

A0 - Galeria d'imatges

Per fer aquest exercici crea un document on expliquis el procés de desenvolupament. A més, puja el codi font al repositori GitHub.

Prepara un servidor apache amb un virtualhost anomenat `gallery.site_[elteunom]`

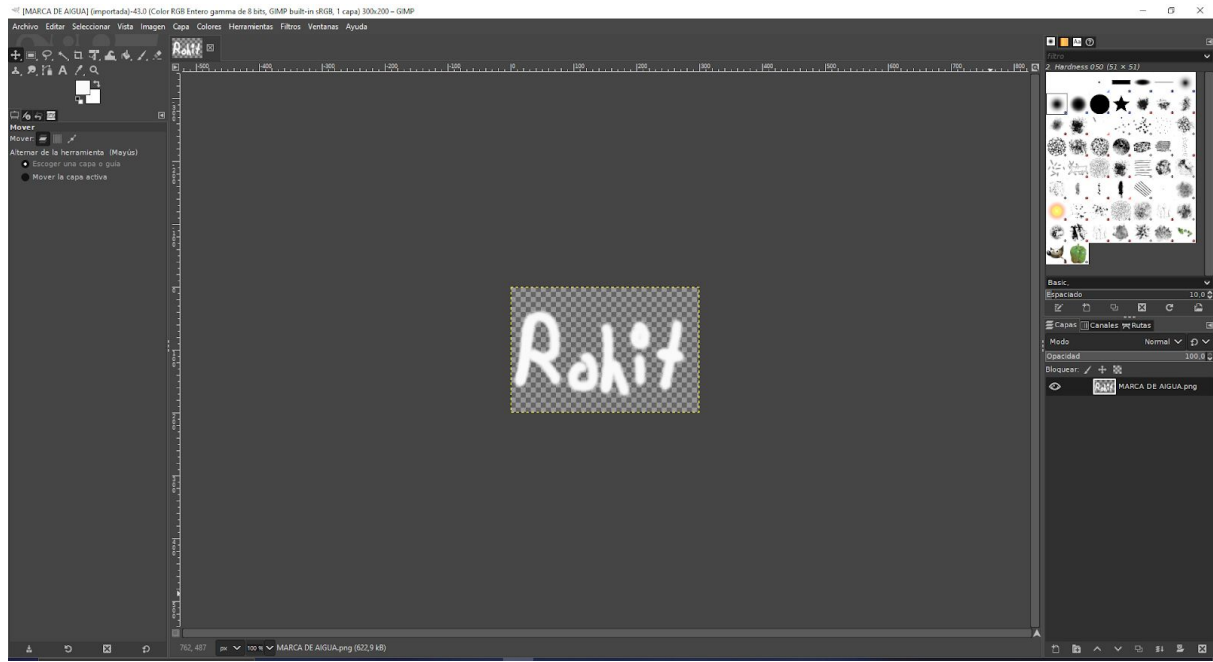
A dins crea una carpeta `image` i, a dins d'aquesta una carpeta `originals`

Descarrega les imatges que trobaràs a la carpeta `recursosA0` i guarda-les a la carpeta `originals`.

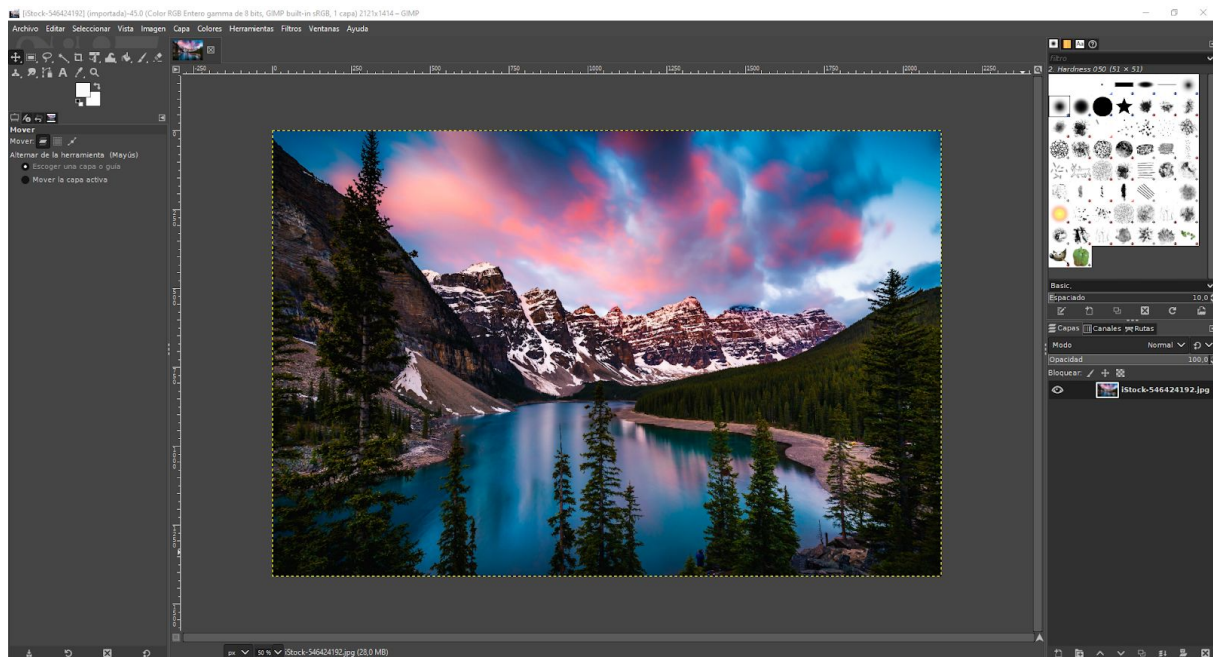
Afegeix una marca d'aigua amb transparència amb el teu nom a sobre dels originals. Utilitza les eines de l'editor gimp per crear-la manualment.

EXERCICI MARCA D'AIGUA:

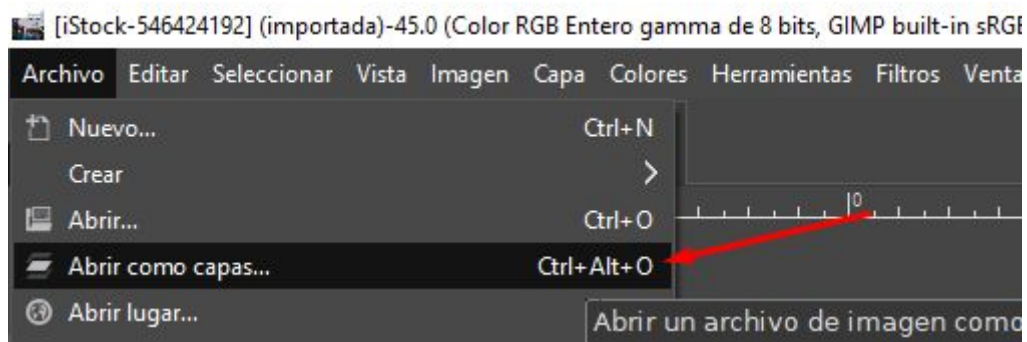
Primer de tot he creat la marca d'aigua amb el fons transparent.



