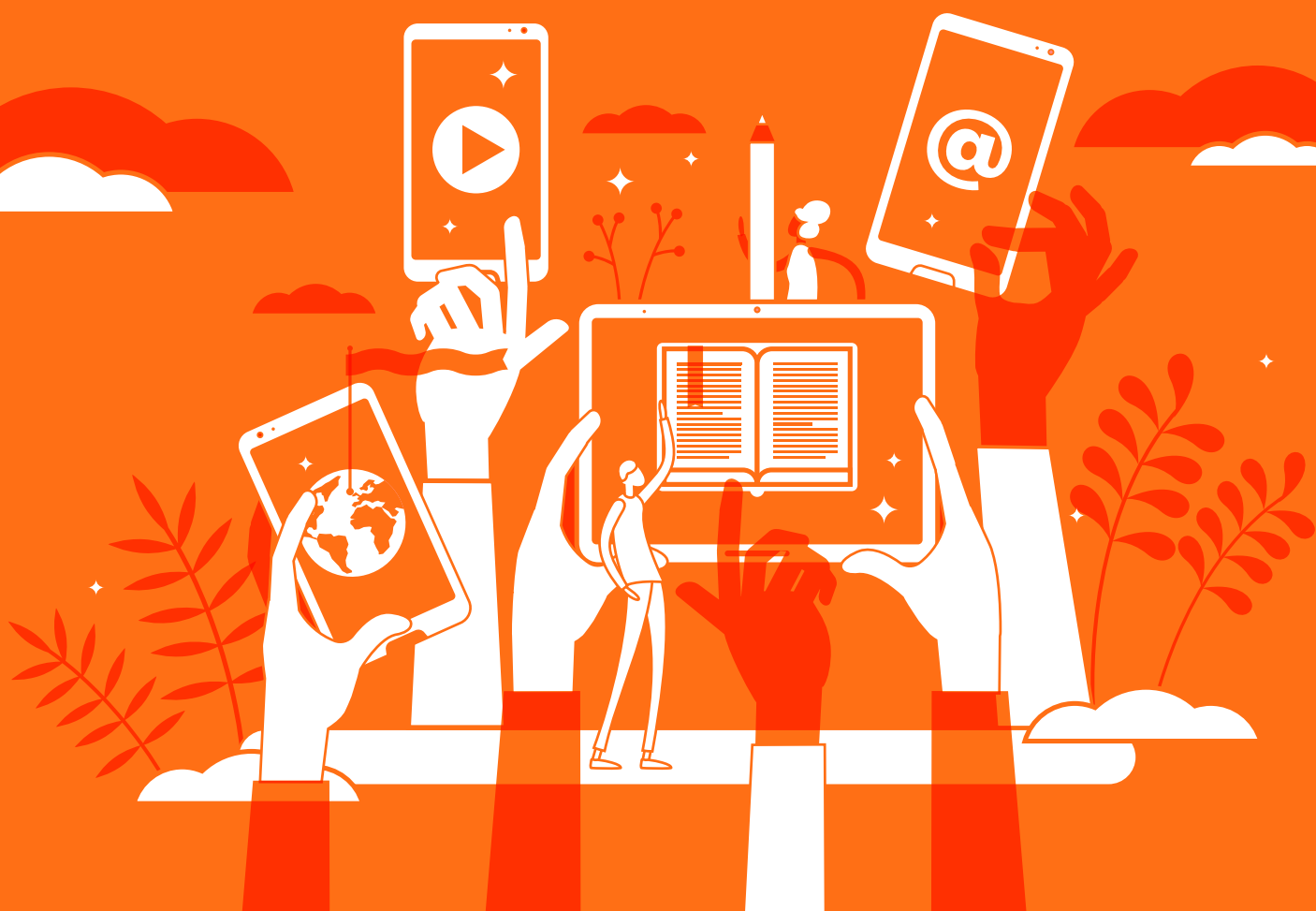




Gaitasun  
digitaletan  
prestakuntza

# 3

## Eduki digitalak sortzea





Gaitasun  
digitaletan  
prestakuntza



Eduki digitalak  
sortzea

# *B2maila*





## AURKIBIDEA

### 3.1 EDUKIAK GARATZEA

- *Testu-prozesadoreen erabilera aurreratua: bulegotikako beste tresna batzuekiko konexioa*
- *Testu-prozesadoreetako (libreak eta online) ohiko tresnak erabiltzea*
- *Testuari aplikatutako efektuak*
- *Irudi animatuak sortzea*
- *Diseinu grafikoko tresnak (libreak eta hodeian)*
- *Irudiak eta bideoa loteka konprimatzea*
- *Webguneak sortzeko aukerak*
- *WCAGeko egokitze-mailak*

### 3.2 EDUKI DIGITALA INTEGRATZEA ETA BIRLANTZEA

- *Aurkezpenetako trantsizioak eta animazio aurreratuak*
- *Bideoa eta audioa birmasterizatzea*

### 3.3 EGILE-ESKUBIDEAK ETA JABETZA INTELEKTUALEKO LIZENTZIAK

- *Obra bati copyrighta ematea*
- *Software librea, freewarea, sharewarea, adwarea*

### 3.4 PROGRAMAZIOA

- *Python lengoaiaren alderdi orokorrak*
- *Python-eko datu-motak*
- *Python-eko aldagaien eta konstanteen alderdi orokorrak*
- *Python-eko funtzioak eta parametro-pasatzeak*
- *Hiztegiak: definizioa eta erabilera nagusiak*
- *Python-eko fitxategien oinarrizko erabilera*
- *Zer da software 2.0?*



