza la promesa, lo que prometemos que vamos a hacer.



Esta función va a recibir dos parámetros:

- El primer parámetro se corresponde con el nombre simbólico de una función, que se va a mandar ejecutar cuando se acepta la promesa, no existe una función con ese nombre, solamente se utiliza en esta función para hacer la llamada a la función de resolución de la promesa.
- El segundo parámetro se corresponde con el nombre simbólico de una función, que se va a mandar ejecutar cuando se rechaza la promesa o cuando existe un error, no existe una función con ese nombre, solamente se utiliza en esta función para hacer la llamada a la función de rechazo o error de la promesa.

En el cuerpo de la función vamos a indicar que deseamos hacer y en un momento llamaremos a la función que resuelve la promesa, en este caso se la puede pasar varios parámetros, y en otro momento llamaremos a la función que rechaza la promesa.

Ejemplo: en este caso cogemos el valor de dos cajas de texto, si el valor de ambas es distinto de la cadena vacía llamamos a la función resolver con el valor de las dos cajas de texto y si no se llama a la función rechazar.

110177101	mos a la janeien receiver con el valor de las ales cajas de cente y el ne se hama a la janeien rechasan
001	<pre>function tratar(resolver, rechazar){</pre>
002	<pre>let nom=document.getElementById("nombre").value.trim();</pre>
003	<pre>let ape=document.getElementById("apellidos").value.trim();</pre>
004	if(nom != "" && ape != "")
005	resolver(nom , ape)
006	else
007	rechazar();
800	}

A continuación nos declaramos la función que resuelve la promesa

Ejemplo: mostramos en una caja de texto la concatenación de los dos parámetros

001	function correcto(uno,dos){
002	<pre>document.getElementById("resultado").value=uno + " " + dos;</pre>
003	}

Y por último nos declaramos la función de rechazo de la promesa o de error de la promesa

Ejemplo: mostramos en una caja de texto un mensaje de error.

001	function errores(){
002	<pre>document.getElementById("resultado").value="Al menos uno de los campos está</pre>
	vacío";
003	}

Ahora vamos a ver el programa completo

promise121.js

001	<pre>if (document.addEventListener)</pre>
002	<pre>window.addEventListener("load",inicio)</pre>
003	else if (document.attachEvent)
004	<pre>window.attachEvent("onload",inicio);</pre>
005	<pre>function inicio(){</pre>
006	<pre>let boton=document.getElementById("poner");</pre>
007	<pre>if (document.addEventListener)</pre>
800	boton.addEventListener("click",enviar)
009	else if (document.attachEvent)
010	<pre>boton.attachEvent("onclick",enviar);</pre>



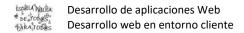
```
011 }
012 | function enviar(){
013
          crear()
014
               .then (correcto)
015
               .catch(errores)
016 }
017
     function crear(){
018
          return new Promise (tratar);
019
      function tratar(resolver, rechazar){
   let nom=document.getElementById("nombre").value.trim();
020
021
          let ape=document.getElementById("apellidos").value.trim();
022
          if (nom!="" && ape!="")
023
              resolver(nom + "
024
                                    + ape)
025
          else
026
              rechazar();
027
028 function correcto(uno){
029
          document.getElementById("resultado").value=uno;
030
031
     function errores(){
032
          document.getElementById("resultado").value="Al menos uno de los campos está
      vacío";
033
     }
```

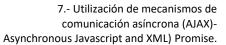
promise122.js

```
001 if (document.addEventListener)
002
          window.addEventListener("load",inicio)
003
      else if (document.attachEvent)
004
          window.attachEvent("onload", inicio);
      function inicio(){
005
006
          let boton=document.getElementById("poner");
007
          if (document.addEventListener)
008
              boton.addEventListener("click",enviar)
009
           else if (document.attachEvent)
010
              boton.attachEvent("onclick",enviar);
011
012 function enviar() {
013
          crear()
014
               .then(correcto)
015
               .catch(errores)
016 }
017
      function crear(){
018
          return new Promise (
019
               function (resolver, rechazar){
                   let nom=document.getElementById("nombre").value.trim();
let ape=document.getElementById("apellidos").value.trim();
020
021
                   if (nom!="" && ape!="")
022
                                         " +ape) // solo se puede enviar un parámetro
                       resolver(nom+ "
023
024
025
                       rechazar();
026
027
          ) ;
028
029
      function correcto(uno){
030
          document.getElementById("resultado").value=uno;
031
032
      function errores(){
033
          document.getElementById("resultado").value="Al menos uno de los campos está
034
     }
```