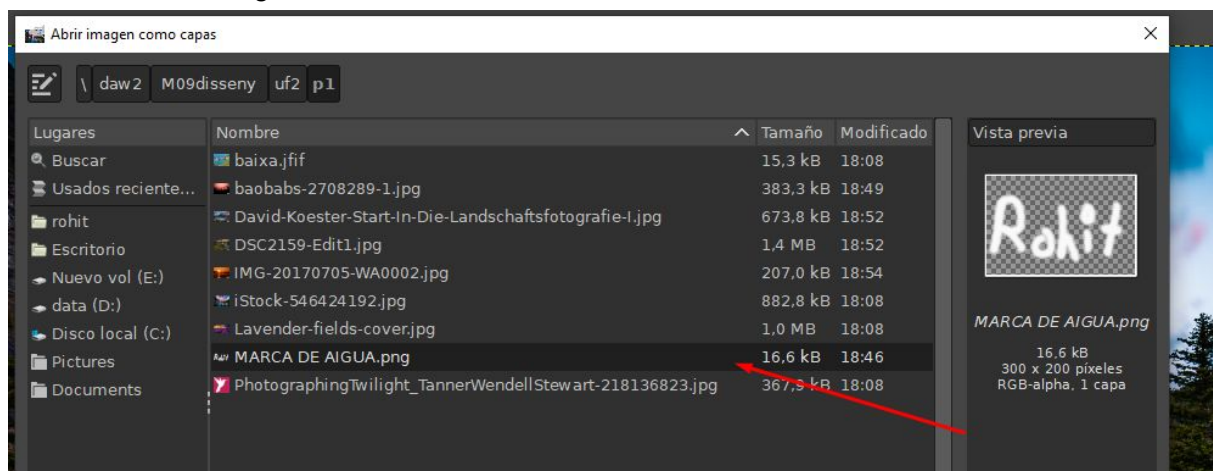
Obrim la imatge que vulguem afegir la marca d'aigua:



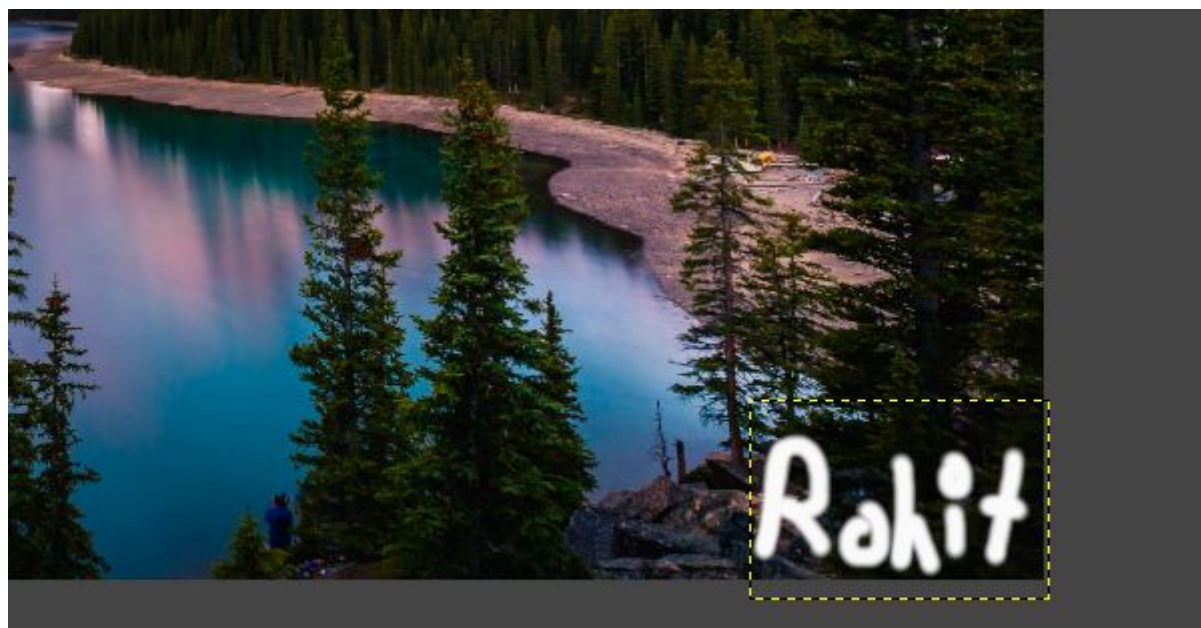
Obrim amb capa:



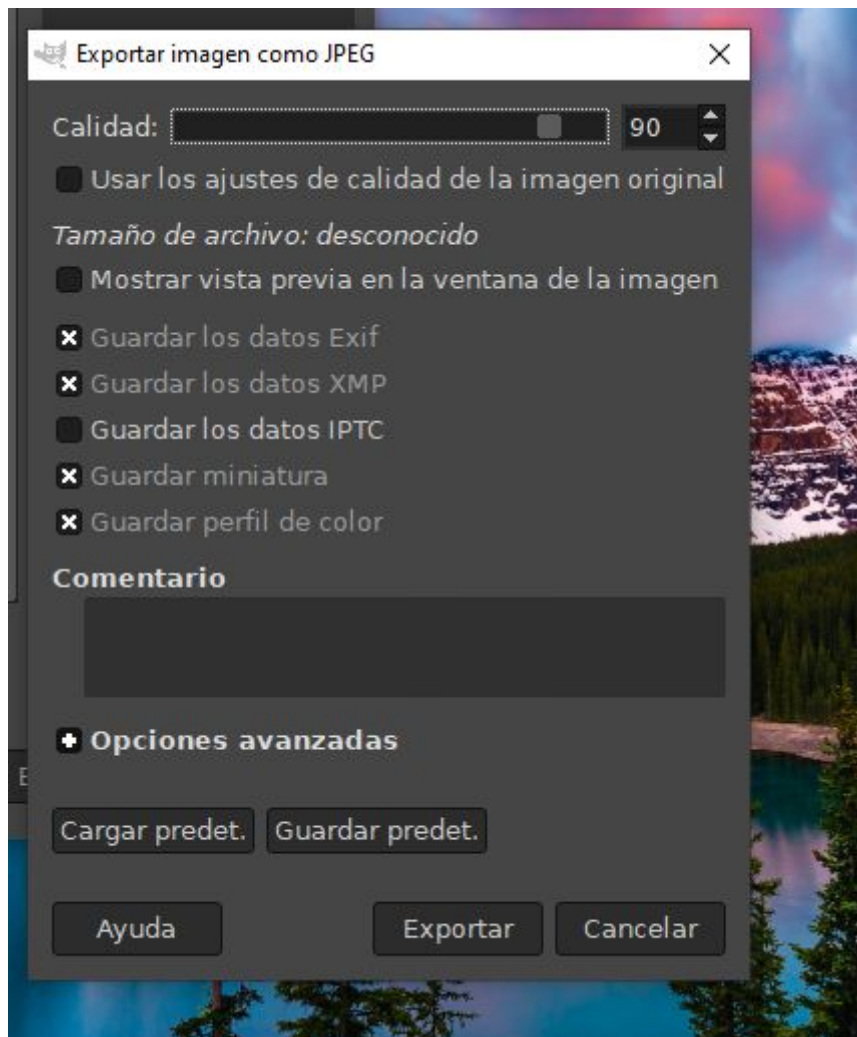
Triem la marca d'aigua:



Posem la marca d'aigua on vulguem.



I per últim exportem la imatge.

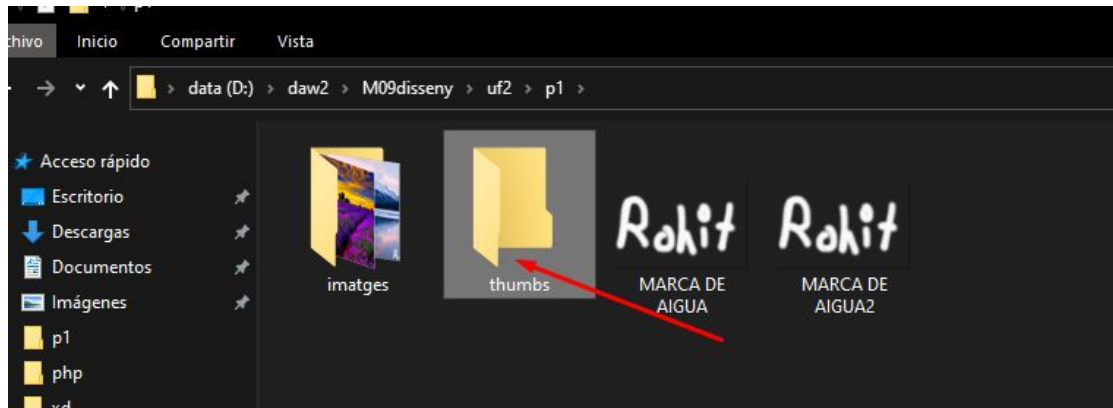


I això ho fem amb totes les imatges.