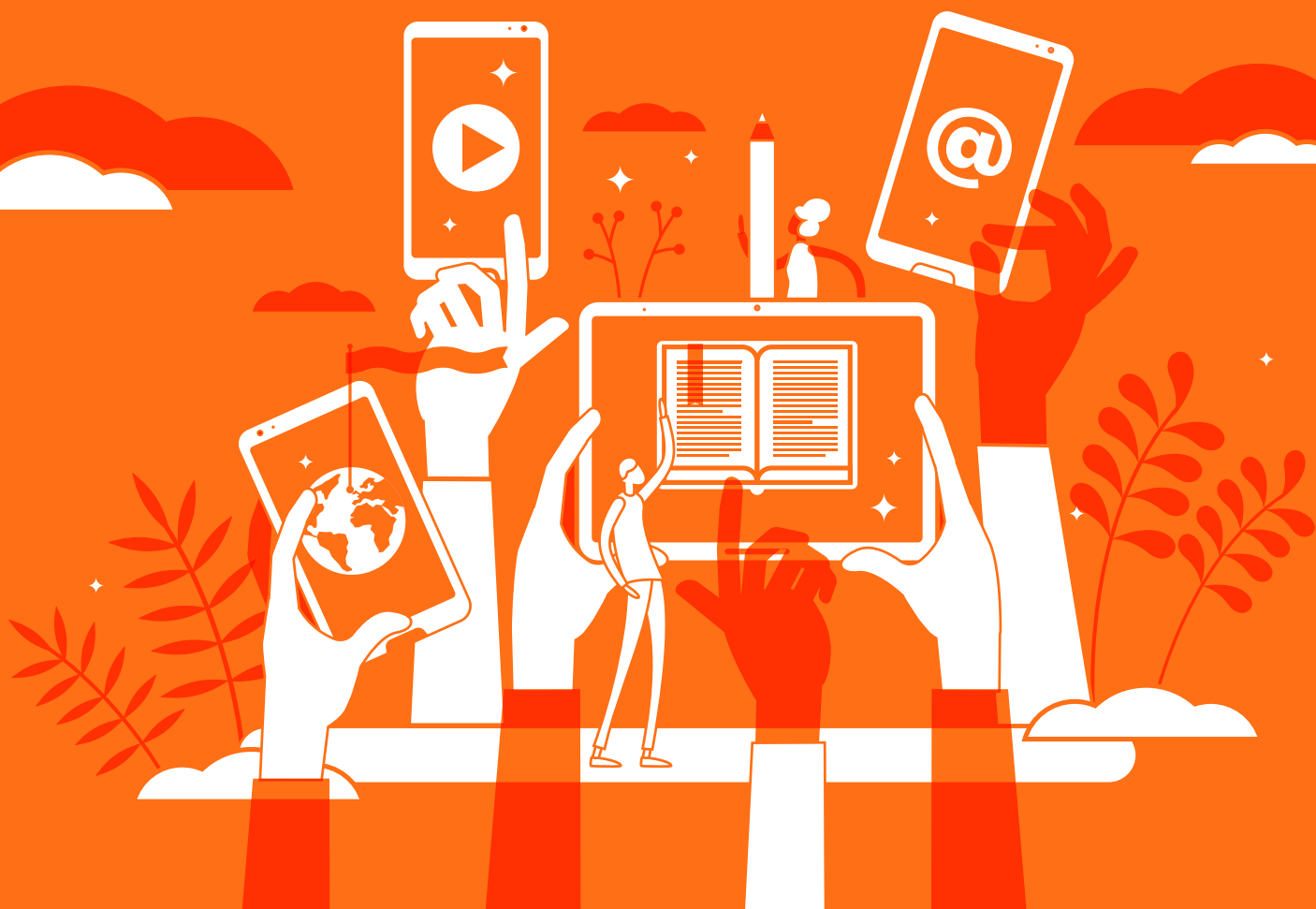


# DigitAll

Eduki digitalak  
sortzea

## 3.1

### EDUKIAK GARATZEA





Eduki digitalak  
sortzea

**B2 maila** 3.1 Edukiak  
garatzea

**Testu-  
prozesadoreen  
erabilera  
aurreratua:  
bulegotikako  
beste tresna  
batzuekiko  
konexioa**





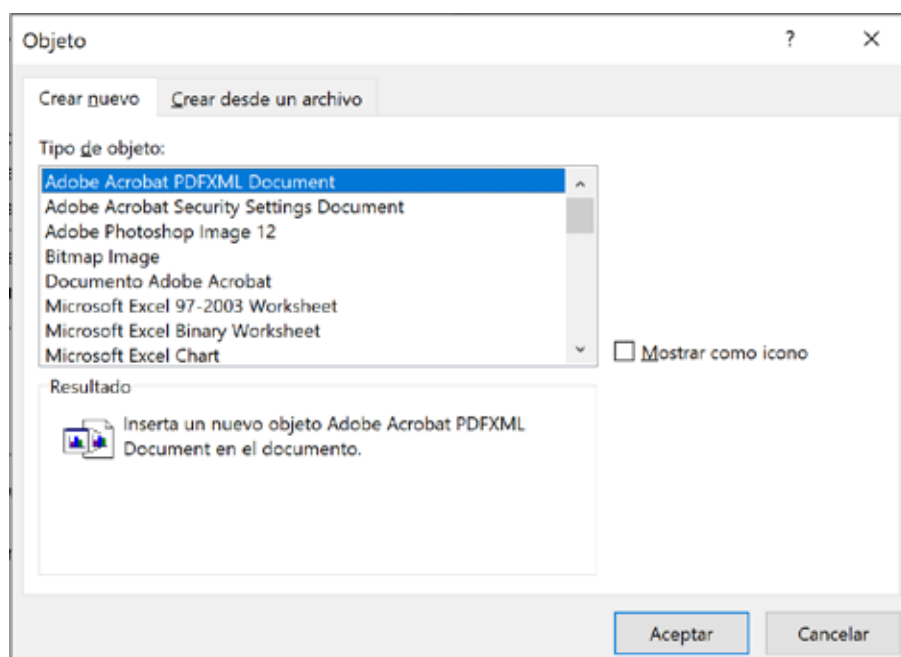
## Testu-prozesadoreen erabilera aurreratua: bulegotikako beste tresna batzuekiko konexioa

### Bulegotika-tresna konektagarriak

Testu-prozesadoreek (guk Microsoften Word programa erabiliko dugu adibide gisa) bulegotikako beste tresna batzuekin konektatzeko gaitasuna dute, laneko produktibitatea eta eraginkortasuna hobetzeko. Hona hemen zenbait aukera: kalkulu-orriak, aurkezpenak eta PDF fitxategiak.

### Beste tresna batzuk Wordekin estekatzeko urratsak

Fitxategiak konektatzea oso erraza da. «**Txertatu**» fitxan «**Objektua**» hautatu baino ez da egin behar.



Irekitzen den menu zabalgarrian, askotariko tresnak izango ditugu aukeran. Wordera gehitzeko aukera izango dugu, berriak sortuz («sortu berria») edo lehendik dagoen fitxategi batean oinarrituta sortuz. Objektuarekiko esteka sortu eta testu-dokumentuan txertatu ondoren, objektuaren gainean klik bikoitza eginez gero, aukera izango dugu txertatutako tresnaren utilitate guztiak erabiltzeko.



## Testu-prozesadoreko kalkulu-orria

Microsoft Exceleko edo Google Sheet-eko kalkulu-orri bat Worden konektatzeko, datuak honela «estekatu» daitezke:

- 1| Ireki elkarrekin estekatu nahi dituzun Word dokumentua eta Exceleko kalkulu-orria.
- 2| Hautatu Exceleko orrian estekatu nahi duzun gelaxka edo gelaxka-tartea.
- 3| Egin klik eskuineko botoiaz hautu beharrekoan eta hautatu «Kopiatu» aukera.
- 4| Itzuli Wordeko dokumentura eta jarri kurtsorea estekatutako datuak txertatu nahi dituzun lekuan.
- 5| Egin klik eskuineko botoiaz eta hautatu «Itsatsi (berezia)».
- 6| Agertzen den leihoan, hautatu «Itsatsi esteka» eta egin klik «Kalkulu-orria» aukeran.

Eta kito! Exceleko orriko datuak Word dokumentuari estekatu zaio.

Horri esker, kalkulu-orriko datuak zuzenean eguneratu daitezke Word dokumentutik, eta taulak, grafikoak txerta daitezke, baita kalkuluak eta analisiak behar dituzten bestelako elementuak ere.