promise123.js

001	<pre>if (document.addEventListener)</pre>
002	<pre>window.addEventListener("load",inicio)</pre>
003	else if (document.attachEvent)
004	<pre>window.attachEvent("onload",inicio);</pre>
005	function inicio(){
006	<pre>let boton=document.getElementById("poner");</pre>
007	<pre>if (document.addEventListener)</pre>
008	boton.addEventListener("click",enviar)
009	else if (document.attachEvent)







010	<pre>boton.attachEvent("onclick",enviar);</pre>
011	}
012	<pre>function enviar(){</pre>
013	crear()
014	.then(
015	function (uno) {
016	<pre>document.getElementById("resultado").value=uno;</pre>
017	})
018	.catch(
019	function () {
020	<pre>document.getElementById("resultado").value="Al menos uno de los</pre>
	campos está vacío";
021	1);
022	}
023	<pre>function crear(){</pre>
024	return new Promise(
025	function (resolver, rechazar) {
026	<pre>let nom=document.getElementById("nombre").value.trim();</pre>
027	<pre>let ape=document.getElementById("apellidos").value.trim();</pre>
028	if (nom!="" && ape!="")
029	resolver(nom + " " + ape)
030	else
031	rechazar();
032));
033	}
prom	ise124.js

promi	ISE124.JS
001	<pre>if (document.addEventListener)</pre>
002	<pre>window.addEventListener("load",inicio)</pre>
003	<pre>else if (document.attachEvent)</pre>
004	<pre>window.attachEvent("onload",inicio);</pre>
005	function inicio()(
006	<pre>let boton=document.getElementById("poner");</pre>
007	<pre>if (document.addEventListener)</pre>
800	<pre>boton.addEventListener("click",enviar)</pre>
009	<pre>else if (document.attachEvent)</pre>
010	<pre>boton.attachEvent("onclick",enviar);</pre>
011	}
012	<pre>function enviar(){</pre>
013	crear()
014	.then(
015	(uno)=>{
016	<pre>document.getElementById("resultado").value=uno;</pre>
017	})
018	.catch(
019	()=>{
020	<pre>document.getElementById("resultado").value="Al menos uno de los</pre>
	campos está vacío";
021	3);
022	}
023	<pre>function crear(){</pre>
024	return new Promise(
025	<pre>(resolver, rechazar)=>{</pre>
026	<pre>let nom=document.getElementById("nombre").value.trim();</pre>
027	<pre>let ape=document.getElementById("apellidos").value.trim();</pre>
028	if (nom!="" && ape!="")
029	resolver(nom + " " + ape)
030	else
031	rechazar();
032	<pre>});</pre>
033	}

promise125.js

001	<pre>if (document.addEventListener)</pre>
002	<pre>window.addEventListener("load",inicio)</pre>
003	else if (document.attachEvent)
004	<pre>window.attachEvent("onload",inicio);</pre>
005	function inicio(){
006	<pre>let boton=document.getElementById("poner");</pre>
007	<pre>if (document.addEventListener)</pre>
800	boton.addEventListener("click",enviar)
009	else if (document.attachEvent)



010	<pre>boton.attachEvent("onclick",enviar);</pre>
011	}
012	<pre>function enviar(){</pre>
013	crear()
014	.then(
015	uno =>{
016	<pre>document.getElementById("resultado").value=uno;</pre>
017	})
018	.catch(
019	()=>{
020	<pre>document.getElementById("resultado").value="Al menos uno de los</pre>
	campos está vacío";
021));
022	}
023	<pre>function crear(){</pre>
024	return new Promise(
025	(resolver, rechazar)=>{
026	<pre>let nom=document.getElementById("nombre").value.trim();</pre>
027	<pre>let ape=document.getElementById("apellidos").value.trim();</pre>
028	if (nom!="" && ape!="")
029	resolver(nom + " " + ape)
030	else
031	rechazar();
032	D;
033	}