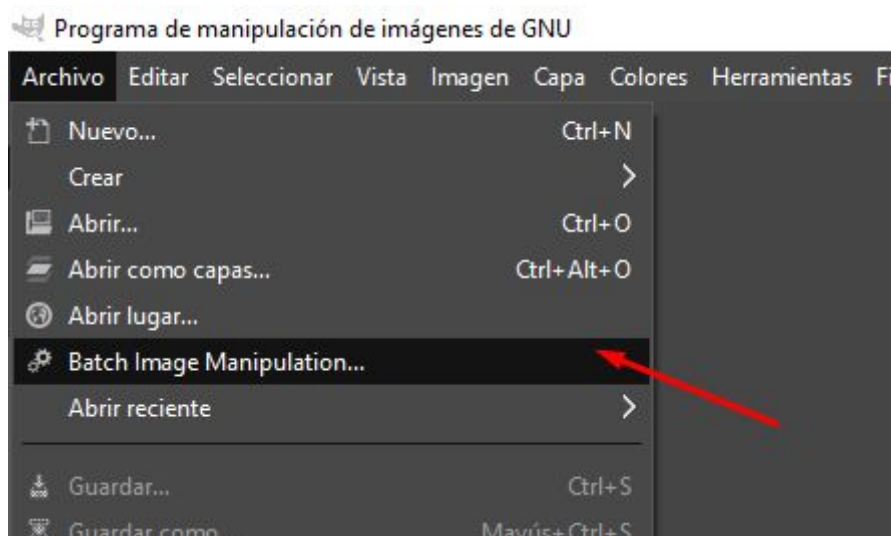
A /image crea una carpeta thumbs on guardis una còpia de les imatges amb la mida reduïda a 200x200 i resolució de 72ppi. En retallar, quedat amb la cantonada superior esquerra de l'original.

EXERCICI THUMBS

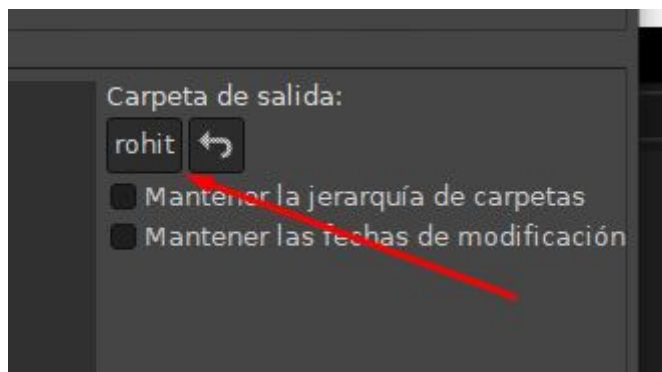
Primer de tot he creat la carpeta thumbs.



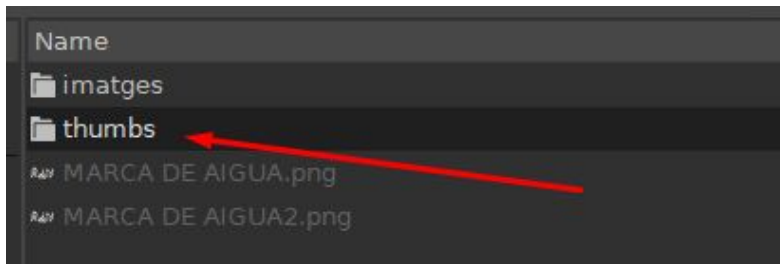
Un cop tinguem el bimp instal·lat l'obrim de la següent forma.



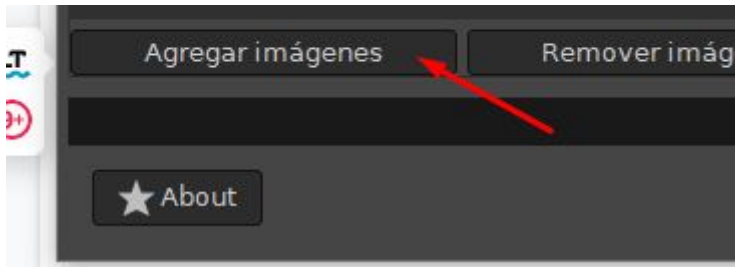
Fem clic on posa carpeta de sortida.



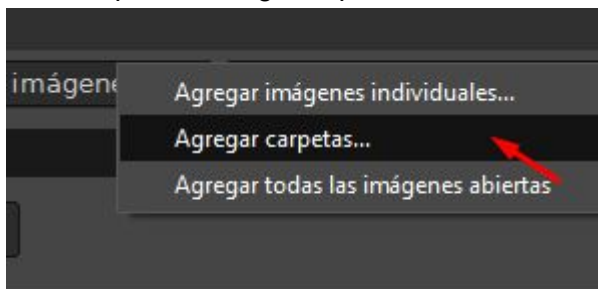
I triem la carpeta de destí.



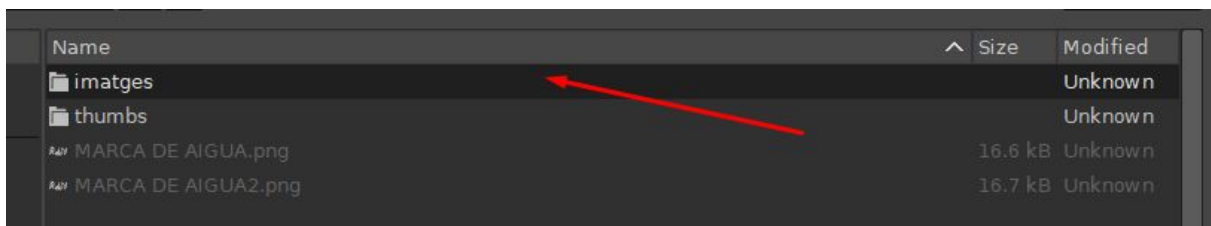
A continuació fem clic en “afegir imatges”.



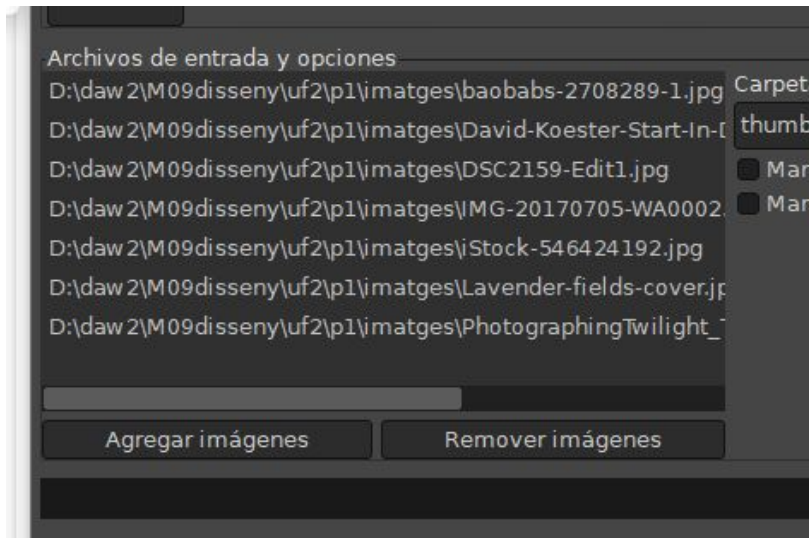
Triem l'opció d' “afegir carpetes”.



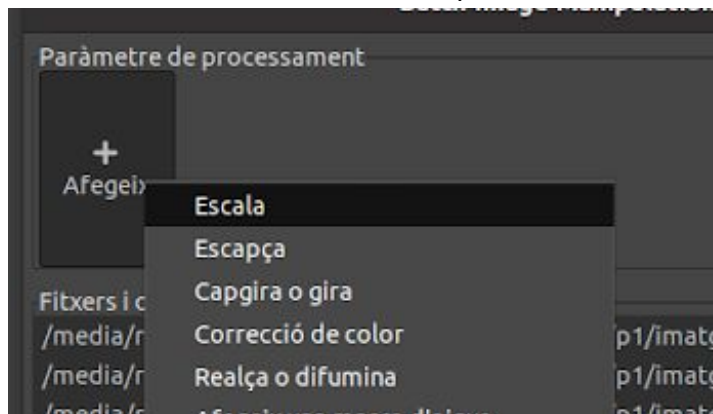
I triem la carpeta en la qual estan totes les imatges que volem modificar.