**Utilitatea:** Irakaslea edo ikaslea bazara eta Wordeko aurkezpen bat zenbakizko datuak barne hartuta egin behar baduzu, datu horiek Exceleko kalkulu-orri bati estekatu ahal dizkiozu; horrela, Wordeko dokumentuan automatikoki eguneratuko dira Exceleko orrian aldaketak egitearekin batera.

## Testu-prozesadoreko PDF fitxategia

Jarraitu beharreko urratsak dokumentu honen 1.1 puntuan daude.

Adibide honetan egiaztatuko duzu zertarako den baliagarria PDF bat testu-dokumentu bati estekatzea:

- 1| Demagun txosten bat idazten ari zarela eta PDF fitxategi batean dauden grafiko edo irudi batzuk sartu behar dituzula. Irudi bakoitza kopiatu eta itsatsi beharrean, aski duzu PDF fitxategia zure Word dokumentuan txertatzea, eta irudi guztiak toki bakar batean izango dituzu eskuragarri.
- 2| Zure Word dokumentuan PDF fitxategi bat txertatzean, ziur egongo zara formatua ez dela aldatuko; izan ere, PDF fitxategia irudi finkoa da, eta fitxategi osoa jasotzen da.





## Aurkezpenak eranstea testu-prozesadorean

Azkenik, aurkezpenak egiteko tresnak, hala nola Microsoft PowerPoint, testu-prozesadoreekin ere konekta daitezke (adibidez, Wordekin) «Txertatu objektua» aukeraren bidez. Bertan, aurkezpen oso bat txerta daiteke Word dokumentu batean.

### Informazio gehiago

Word beste tresna batzuekin konektatzeak, hala nola Exceleko kalkulu-orriekin edo PowerPointeko aurkezpenekin, txosten osoak eta profesionalak sortzea errazten du. Erabiltzaileek datuak eta grafikoak zuzenean inporta ditzakete beste aplikazio batzuetatik, eta horrek arindu egiten du sortze-prozesua, eta aurkeztutako informazioaren zehaztasuna hobetzen du. Horrek denbora aurrezten du eta akatsak murrizten ditu. Informazio gehiago izateko, sartu webgune honetan:

[e.digitall.org.es/objetos-vinculados](http://e.digitall.org.es/objetos-vinculados)







Eduki digitalak  
sortzea

**B2 maila** 3.1 Edukiak  
garatzea

**Testu-  
prozesadoreetako  
(libreak eta  
online)  
ohiko tresnak  
erabiltzea**



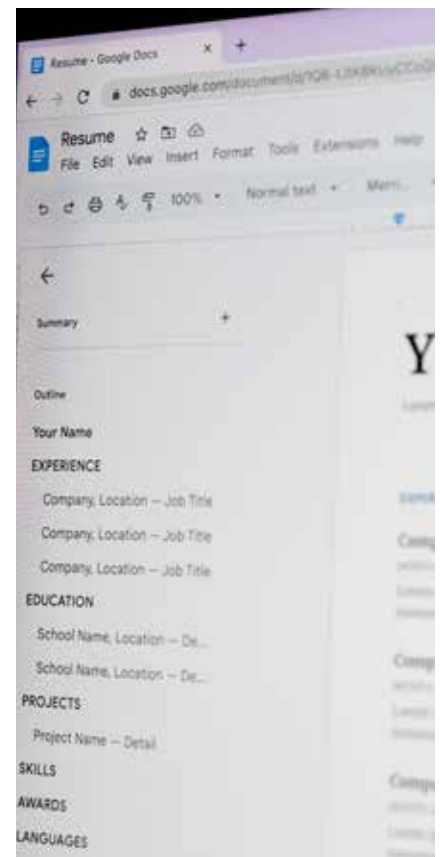


## Testu-prozesadoreetako (libreak eta online) ohiko tresnak erabiltzea

Testu-prozesadoreetako ohiko tresnetako asko testu-prozesadore libreetan eta onlinekoetan ere badaude eskuragarri; horrek esan nahi du ez dela beharrezkoa software garestia erostea funtzio horietarako sarbidea izateko. Hona hemen lineako testu-prozesadore libreetan eta onlinekoetan aurkituko ditugun ohiko tresnetako batzuk.

### Ohiko tresnak

- 1 | Testu-edizioa:** aukera ematen dute testu-dokumentuak sortzeko, editatzeko eta haiei formatua emateko. Iturriak, tamainak, estiloak, koloreak, testu-lerrokadura alda daitezke, eta letra lodia, etzana, azpimarratua eta bestelako testu-efektuak aplikatu daitezke.
- 2 | Formatu-tresnak:** bai testu-prozesadore libreek, bai online testu-prozesadoreek formatu-tresnak dituzte dokumentuaren diseinua doitzeko, hala nola marjinak, lerroen eta lerrokaden arteko tarteak, koskak, eta abar.
- 3 | Irudiak eta beste multimedia-elementu batzuk txertatzea:** aukera ematen dute dokumentuetan irudiak, grafikoak, taulak eta bestelako multimedia-elementuak txertatzeko. Elementu horiek editatu, doitu, birdimentsionatu eta dokumentuaren barruan mugitu daitezke, beharrezkoa den heinean.
- 4 | Testua berrikustea:** bai testu-prozesadore libreek, bai online testu-prozesadoreek testua berrikusteko tresnak dituzte, ortografia eta gramatika zuzentzeko, bai eta dokumentuko aldaketan eta iruzkinen jarraipena egiteko ere.
- 5 | Dokumentuak gorde eta partekatzea:** dokumentuak askotariko formatuetan gordetzeko aukera ere ematen dute (besteak beste, DOC, DOCX, PDF eta RTF). Gainera, dokumentuak beste erabiltzaile batzuekin parteka daitezke, denbora errealean elkarlanean aritzeko eta berrikuspenak nahiz iruzkinak egiteko.





**6 | Txantiloiak eta estiloak:** bai testu-prozesadore libreek, bai online testu-prozesadoreek aurrez definitutako txantiloiak eta estiloak eskaintzen dituzte, dokumentuei itxura profesionala eta sendoa emateko. Txantiloia eta estiloak erabiltzailearen beharren arabera pertsonaliza daitezke.

**7 | Bilatu eta ordeztzeko tresnak:** bi testu-prozesadore motek bilatu eta ordeztzeko tresnak dituzte; aukera ematen dute dokumentuan hitzak edo esaldiak aurkitu eta ordeztzeko, eta horrek erraztu egiten du dokumentu luzeak editatzea eta zuzentzea.

**8 | Hodeian autogordetzea eta sinkronizatzea:** online testu-prozesadoreen abantaila da hodeian sinkronizatzeko aukera ematen dutela, eta horri esker, dokumentuak automatikoki online gordetzen dira, eta askotariko gailuetatik atzi daitezke. Horrek lan kolaboratiboa errazten du, bai eta dokumentuak edozein lekutatik eta edozein unetan atzitzea ere.