Una cuarta forma de crear una promesa es:

```
Para crear una promesa deberemos poner 
variable= new Promise(nombre-función-1)
.then(función-resuelve)
.catch(función-error);
```

Creamos una promesa e indicamos el nombre de una función, que contiene el código que se va a ejecutar cuando se realiza la promesa (la acción que prometemos que vamos a desarrollar), con lo métodos **then** y **catch**.

En el método **then** deberemos poner el nombre de una función, que está declarada más adelante y que se va a encargar de realizar las acciones prometidas.

En el método **catch** deberemos poner el nombre de una función, que está declarada más adelante y que se va a encargar de realizar las acciones que se van a realizar cuando se rechaza la promesa o hay un error en la misma.

001	<pre>function enviar(){</pre>
002	<pre>let mipromesa= new Promise(tratar)</pre>
003	.then(correcto)
004	.catch(errores)
005	3

Vamos a declarar la función que se va a ejecutar cuando se realiza la promesa, lo que prometemos que vamos a hacer.

Esta función va a recibir dos parámetros:

 El primer parámetro se corresponde con el nombre simbólico de una función, que se va a mandar ejecutar cuando se acepta la promesa, no existe una función con ese nombre, solamente se utiliza en esta función para hacer la llamada a la función de resolución de la promesa.



 El segundo parámetro se corresponde con el nombre simbólico de una función, que se va a mandar ejecutar cuando se rechaza la promesa o cuando existe un error, no existe una función con ese nombre, solamente se utiliza en esta función para hacer la llamada a la función de rechazo o error de la promesa.

En el cuerpo de la función vamos a indicar que deseamos hacer y en un momento llamaremos a la función que resuelve la promesa, en este caso se la puede pasar varios parámetros, y en otro momento llamaremos a la función que rechaza la promesa.

Ejemplo: en este caso cogemos el valor de dos cajas de texto, si el valor de ambas es distinto de la cadena vacía llamamos a la función resolver con el valor de las dos cajas de texto y si no se llama a la función rechazar.

001	<pre>function tratar(resolver, rechazar){</pre>
002	<pre>let nom=document.getElementById("nombre").value.trim();</pre>
003	<pre>let ape=document.getElementById("apellidos").value.trim();</pre>
004	if(nom != "" && ape != "")
005	resolver(nom , ape)
006	else
007	rechazar();
800	}

A continuación nos declaramos la función que resuelve la promesa

Ejemplo: mostramos en una caja de texto la concatenación de los dos parámetros

001	<pre>function correcto(uno,dos){</pre>
002	<pre>document.getElementById("resultado").value=uno + " " + dos;</pre>
003	}

Y por último nos declaramos la función de rechazo de la promesa o de error de la promesa

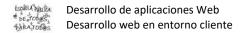
Ejemplo: mostramos en una caja de texto un mensaje de error.

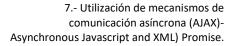
001	<pre>function errores(){</pre>
002	<pre>document.getElementById("resultado").value="Al menos uno de los campos está</pre>
	vacío";
003	}

Ahora vamos a ver el programa completo

promise131.js

001	<pre>if (document.addEventListener)</pre>
002	<pre>window.addEventListener("load",inicio)</pre>
003	else if (document.attachEvent)
004	<pre>window.attachEvent("onload",inicio);</pre>
005	function inicio(){
006	<pre>let boton=document.getElementById("poner");</pre>
007	<pre>if (document.addEventListener)</pre>
800	boton.addEventListener("click",enviar)
009	else if (document.attachEvent)
010	<pre>boton.attachEvent("onclick",enviar);</pre>
011	}
012	<pre>function enviar(){</pre>
013	<pre>let mipromesa= new Promise(tratar)</pre>
014	.then(correcto)
015	.catch(errores)
016	}