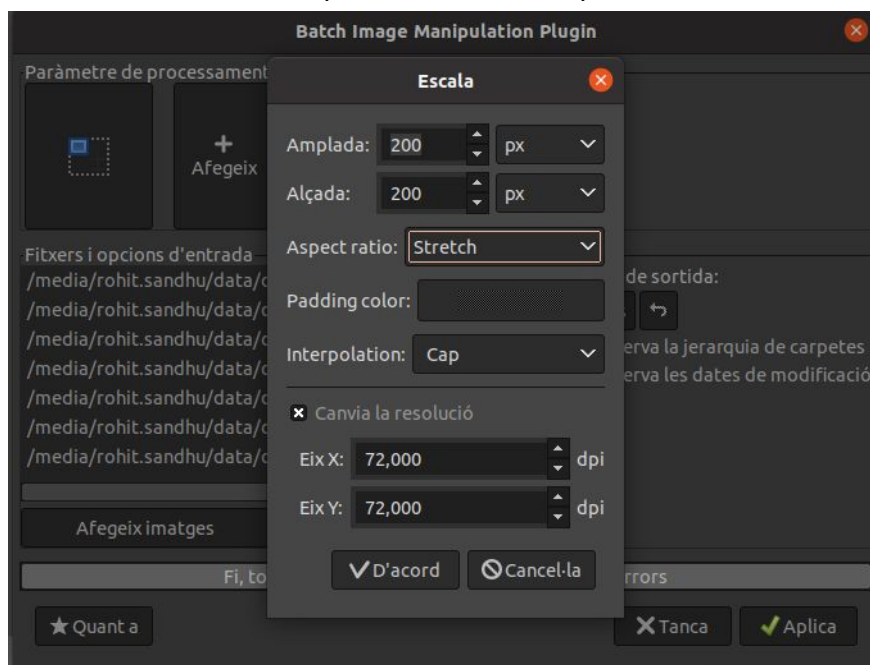
Com podem veure s'han carregat totes imatges de la carpeta "imatges".



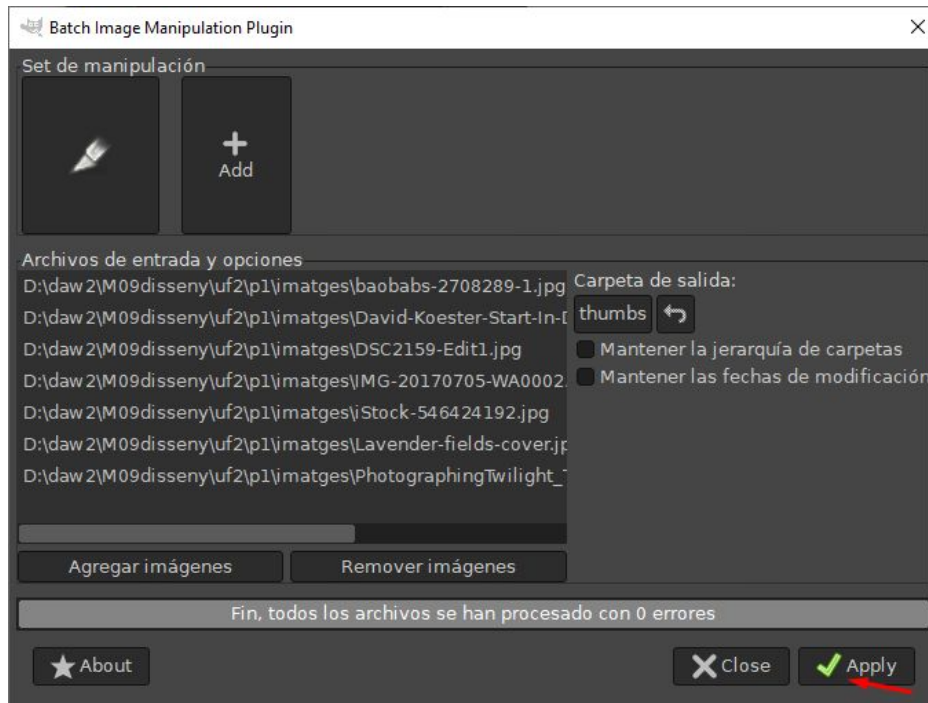
Fem clic en l'icona de "+" i triem l'opció "Escalar".



A continuació triem les opcions dels canvis que volem fer.

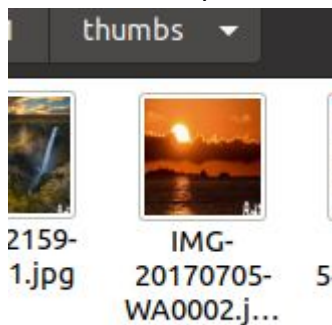


Per últim fem clic en “Apply”.



Comprovació:

Anem en la carpeta “thumbs” i fem clic en una de les imatges.



Com podem veure a la imatge s'han fet els canvis i només es veu 200x200 la part superior esquerra.



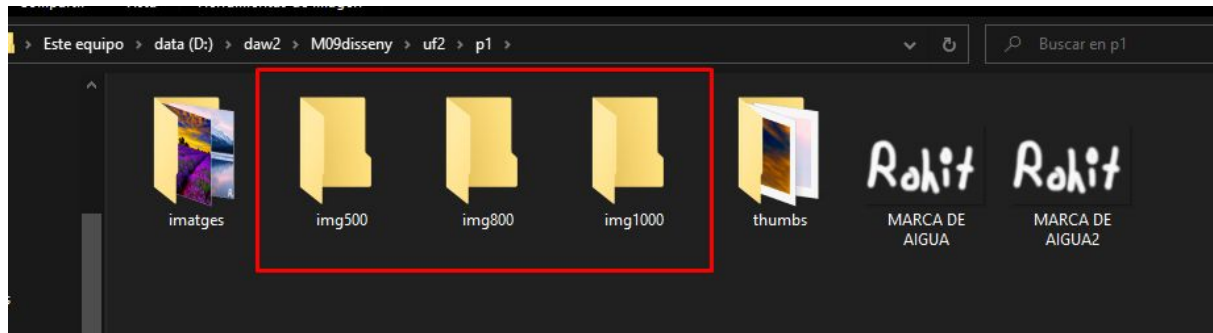
O també podem anar a les propietats de la imatge i ho podem veure.

Bàsic	Permisos	Obre
Tipus d'imatge	jpeg (JPEG)	
Amplada	200 píxels	
Alçada	200 píxels	
Programari	GIMP 2.10.18	
Creat el	2021:02:01 18:48:29	

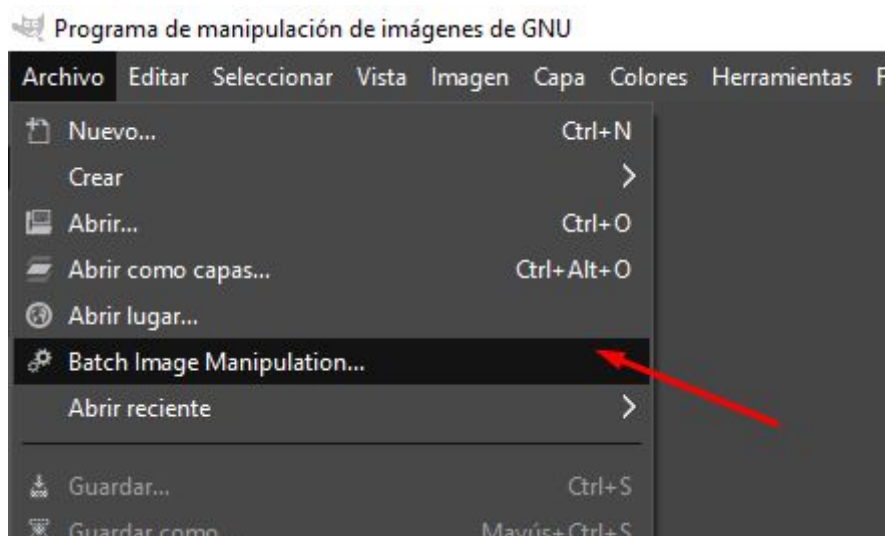
A /image crea tres carpetes img500, img800 i im1000 amb les imatges reduïdes a 500px, 800px i 1000px d'amplada i 72ppi, sempre respectant l'aspect ratio original.

EXERCICI 3 CARPETES

IMG500



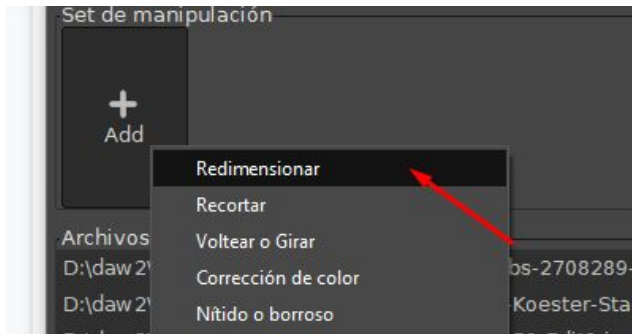
Obrim les opcions del bimp de la següent forma.