## Prozesadore libreak eta online prozesadoreak

Testu-prozesadore libreen adibide batzuk: LibreOffice Writer eta Apache OpenOffice Writer. Kode irekiko suiteen parte dira, eta doan deskargatu eta erabil daitezke ordenagailu lokaletan. Online testu-prozesadoreen adibide batzuk: Google Docs eta Microsoft Office Online; web-aplikazioak dira, eta web-nabigatzaile baten bidez joko dugu haietara. Hodeian testua prozesatzeko tresnak eskaintzen dituzte.

Taula honetan testua editatzeko programa libre batzuk daude, eta zer webgunetan aurkituko ditugun.

### Testu-prozesadoreen alternatibak

Programa	Bateragarritasuna Officerekin	Webgunea
OpenOffice Writer	bistaratu	<a href="http://openoffice.org">openoffice.org</a>
LibreOffice Writer	dokumentuak bistaratu/aldatu	<a href="http://es.libreoffice.org">es.libreoffice.org</a>
Google Docs	dokumentuak bistaratu/aldatu	<a href="http://docs.google.com">docs.google.com</a>
WPS Office Writer	dokumentuak bistaratu/aldatu	<a href="http://wps.com/office">wps.com/office</a>



### **i** Informazio gehiago

Ezagutu testua online prozesatzeko tresnen aukera guztiak. Aukera ematen dute testua editatzeko eta formateatzeko ez ezik, baita hodeian gordetzeko, denbora errealeko lankidetzan aritzeko eta beste aplikazio erabilgarri batzuekin integratzeko ere.



Web-orri honetan Microsoft Worden doako 12 aukera dituzu.

[e.digitall.org.es/freeword-processors](https://e.digitall.org.es/freeword-processors)



Eduki digitalak  
sortzea

*B2 maila* 3.1 Edukiak  
garatzea

# Testuari aplikatutako efektuak



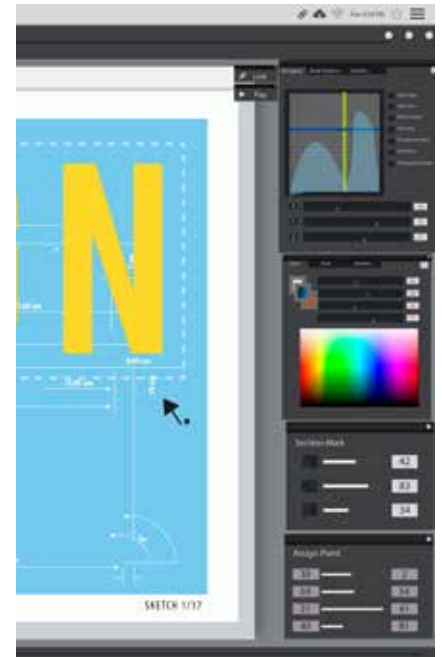


## Testuari aplikatutako efektuak

Diseinu grafikoko testua ez da, besterik gabe, mihise batean hitzak idaztea; tresna indartsu eta dinamikoa izan daiteke lana eraldatzeko eta erakargarriagoa egiteko.

Diseinu grafikoko tresnek (besteak beste, Adobe Photoshop, Gimp edo Illustrator) aukera ematen digute testu-efektu harrigarriekin lan egiteko aukera ematen digute, hots, testu-prozesadore estandar batek (adibidez, Microsoft Word) egin dezakeenaz haragoko efektuekin. Wordek testu-efektuen oinarriko sorta bat eskaintzen duen arren, diseinu grafikoko tresnek sakontasun eta malgutasun handiagoa eskaintzen dute.

Jarraian, testua diseinatzeko aukeretako batzuk eta haien utilitate nagusiak deskribatuko ditugu.



### Efektu arrunt batzuen definizioak

#### 1 | Alaka eta erliebea.

Alaka eta erliebearen efektuek 3D itxura ematen diote testuari. Efektu horien ondorioz badirudi testua gailentzen edo hondoratzen ari dela orrian. Doikuntzei esker, kontrola daitezke bai alakaren sakonera, bai argiaren norabidea, zeinak itzala eta erliebea sortzen dituen. Alakak testuaren eta hondoaren arteko trantsizioa sortzen du; oro har, testuaren ertzak hondoaren gainetik altxatuko balira bezalako efektuaren bidez. Erliebea, bestalde, alakaren antzekoa da, baina sakonera-itxura sortzen du testuaren barrutik kanporantz, eta horren ondorioz, testua zizelkatuta edo hondoan presionatuta dagoela dirudi.

#### 2 | Itzal paraleloa.

Itzal jaurtia ere esaten zaio; itzal paraleloak itzala gehitzen du testuaren atzean, eta horren ondorioz, testua hondoaren gainean flotatzen ari dela dirudi. Itzalaren distantzia, angelua eta lausotzea doitu daitezke askotariko efektuak sortzeko.



### 3 | Barneko eta kanpoko argiztapena (distira).

Bi efektu horiek funtsezkoak dira testuari testura- eta dimentsio-sentsazioa emateko. Testuaren argiztapena manipulatzeko, eszenan argi-iturri bat dagoelako sentsazioa sortzeko. Barruko argiztapenaren ondorioz ematen du argia testuaren erdigunetik datorrela, eta kanpoko argiztapenak, berriz, argia testuaren kanpotik datorrela. Efektu horiek erabilgarriak dira testu metalikoa edo neonezkoa sortzeko.

### 4 | Degradatua (degradatua gainjartzea).

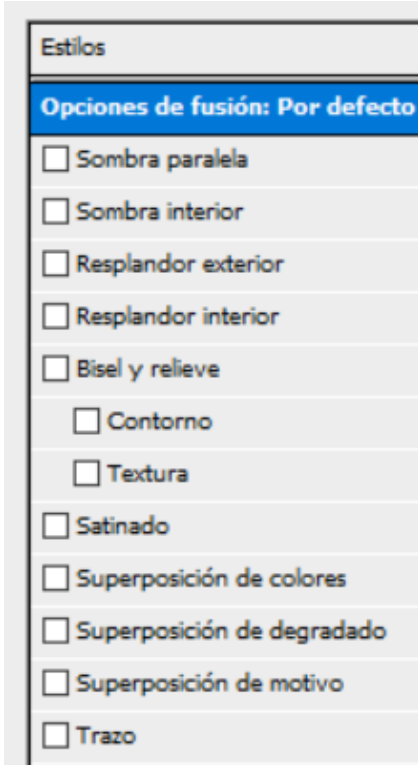
Efektu hau erabiltzen da testuaren barruan kolore batetik besterako trantsizio leuna sortzeko. Erabil daiteke sakonera gehitzeko, argi- eta itzal-efektu bat sortzeko, edo, besterik gabe, testua bisualki interesgarriagoa eta erakargarriagoa izan dadin.

## Adibidea Adobe Photoshop eta GIMP programetan

Jarraian, testu simple bat jarriko dugu mihise batean, eta aurretik azaldutako tresna bakoitza aldatuko dugu, Adobe Photoshopen geruza-estiloaren menu zabalgarria erabiliz (1. irudia - eskuinean).