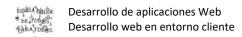
```
017 | function tratar(resolver, rechazar){
018
          let nom=document.getElementById("nombre").value.trim();
          let ape=document.getElementById("apellidos").value.trim();
if (nom!="" && ape!="")
019
020
              resolver(nom + "
021
                                   + ape)
022
          else
023
              rechazar();
024 }
025
    function correcto(uno,dos){
026
          document.getElementById("resultado").value=uno;
027
028
     function errores(){
029
          document.getElementById("resultado").value="Al menos uno de los campos está
030 }
```

promise132.js

```
001 if (document.addEventListener)
          window.addEventListener("load",inicio)
002
003
     else if (document.attachEvent)
          window.attachEvent("onload",inicio);
004
005 | function inicio(){
006
          let boton=document.getElementById("poner");
007
          if (document.addEventListener)
800
              boton.addEventListener("click",crear)
009
          else if (document.attachEvent)
              boton.attachEvent("onclick",crear);
010
011
012 function crear(){
013
          let mipromesa= new Promise(
              function (resolver, rechazar){
014
015
                  let nom=document.getElementById("nombre").value.trim();
                  let ape=document.getElementById("apellidos").value.trim();
if (nom!="" && ape!="")
016
017
018
                      resolver(nom + "
                                           + ape)
019
                  else
020
                      rechazar();
021
              }
022
023
              .then(function (uno ){
                  document.getElementById("resultado").value=uno;
024
025
026
              .catch(function (){
                  document.getElementById("resultado").value="Al menos uno de los campos
027
      está vacío";
028
              });
029
```

nromico122 ic

promi	ise133.js
001	<pre>if (document.addEventListener)</pre>
002	<pre>window.addEventListener("load",inicio)</pre>
003	else if (document.attachEvent)
004	<pre>window.attachEvent("onload",inicio);</pre>
005	function inicio(){
006	<pre>let boton=document.getElementById("poner");</pre>
007	<pre>if (document.addEventListener)</pre>
800	boton.addEventListener("click",crear)
009	else if (document.attachEvent)
010	<pre>boton.attachEvent("onclick",crear);</pre>
011	}
012	<pre>function crear(){</pre>
013	<pre>let mipromesa= new Promise(</pre>
014	(resolver, rechazar) => {
015	<pre>let nom=document.getElementById("nombre").value.trim();</pre>
016	<pre>let ape=document.getElementById("apellidos").value.trim();</pre>
017	if (nom!="" && ape!="")
018	resolver(nom + " " + ape)
019	else
020	rechazar();
021	}
022)
023	.then((uno)=>{



7.- Utilización de mecanismos de comunicación asíncrona (AJAX)-Asynchronous Javascript and XML) Promise.



024	<pre>document.getElementById("resultado").value=uno;</pre>
025	1)
026	.catch(()=>{
027	document.getElementById("resultado").value="Al menos uno de los campos
	está vacío";
028	1);
029	}

promise134.js

prom	156154.15
001	<pre>if (document.addEventListener)</pre>
002	<pre>window.addEventListener("load",inicio)</pre>
003	else if (document.attachEvent)
004	<pre>window.attachEvent("onload",inicio);</pre>
005	function inicio() {
006	<pre>let boton=document.getElementById("poner");</pre>
007	<pre>if (document.addEventListener)</pre>
800	boton.addEventListener("click",crear)
009	else if (document.attachEvent)
010	<pre>boton.attachEvent("onclick",crear);</pre>
011	}
012	<pre>function crear(){</pre>
013	<pre>let mipromesa= new Promise(</pre>
014	(resolver, rechazar) => {
015	<pre>let nom=document.getElementById("nombre").value.trim();</pre>
016	<pre>let ape=document.getElementById("apellidos").value.trim();</pre>
017	if (nom!="" && ape!="")
018	resolver(nom + " " + ape)
019	else
020	rechazar();
021	1
022)
023	.then(uno =>{
024	<pre>document.getElementById("resultado").value=uno;</pre>
025	1)
026	.catch(()=>{
027	<pre>document.getElementById("resultado").value="Al menos uno de los campos</pre>
	está vacío";
028));
029	}