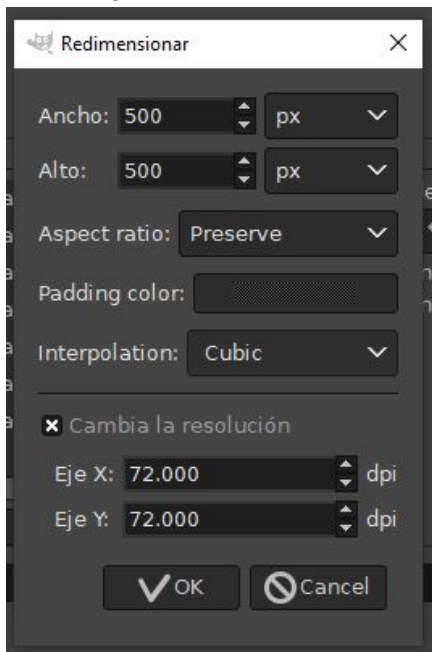
Triem la carpeta de destí.



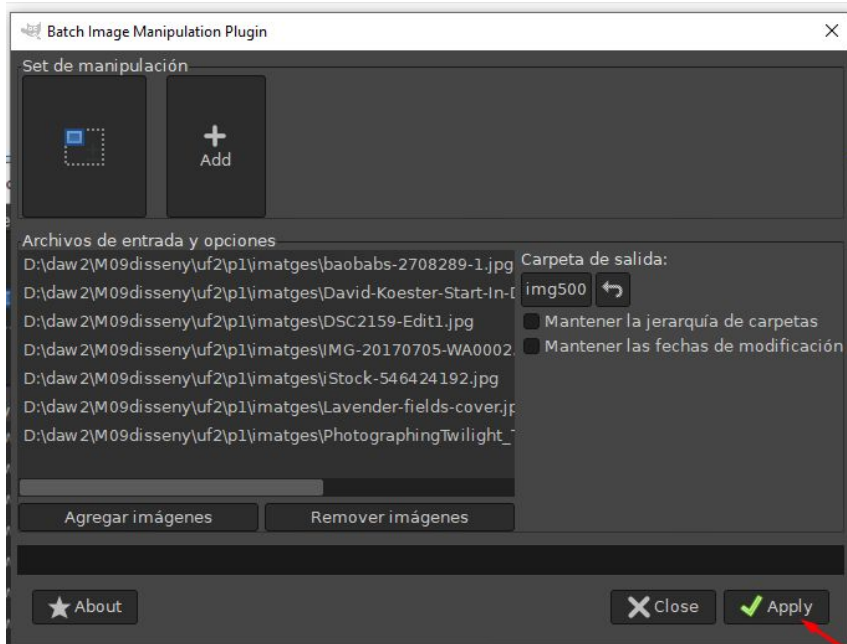
A continuació fem clic en icona amb un “+” i triem l’opció redimensionar.



En la següent finestra triem tots els canvis que volem fer.

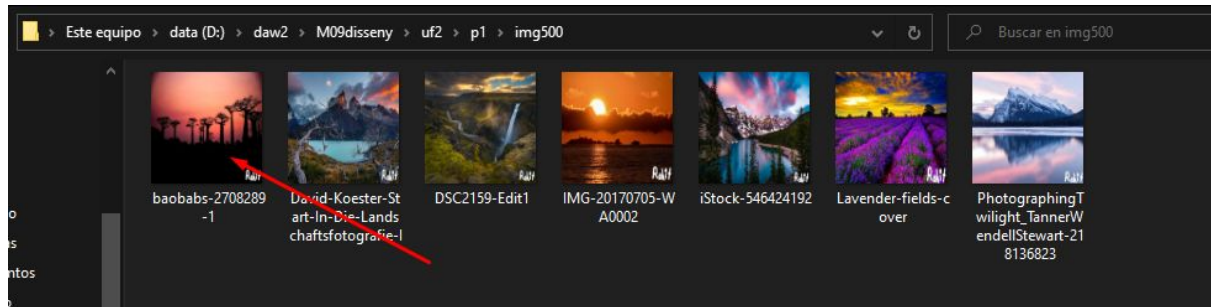


I per últim fem clic en el botó de “Apply”.



Comprovació:

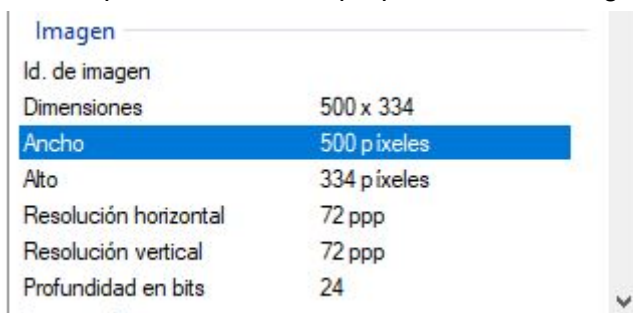
Per fer la comprovació anem a la carpeta de destí que era la “img500”.



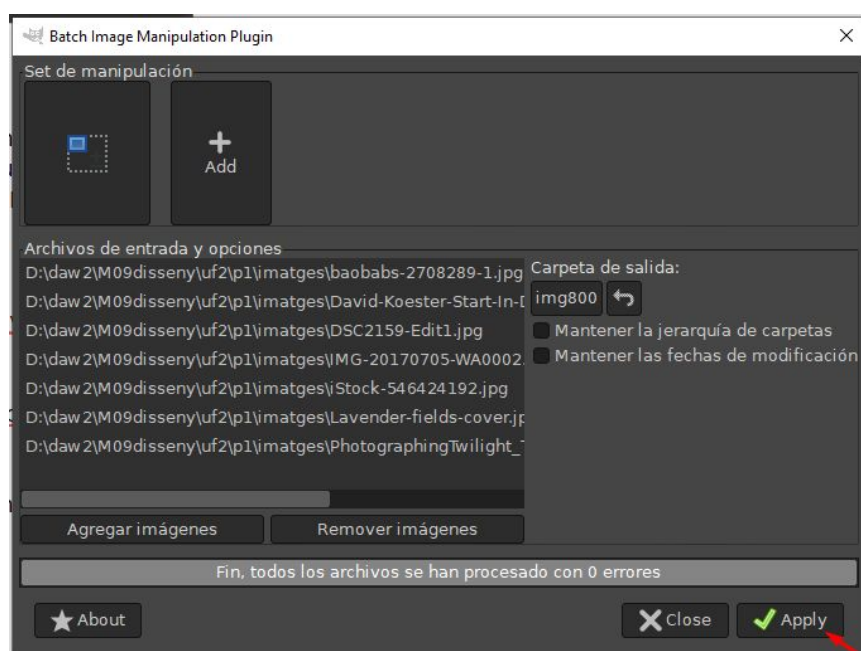
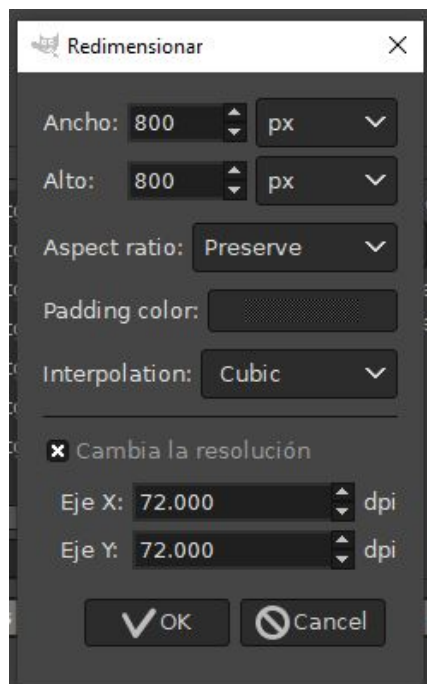
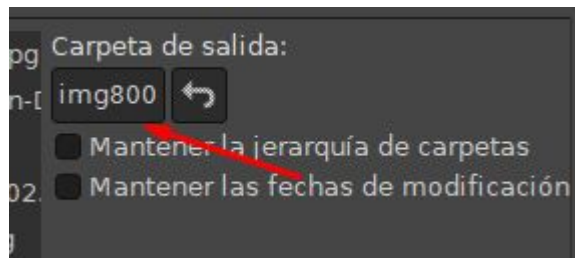
Fem clic en una imatge per veure s'ha redimensionat.



O també podem anar a les propietats de la imatge per veure si s'han fet els canvis.