Era horretan, 2. iruditik 3. irudira igaroko gara (behean), honako hauek erabilia: itzala paralelo normala, % 77ko opakutasunarekin eta 150°-ko angeluarekin; barneko distira bilbea trama fusio-moduarekin eta % 75eko opakutasunarekin; alaka, barneko ingletearekin eta % 300eko sakonerarekin, eta % 42ko degradatu-gainjartzearekin.

1. irudia. Geruza-estiloak.



2. irudia. Hasierako testua.



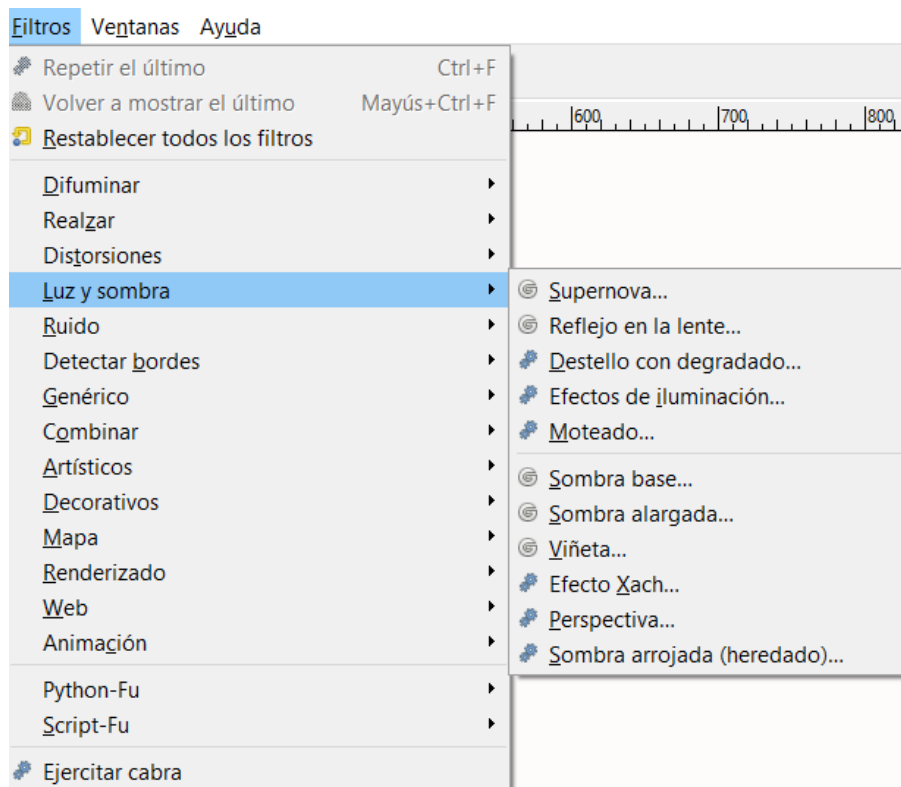
3. irudia. Testu editatua.



1. irudia. Geruza-estiloak.



GIMP programan gauza bera egiteko, iragazkien fitxa erabiliko dugu, eta haren bitartez hainbat ezaugarri gehitu ahal izango dizkiogu testuari, hala nola argiztapena, itzala eta testurak.



4. irudia. Iragazkien fitxa.

### Informazio gehiago

Esperimentatzea funtsezkoa da. Diseinu grafikoko tresnek aukera lehenetsiak izan ohi dituzte efektu horietarako, baina pertsonalizazio-maila handirako aukera ere badago. Garrantzitsua da gogoraraztea batzuetan gutxiago gehiago dela. Testu-efektuak transmititzen ari den mezua hobetzeko erabili behar dira, eta EZ mezutik arreta galarazteko.

Gomendagarria da **Adoberen** orri ofiziala ([adobe.com](https://adobe.com)), edo **GIMP** ([docs.gimp.org](https://docs.gimp.org)) programarena bisitatzea, tutorial eta urratsez urratseko gida oso baliagarriak dituztelako.





Eduki digitalak  
sortzea

**B2 maila** 3.1 Edukiak  
garatzea

# Irudi animatuak sortzea





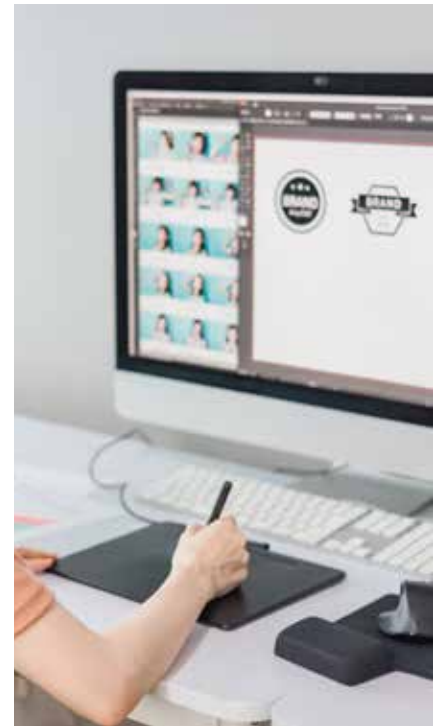
## Irudi animatuak sortzea

Irudi animatuak sortzea praktika eta pazientzia eskatzen duen diziplina bat da. Hala ere, tresna egokiak eta ezagutza zuzena edukiz gero, animazio harrigarriak sor ditzakezu zure publikoaren arreta bereganatzeko.

Jarraian, irudi horiek urratsez urrats nola egin azalduko dugu.

### Aukeratu software egokia

GIMPen funtzio erabilgarrienetako bat hauxe da: GIF edo PNG animatuko irudiak sortzeko gaitasuna. Formatu horiek asko erabiltzen dira webgunean eta beste bitarteko digital batzuetan, moldakorrak direlako eta animazioa jasateko gaitasuna dutelako. Txosten honetan, urratsez urrats deskribatuko dugu nola sortu horrelako irudiak GIMP erabiliz.



### Prozedura

#### Animaziorako geruzak sortzea

Zure animazioko koadro bakoitza GIMPeke geruza bati dagokio. Beste geruza bat sortzeko, joan, menuan, «Geruza» aukerara, eta gero, «Geruza berria» aukerara. Leiho bat irekiko da, eta, bertan, izen bat esleitu ahal izango diozu geruzari (adibidez, «1. koadro»), tamaina zehaztu ahal izango duzu (jatorrizko irudiarena ez bada), bai eta hondoko kolorea ere.

Errepikatu prozesua zure animazioko koadro bakoitzean. Gogoratu geruzen ordenak geruzen panelean (normalean mihisearen eskuinaldean) zehaztuko duela koadroen ordena animazioan. Goiko geruza lehen koadroari dagokio, eta beheko geruza azken koadroari. Zure animazioko koadro bakoitza GIMPeke geruza bereizi bat izango da. Klikatu «Geruza» > «Geruza berria» aukeran, zure koadroak sortzen hasteko.