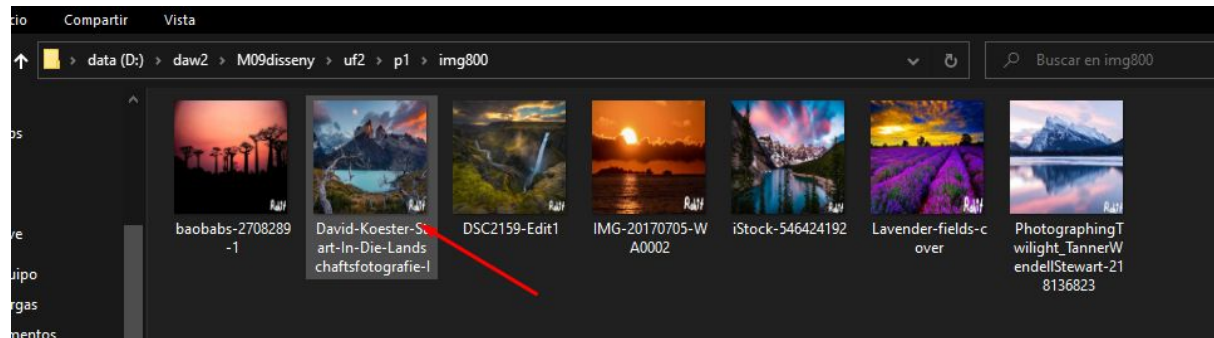


IMG800

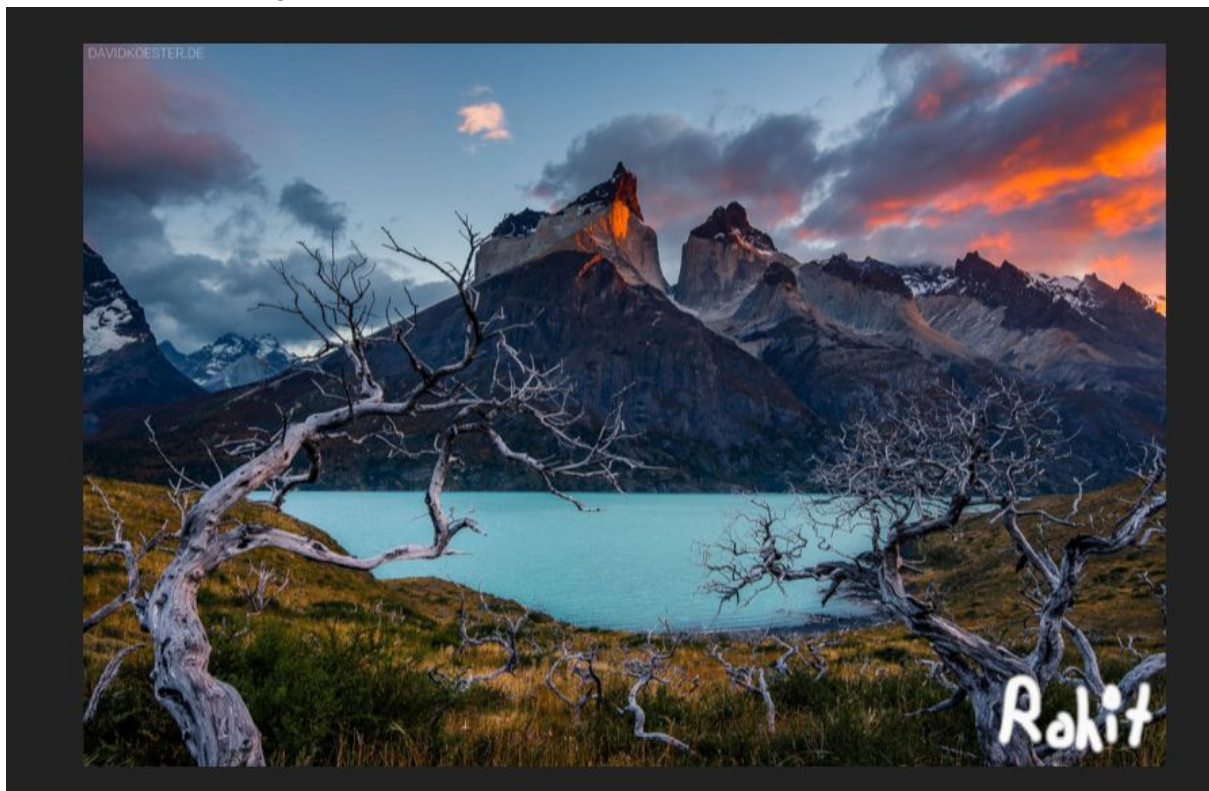


COMPROVACIÓ:

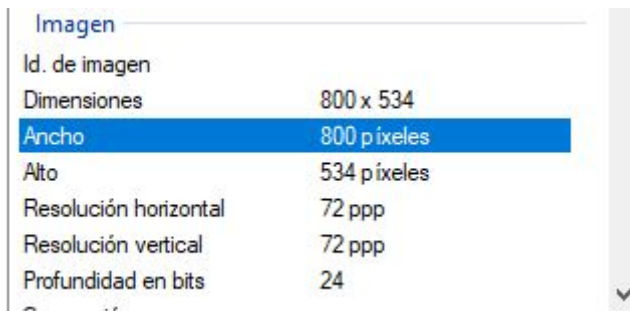
Per fer la comprovació anem a la carpeta de destí que era la “img800”.



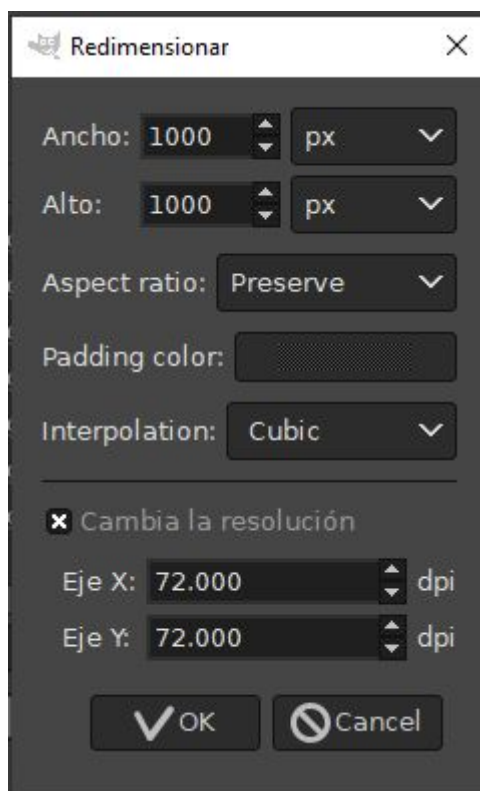
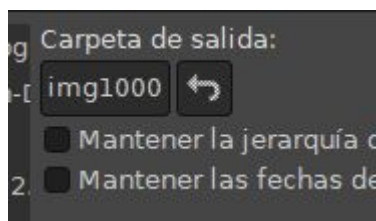
Fem clic en una imatge per veure s'ha redimensionat.

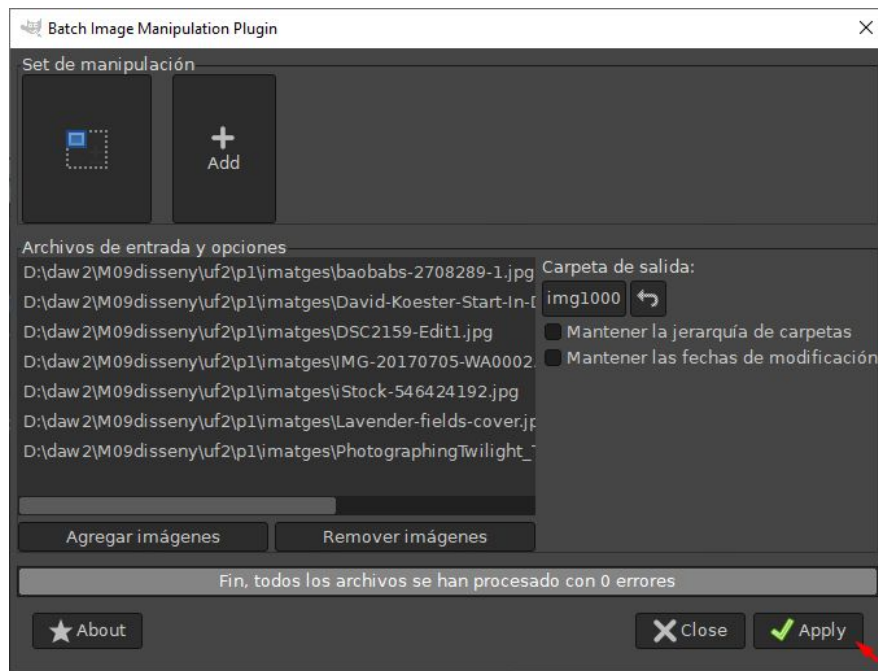


O també podem anar a les propietats de la imatge per veure si s'han fet els canvis.



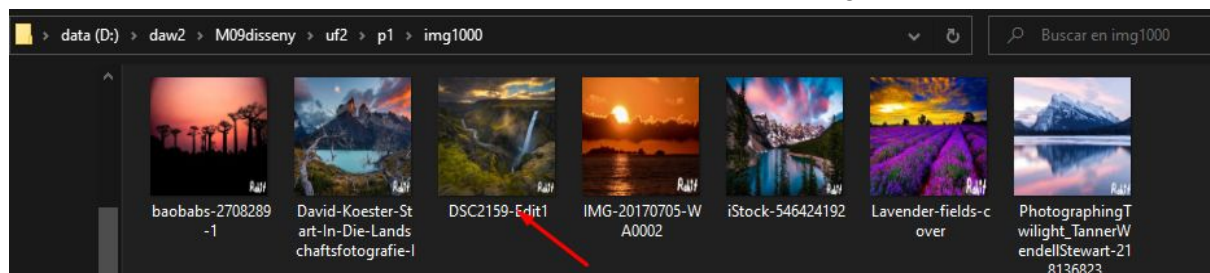
IMG1000





COMPROVACIÓ:

Per fer la comprovació anem a la carpeta de destí que era la “img1000”.



Fem clic en una imatge per veure s'ha redimensionat.



O també podem anar a les propietats de la imatge per veure si s'han fet els canvis.

Imagen	
Id. de imagen	
Dimensiones	1000 x 715
Ancho	1000 píxeles
Alto	715 píxeles
Resolución horizontal	72 ppp
Resolución vertical	72 ppp
Profundidad en bits	24

Crea un document index.html amb una galeria d'imatges on apareguin 8 caixes de 200px x 200px una al costat de l'altra utilitzant una llista, amb un gap de 20px i una vora negra de 2px. Utilitza display:flex.

Posa-hi les 8 imatges que has descarregat amb la mida original

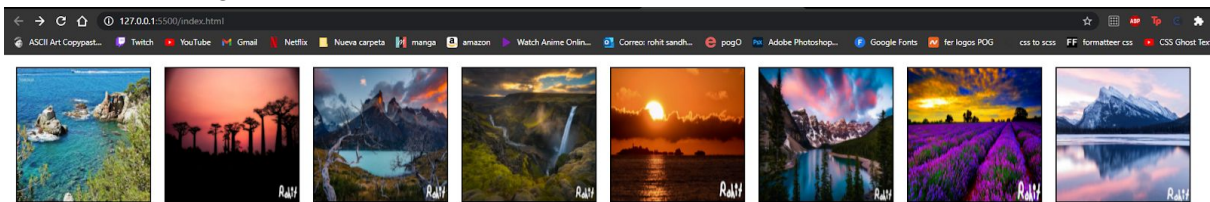
Fent hover sobre les imatges ha de fer algun efecte visible (transparencia, color del border, etc...)

Fent click a les imatges s'ha de mostrar la imatge a mida real.

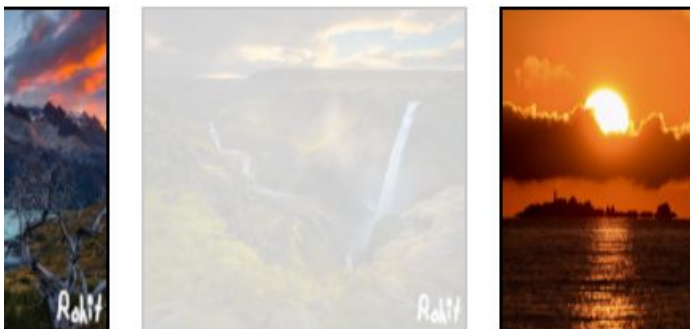
Amb l'inspector del Chrome localitza la mida total de la pàgina.

ÚLTIM EXERCICI:

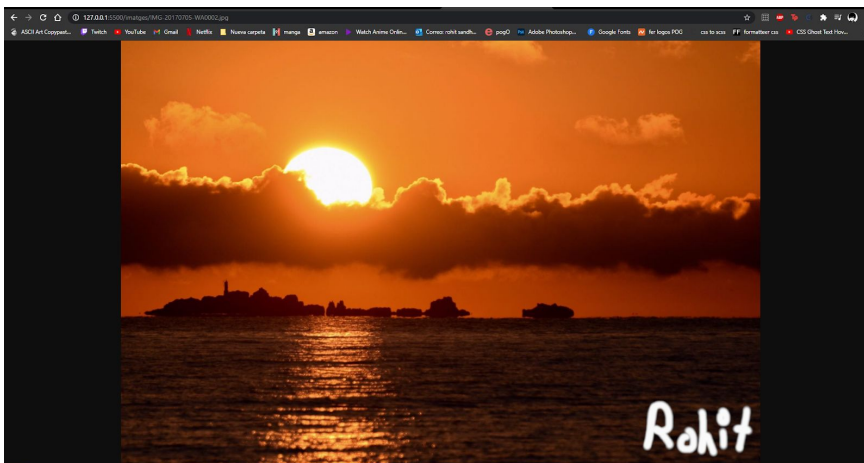
Així es veu la pàgina.



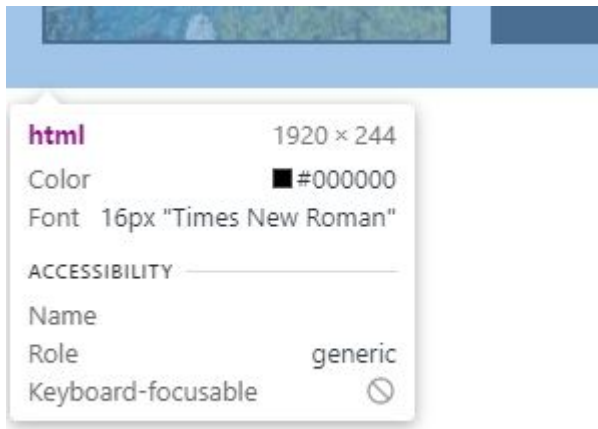
On hover es veu així:



Si cliques en la imatge t'obre la imatge amb la seva mida original.



Això és la mida de la pàgina:



CODI:

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width,
initial-scale=1.0">
  <title>Rohit page</title>
</head>

<body>

  <style>
    .divFlex {
      display: flex;
    }

    .img200 {
      width: 200px;
      height: 200px;
      margin: 10px;
      border: 2px solid black;
    }

    .img200:hover {
      opacity: 0.2;
    }
  </style>
```

```
<div class="divFlex">

    <a href="imatges/baixa.jfif">
        
    </a>

    <a href="imatges/baobabs-2708289-1.jpg ">
        
    </a>

    <a
href="imatges/David-Koester-Start-In-Die-Landschaftsfotografie-I.jpg ">
        
    </a>

    <a href="imatges/DSC2159-Edit1.jpg ">
        
    </a>

    <a href="imatges/IMG-20170705-WA0002.jpg ">
        
    </a>

    <a href="imatges/iStock-546424192.jpg ">
        
    </a>
```

```
        <a href="imatges/Lavender-fields-cover.jpg ">
            
        </a>

        <a
href="imatges/PhotographingTwilight_TannerWendellStewart-218136823.jpg
">
            
        </a>

    </div>

</body>

</html>
```

A1: Carregant la imatge correcta

Per controlar ràpidament quina imatge s'està mostrant cal que editeu i marqueu una d'elles amb el text 500 / 800 / 1000 (en cada resolució)

Creem imatges amb les marques d'aigua.



Per la part de les miniatures, substitueix les imatges per les que has reduït a la carpeta thumbs.