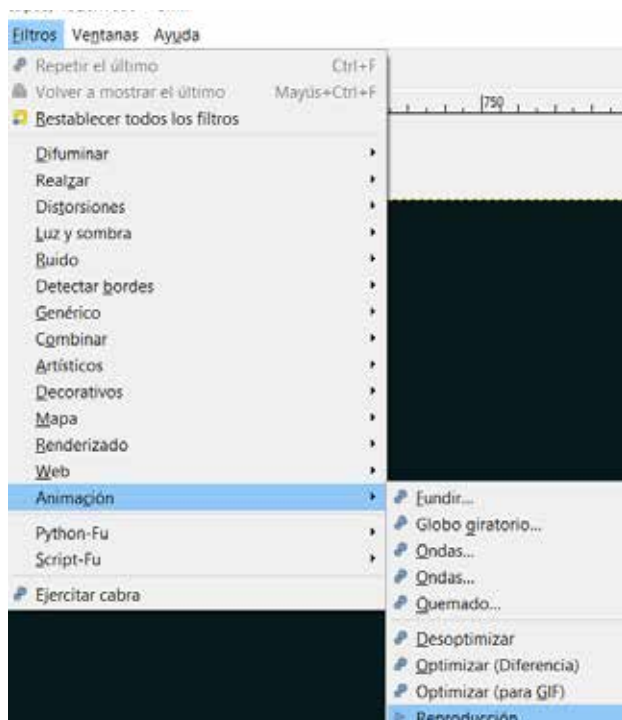


## Animaziorako geruzak sortzea

Geruzak sortuta, zure animazioko koadro bakoitza diseinatzeko has zaitezke. Aukeratu geruzak-panelen zer geruzak editatu nahi dituzun, eta erabili GIMPeko tresnak zure diseinua sortzeko. Adibidez, lehen koadroan lerro bat marraztu nahi baduzu, hautatu «1. koadroa» geruzak, aukeratu tresnak-panelen «Arkatza» tresnak, aukeratu kolorea eta marraztu zure lerroa. Marrazketako, betegarriko, testuko eta bestelako tresnak gehi ditzakezu.

## Animazioaren aurrebista

Zure geruzak guztiak diseinatzeko amaitu ondoren, zure animazioaren aurrebista lor dezakezu. Joan «Iragazkiak» menura, gero «Animazioa» atalera eta hautatu «Erreproduzitu». Aurrebista-leiho bat irekiko da, zure animazioa nola gelditu den ikusteko. Doikuntzaren bat egin behar baduzu, itxi aurrebista-leihoak eta itzuli zure geruzetara, beharrezko diren aldaketak egiteko.





## GIF edo PNG animatu gisa esportatzea

Zure animazioarekin pozik zaudenean, esportatzeko ordua da. Joan «Fitxategia» menura eta hautatu «Esportatu honela». Elkarrizketa-leihoan, aukeratu zer formatu nahi duzun (GIF edo PNG), jarri izen bat zure fitxategiari eta klikatu «Esportatu» aukeran. Elkarrizketa-koadro honetan, markatu «Animazio gisa» laukia. Beste parametro batzuk ere doitu ditzakezu, hala nola koadroen arteko atzerapen-denbora eta animazioa mugarik gabe errepikatuko den. Amaitu ondoren, klikatu «Esportatu» aukeran.



### **i** Informazio gehiago

Animazioa arte bat da, eta arte guztietan bezala, badira jarraitu beharreko printzipio batzuk lan eraginkorrak eta erakargarriak sortuko baditugu. Disney Ollie Johnston eta Frank Thomas animatzaileek animazioaren 12 printzipioak finkatu zituzten; abiapuntu bikaina dira, eta animazioaren «Biblia»tzat hartzen dira.

Xehetasun gehiago nahi izanez gero, programako tutorialak ikustea gomendatzen dizugu.

[gimp.org.es/tutoriales/animaciones](http://gimp.org.es/tutoriales/animaciones)



Eduki digitalak  
sortzea

**B2 maila** 3.1 Edukiak  
garatzea

# Diseinu grafikoko tresnak (libreak eta hodeian)





## Diseinu grafikoko tresnak (libreak eta hodeian)

### Irudi-diseinuaren hastapenak

Irudi bat diseinatzea funtsezko gaia da eguneroko edo laneko egoera askotan, eta horri esker, gure argazkiak hobetu ahal izango ditugu, argazkizaleak bagara, eta, diseinu grafikoa bidez, curriculum erakargarriagoa lortuko dugu, lanpostu bat eskuratzeko orduan beste hautagai batzuegandik bereizi gaitzen.

Diseinu grafiko ona egiteko, beti izango dugu profesional batengana jotzeko aukera; halere, gaur egun dauden askotariko tresnek nabarmen erraztuko digute irudiak diseinatzeko lana. Tresna horiek, oro har, diseinu-idea lehenetsi batzuk ematen dizkiote erabiltzaileari, txantilo moduan, eta erraz editatzeko aukera ematen dute. Horrek izugarri errazten du gure irudia diseinatzea, baita diseinu grafikoa buruzko oinarrizko nozioak baino ez baditugu ere.

Dokumentu honetan, tresna horiek bi multzo handitan banatuko ditugu: gure ekipoa instalatzeko tresna libreak eta webgunean eskuragarri daudenak.

### 2D irudiak diseinatzeko tresna libreak

2D irudiak diseinatzeko tresna libreak alternatiba eskuragarria, pertsonalizagarria eta kalitate handikoa dira software jabetuneko tresnen aldean, eta horregatik, aukera ona dira diseinatzaile eta artistentzat.

Tresna horien abantaila handien artean dago doan deskargatu eta erabil daitezkeela, eta kode irekikoak direla; horri esker, erabiltzaileek komunitatean lan egin dezakete, eta hobekuntzak edo funtzionalitate berriak proposa ditzaketela.

Arlo honetan, dagoeneko lan egin dugu gehien, ziurrenik, erabiltzen diren biek:

- **GIMP:** irudi-editore honek askotariko tresnak ditu argazkiak editatzeko, grafikoak sortzeko eta ilustrazio digitalerako.
- **Inkscape:** grafiko bektorialen software hau ilustrazioak, grafikoak, logotipoak eta diseinu konplexuak sortzeko erabiltzen da.





Software pribatiboan gehien erabiltzen diren alternatibak Adobe Photoshop eta Adobe Illustrator dira, hurrenez hurren.

## Hodeian lan egiteko tresnak

Hodeian tresna ugari ditugu irudien diseinua lantzeko. Bakoitzak bere berezitasunak baditu ere, tresna guztietan dauden ezaugarri orokor batzuk deskribatuko ditugu:

- **Txantiloiak:** aurrez diseinatutako txantilo ugari dituzte diseinuak askotariko formatuetan sortzeko (adibidez, sare sozialetako argitalpenak, aurkezpenak eta bisita-txartelak).
- **Baliabideen liburutegia:** baliabideen liburutegi zabala eskaintzen dute, hala nola irudiak, ikonoak eta iturriak, diseinuak pertsonalizatzeko eta hobetzeko.
- **Editore erabilerraza:** intuitiboak eta irisgarriak dira, eta tresna sinpleak dituzte elementuak gehitzeko eta editatzeko, koloreak doitzeko, iturriak aldatzeko eta abar.
- **Lankidetzak:** erabiltzaileei aukera ematen diete proiektuak partekatzeko eta haietan lankidetzan aritzeko, eta hori oso aproposa da diseinu-taldeentzat edo talde-lanak egiteko.
- **Hodeian biltegitratzea:** diseinuak hodeian biltegitratzen dituzte, eta horrek aukera ematen du haiek edozein gailutatik atzitzeko eta editatzeko, betiere Interneteko konexioa edukiz gero.
- **Irteera-formatu ugari:** gure diseinuak irudi-formatu askotara esportatzeko aukera ematen dute. Batzuek beste formatu batzuetara esportatzeko aukera ematen dute (adibidez, PDF), edo are aurkezpen edo bideo laburrak sortzeko ere.