```
<a href="thumbs/baobabs-2708289-1.jpg ">
  
</a>

<a href="thumbs/David-Koester-Start-In-Die-Landschaftsfotografie-I.jpg ">
  
</a>

<a href="thumbs/DSC2159-Edit1.jpg ">
  
</a>

<a href="thumbs/IMG-20170705-WA0002.jpg ">
  
</a>

<a href="thumbs/iStock-546424192.jpg ">
  
</a>

<a href="thumbs/Lavender-fields-cover.jpg ">
  
</a>

<a href="thumbs/PhotographingTwilight_TannerWendellStewart-218136823.jpg ">
  
</a>
</div>
```

A sota de les miniatures crea un div (id="detail") centrat que faci al 80vw de l'amplada del navegador i 50vw de l'alçada.

Creem el div amb les mides corresponents:

```
<div id="detail" style="width: 80vw; height: 50vw; margin: auto;">

</div>
```

Amb css no tenim manera de seleccionar quina imatge hem de descarregar si fem servir . Si volem que es carregui la imatge òptima amb CSS haurem de posar la imatge com a background-image.

Ara, crea media querys per tal que es carregui la imatge òptima aprofitant les imatges (500, 800 i 1000) que has generat. Per poder especificar la imatge amb css hauràs de posar-la com a imatge de fons amb

```
background-size: cover;
background-position: center;
```

```
@media (max-width: 500px) {
  #detail {
    background-size: cover;
    background-position: center;
    background-image: url('imatgesNoves/500.jpg');
  }
}

@media (max-width: 1000px) {
  #detail {
    background-size: cover;
    background-position: center;
    background-image: url('imatgesNoves/800.jpg');
  }
}

@media (min-width: 1000px) {
  #detail {
    background-size: cover;
    background-position: center;
    background-image: url('imatgesNoves/1000.jpg');
  }
}
```


**Observa com la imatge queda retallada en funció de la seva proporció.
Veient que utilitzar background-image no és la millor opció, anem a treballar amb una altra etiqueta d'html5: <picture>**

Sí, queda retallada.



Copia el document page.html a index.html amb un full d'estils diferent i elimina les propietats que has afegit per background-image i utilitza les propietats source i media de picture per aconseguir que es carregui la imatge més adient

```
<picture>
  <source media="(min-width:1000px)" srcset="images/img1000/imatge.jpg">
  <source media="(min-width:800px)" srcset="images/img800/imatge.jpg">
  x
</picture>
```

```
<picture>
  <source media="(min-width:1000px)" srcset="imatgesNoves/1000.jpg">
  <source media="(min-width:800px)" srcset="imatgesNoves/800.jpg">
  x
</picture>
```

Resultat:

Més gran que 1000px:



Més gran que 800px:



La resta:



A2: JQuery

Aquest exercici l'has de pujar a una nova branca al github. s.

Substitueix l'element <picture> per un .

```
<img id="main-image" src="" style="display:none">
```

Incorpora la llibreria jQuery a index.html. Crea i incorpora un fitxer js/daw_gallery.js on escriuràs el codi javascript.

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
```

```
<script src="js/daw_gallery.js"></script>
```

Escriu amb jQuery (no CSS) un programa que respongui a :hover afegint una classe "thumb-rollover" provocant un petit escalat del thumb (a 1.1). En sortir de :hover l'efecte ha de desaparèixer. Investiga transform: scale(1.1); i transition: all .4s ease-in-out; per fer l'efecte més suau.

```
$( "img" ).hover(
  function() {
    $(this).addClass("thumb-rollover");
  },
  function() {
    $(this).removeClass("thumb-rollover");
  }
);
```

Escriu un programa amb jQuery que respongui al click sobre les miniatures canviï el border a vermell (només el darrer click, les altres ha de tenir border negre).

```
6
7
8  $( "img" ).click(function() {
9
10     $( "img" ).css( "border", "2px solid black" );
11     $(this).css( "border", "2px solid red" );
12
13  });
```

A sobre de la caixa #detail afegeix una caixa #image-title

Escriu un programa amb jQuery que respongui al click sobre les miniatures i posi a #image-title el nom del fitxer sense ruta i sense extensió (investiga amb string.split('/')).

```
$("#img").click(function(){
    var nom_img = $(this).attr("src").split("/");
    console.log(nom_img[1]);
    nom_img = nom_img[1].split(".");
    $("#image-title").text( nom_img[0])
})
```

Finalment, aprofita jQuery().width(); per conèixer l'amplada del div #detail, substitueix l'atribut src per la ruta correcta al fitxer d'imatge òptim. generant la ruta cap a img500, img800 o img1000. També amb jQuery, elimina el "display:none".

A3: Modal

A continuació cal que creis una classe css anomenda modal:

```
.modal {
    /* display: none; Hidden by default */
    position: fixed; /* Stay in place */
    z-index: 1; /* Sit on top */
    left: 0;
    top: 0;
    width: 100%; /* Full width */
    height: 100%; /* Full height */
    overflow: auto; /* Enable scroll if needed */
    background-color: rgb(0,0,0); /* Fallback color */
    background-color: rgba(0,0,0,0.4); /* Black w/ opacity */
}

/* Modal Content/Box */
.modal img {
    background-color: #fefefe;
    margin-top: 20%;
    padding: 20px;
    border: 1px solid #888;
    width: 80%; /* Could be more or less, depending on screen size */
}

.center {
    display: block;
    margin-left: auto;
```



```

margin-right: auto;
background-color: blue;
width: 80%
}

#thumbs li img.selected{
border: solid 2px red;
}

```

Cal que utilitzeu jQuery per afegir la classe .modal (jQuery().addClass()) quan es faci click en una miniatura, de manera que la imatge aparegui en una caixa flotant.

```

$(".img200").click(function() {
    var nom_img = $(this).attr("src").split("/");
    console.log(nom_img[1]);
    nom_img = nom_img[1].split(".");
    $("#image-title").text(nom_img[0]);

    $("#main-image").attr("src", "imatges/" + nom_img[0] + "." + nom_img[1])
    $('#main-image').css('display', 'block');

    $("#detail").addClass("modal");
    $("#modal-controller").removeClass("amagar");
})

```

A dins de detail, crea un div id="modal-controller" amb uns enllaços Prev, Next i Close que no portin enlloc (href="")

```

<div id="modal-controller" class="amagar">
    <a class="prev">PREV</a>
    <a class="next">NEXT</a>
    <a class="close">CLOSE</a>
</div>
</div>

```

Amb jQuery respon al click sobre "Close" fent que la caixa detail s'amagui i la imatge seleccionada perdi el border vermell (removeClass())

```

$(".close").click(function() {
    $("#modal-controller").addClass("amagar");
    $("#detail").removeClass("modal");
    $("#main-image").css("display", "none");
})

```

Programa el comportament de prev i next fent que resulti rotatiu (després del darrer element ha mostrar-se el primer).

Algunes funcions que et poden ser útils:

- parent()
- siblings()

- next() / prev()
- first() / last()
- trigger();

```

$(".prev").click(function() {

    if ($("#selected").prev().index() == -1) {
        console.log("xd entra en el if");
        let srcImg = $(".img200").last().attr("src").split("/");
        $(".img200").first().removeClass("selected");
        $(".img200").last().addClass("selected");

        console.log(srcImg)
        $("#main-image").attr("src", "img1000" + "/" + srcImg[1]);
    } else {
        let srcImg = $(".selected").prev().attr("src").split("/");

        console.log("ha entrat en el segon if");

        $(".selected").prev().addClass("tempClass");
        $(".selected").removeClass("selected");
        $("#main-image").attr("src", "img1000" + "/" + srcImg[1]);
        $(".tempClass").addClass("selected");
        $(".selected").removeClass("tempClass")
        console.log(srcImg);
    }
})

$(".next").click(function() {

    console.log($("#selected").next().index());

    if ($("#selected").next().index() == -1) {

        console.log("xd entra en el if dspfojds1ñfksdñlñfksdñlñk");

        let srcImg = $(".img200").first().attr("src").split("/");

        $(".img200").last().removeClass("selected");

        $(".img200").first().addClass("selected");

        console.log(srcImg)
        $("#main-image").attr("src", "img1000" + "/" + srcImg[1]);
    } else {
        let srcImg = $(".selected").next().attr("src").split("/");

        console.log("ha entrat en el segon if");

        $(".selected").next().addClass("tempClass");
        $(".selected").removeClass("selected");
        $("#main-image").attr("src", "img1000" + "/" + srcImg[1]);
        $(".tempClass").addClass("selected");
        $(".selected").removeClass("tempClass")
        console.log(srcImg);
    }
})

```