Eskuragarri dauden guztietatik, ziurrenik Canva da erabilienetako bat, eta, horregatik, labur-labur deskribatuko ditugu haren ezaugarri nagusiak.

### OHARRA

Gogoratu, tresna horiez gain, irudi-banku libreak ere badituzula erabilgarri, hala nola Pixabay; kalitatezko irudi ugari jartzen dituzte zure eskura.





## Irudiak Canva erabiliz diseinatzea

Diseinu-tresna ezagunenetakoa bat da. Irudiak modu errazean diseinatzeko aukera emateaz gain, irudian ikusten dugun bezala, liburutegi zabala ere badu, eta bertan dauden txantiloiek lagunduko digute gure irudia hutsetik sortzen. Horrela, Canvak txantilo ugari eskaintzen dizkigu gaika banatuta (adibidez, sare sozialak), eta gaiok, era berean, beste zehatzago batzuetan banatzen dira (adibidez, Instagramen istorioak). Azkenik, aukera ematen digu, halaber, gai bat aukeratu beharrean, aukera zabaletik zer dokumentu mota sortu nahi dugun aukeratzeko (adibidez, infografia bat), eta, ondoren, txantiloiak eskaintzen dizkigu dokumentua erraz editatzeko, eta ondorioz, estetika atsegineko ideia bat ematen digu. Gainera, zenbait ordainketa-plan ditu, doako kontuaren ezaugarriak hobetzeko; haietan funtzionalitate gehiago daude, eta baliteke interesgarriak izatea maila profesionalan. Adibidez, doako kontuak 5 Gb-ko memoria ematen du, eta profesionalak, berriz, 1 Tb-ra arte.



Canva tresnaren xehetasuna, diseinu-zentroaren barruan txantiloiak blokeka antolatuta erakusten.





## Ondorioak

Laburbilduz, irudiak diseinatzeko tresna horiek zabaldua daude irisgarriak malguak eta eskuragarriak direlako. Bereziki, hodeian irudiak diseinatzeko lineako tresnek abantaila erantsi bat dute: diseinatzaileek edozein lekutatik lan egin dezakete beren diseinuetan, Internetarako konexioa edukiz gero, eta beste diseinatzaile batzuekin proiektuak parteka ditzakete eta lankidetzan jardun dezakete.

Online lankidetzak garrantzitsua ez dela iruditu arren, tresna horien oso ezaugarri garrantzitsua da; izan ere, toki berean ez dauden lantalde bateko kideek aukera dute denbora errealean proiektu batean lankidetzan aritzeko, eta horrek hobetu egiten du eraginkortasuna —taldekideek edozein lekutatik parte har baitezakete, Internetarako konexioa edukiz gero—, bai eta lanaren kalitatea ere —lantaldeko kideek trebetasun eta ikuspegi desberdinak eskaini baititzakete—.

### Informazio gehiago

Animatu zure lehen diseinuak hodeian egitera. Guk Canva erabiltzea gomendatzen dizugu, ezaugarri guztiak betetzen baititu eremu horretako tresna arrakastatsua izan dadin, eta, oso erabilia denez, erraza da informazioa nahiz probatzeko ideia berriak aurkitzea. Baina hona hemen bost aukera oso interesgarri:

- **Adobe Express:** [adobe.com/es/express](https://adobe.com/es/express)
- **Desygner:** [desygner.com/es](https://desygner.com/es)
- **Piktochart:** [piktochart.com](https://piktochart.com)
- **Visme:** [visme.co/es](https://visme.co/es)
- **Vista Create:** [create.vista.com/es](https://create.vista.com/es)



Eduki digitalak  
sortzea

**B2 maila** 3.1 Edukiak  
garatzea

**Irudiak  
eta bideoa  
loteka  
konprimatzea**





# Irudiak eta bideoa loteka konprimatzea

## Irudiak eta bideoa loteka konprimatzea

Irudi bat baino gehiago konprimatu nahi izanez gero, komeni da zeregin hori aldi berean egitea, denbora eta lana optimizatzeko. Hori egiteko modu bat loteka konprimatzea da. Zeregin hori honako urrats hauen bidez egin daiteke:

**1 | Irudiak antolatzea:** ordenagailuan karpeta edo direktorio bat sortzea eta direktorio horretan konprimatu nahi ditugun irudi guztiak jartzea. Ziurtatu behar da irudiak konpresioarekin bateragarria den formatu batean daudela (adibidez, JPEG edo PNG).

**2 | Sortaka konprimatzeko tresna bat aukeratzea:** hainbat tresna daude eskuragarri, irudiak loteka azkar eta erraz konprimatzeko. Hona hemen aukera ezagun batzuk:

- **Adobe Photoshop:** Adobe Photoshoperako sarbidea izanez gero, haren «irudi-prozesadorea» funtzioa erabil daiteke irudiak loteka konprimatzeko. Konpresioa egiten duen ekintza bat sortuz, eta, ondoren, direktorioko irudi guztiei aplikatuz.
- **GIMP:** dagoeneko hitz egin eta landu dugu ikastaro honetan irudiak editatzeko doako eta kode irekiko software libreko alternatiba honekin. «Lote-prozesadorea» funtzioa erabil daiteke nahi ditugun irudiak konprimatzeko. Prozesadorea konfiguratu behar da hautatutako direktorioan irudi guztiei konpresioa aplikatzeko.
- **Online utilitateak:** online tresna ugari daude irudiak loteka eta doan konprimatzeko. Erabili ohi diren aukera batzuk: **TinyPNG** ([tinypng.com](http://tinypng.com)), **Compressor.io** ([compressor.io](http://compressor.io)) eta **Optimizilla** ([imagecompressor.com/es](http://imagecompressor.com/es)). Tresna horiei esker, askotariko irudiak aldi berean kargatu eta konprimatu ahal dira.



Loteka prozesatzea GIMPen. Aukera ematen du ekintza bakar batez irudi batzuk birdimentsionatu, konprimatu eta esportatzeko.



**3 | Konpresioa konfiguratzeko:** erabili beharreko tresna hautatu ondoren, baliteke konpresio-doikuntzak konfiguratu behar izatea. Nahi den konpresio-maila aukera daiteke; izan ere, ikastaro honetan azaldu dugun bezala, konpresio-maila handiagoak gehiago murrizten du fitxategiaren tamaina, baina baliteke irudiaren kalitatean ere eragitea. Premietarako egokia den oreka aurkitu behar da fitxategiaren kalitatearen eta tamainaren artean.

**4 | Konpresio-prozesua hasteko:** konpresio-doikuntzak konfiguratu ondoren, loteka konprimatzeko prozesua hasi behar da. Horretarako, baliteke beharrezkoa izatea irudiak online tresnan kargatzea edo lote-prozesadorearen irudiak editatzeko softwarean exekutatzeko. Tresnak irudi guztiak prozesatuko dituzte hautatutako direktorioan, eta doikuntzen arabera konprimatuko dituzte.

**5 | Irudi konprimatuak gordetzea:** konpresio-prozesua amaitu ondoren, tresnak irudi konprimatuak emango ditu. Berariazko kokapen batean gorde beharko dira, jatorrizko irudiekin nahas ez daitezela. Horretarako, gomendagarria da beste direktorio bat sortzea edo fitxategi-izenei atzikia gehitzea, irudi konprimatuak bereizteko.



Komeni da jatorrizko irudien segurtasun-kopia bat egitea konprimatu aurretik, prozesuan zehar arazoren bat gertatzen bada daturik gal ez dadin.

Premietarako egokia den oreka aurkitu behar da fitxategiaren kalitatearen eta tamainaren artean.



## Bideoa loteka konprimatzea

Zenbait bideo loteka konprimatzea baliagarria da fitxategien tamaina murrizteko eta haien banaketa errazteko. Bideoak loteka konprimatzeko oinarritzko urratsak irudiekin egin beharreko berak izango dira:

**1 | Antolatu bideoak:** irudiekin bezala, komenigarria izango da ordenagailuan karpeta edo direktorio bat sortzea eta konprimatu nahi ditugun bideo guztiak direktorio horretan jartzea. Komeni da egiaztatzea bideoak konpresioarekin bateragarriak diren formatuetan daudela (adibidez, MP4, AVI eta MKV).

**2 | Loteka konprimatzeko tresna bat aukeratzea:** zenbait tresna erabilgarri bideoak loteka eraginkortasunez konprimatzeko aukera ematen dute. Hona hemen aukera batzuk:

- **HandBrake:** kode irekiko doako tresna honek bideoak loteka konprimatzeko aukera ematen du. Konpresio-doikuntzak konfiguratu daitezke, hala nola tamaina, kalitatea eta irteera-formatua, eta gero, hautatutako direktorioan bideo guztiei aplikatu.
- **FFmpeg:** bideoak prozesatzeko komando-lerro oso ahaltsu eta moldakorra da. Loteko script bat idatz daiteke, eta hark FFmpeg erabiltzea bideo guztiak nahi dugun direktorioan konprimatzeko. Script-ak konpresio-parametroak doitzeko beharrezkoak diren komandoak izango ditu, hala nola bit-tasa, bereizmena eta irteera-formatua.
- **Bideoa editatzeko programak:** bideoak editatzeko programa batzuek, hala nola Adobe Premiere Pro eta Final Cut Pro programek, bideoak loteka konprimatzeko gaitasuna ere badute. Bideo guztiak programan inporta daitezke, konpresio-doikuntzak konfiguratu, eta ondoren, lote gisa esporta daitezke.

**3 | Konpresio-doikuntzak konfiguratzeko:** konpresio-tresna hautatu ondoren, konpresio-doikuntzak beharren arabera konfiguratu behar dira. Zenbait parametro doitu daitezke, hala nola bereizmena, bit-tasa, konpresio-kodeka eta lortutako fitxategiaren kalitatean eta tamainan eragina



duten beste alderdi batzuk. Irudietan bezala, garrantzitsua da oreka aurkitzea kalitate bisualaren eta nahi dugun fitxategi-tamainaren artean.

**4 | Loteka konprimatzeko prozesua gauzatzea:** konpresio-doikuntzak konfiguratu ondoren, loteka konprimatzeko prozesua hasi behar da hautatutako tresna erabiliz. Horrek berekin ekarriko du bideoak tresnan kargatzea eta haietako bakoitzean konpresio-doikuntzak aplikatzea, edo loteko script bat exekutatzea, baldin eta komando-lerroko tresnaren bat erabiltzen ari bagara.

**5 | Bideo konprimatuak gordetzea:** konpresio-prozesua amaitu ondoren, tresnak konprimatuta emango ditu bideoak. Berriz ere, kokapen espezifiko batean gorde behar dira, jatorrizko bideoekin nahas ez daitezten.

Bideo-konpresioak kalitate-galera dakar, eta, horregatik, komeni da aldez aurretik bideo bakar batekin probak egitea eta konpresio-parametroak doitzea, era horretan, betiere gure beharren arabera, oreka egokia aurki dezagun kalitatearen eta fitxategi-tamainaren artean.

#### Informazio gehiago

Gehiago jakin nahi baduzu, kontsultatu Cliff Wootton-en *Compresión de audio y video* liburua (Anaya argitaletxea).





Eduki digitalak  
sortzea

**B2 maila** 3.1 Edukiak  
garatzea

# Webguneak sortzeko aukerak





## Webguneak sortzeko aukerak

### Sarrera

Azalduko ditugun tresnak izan aurretik, web-orri bat sortzeko mugarik handiena zen programazio-ezagutzak ezinbestekoak zirela. Kodea idaztea trebetasun ikasten zaila da, eta pertsona eta enpresa askok ezin zituzten web-garatzailleak kontratatu webgunea sor ziezaieten.

Alabaina, webguneak sortzeko plataforma irisgarriak eta erabilerrazak sortu zirenean, muga hori nabarmen murriztu zen. Orain, ordenagailua (edo telefono mugikorra) eta Interneteko konexioa duen edonork sor dezake itxura profesional eta pertsonalizatuko web-orri bat, hutsetik kodetzen ikasi beharrik gabe.

### Gaur egungo aukerak

Orain web-orriak hutsetik kodea idatzi beharrik gabe sortzeko aukera ezagunenetako batzuk aurkeztuko dizkizugu. Bakoitzak bere alderdi positiboak eta negatiboak ditu, eta aukera optimoa erabiltzaile bakoitzaren beharren arabera izango da. Oro har, bi kategoriatan sailkatzen dira:

- **Webguneak eraikitze online plataformak:** hodeian ostatatutako tresnak dira, eta normalean, «heldu eta askatzeko» sistemetan oinarritzen dira (ingelesez, *drag and drop*); hau da, erabiltzaileek elementuak gehitu eta pertsonalizatu ditzakete beren webgunean, eta aski dute haiek arrastatzea eta nahi duten lekuan jartzea. Gainera, aurrez diseinatutako txantiloiak izaten dituzte, eta erabiltzaileek pertsonalizatu ahal dituzte. Halakoak dira, besteak beste, Wix, Weebly eta Squarespace tresnak.







- **Edukiak kudeatzeko sistemak (CMS, ingelesezko siglen arabera):** aurrekoek ez bezala, CMSen potentzial guztiari etekina ateratzeko, ezagutza tekniko handixeagoa behar da, baita kodetze-trebetasunak ere. Wordpress da ezagunena eta gehien erabiltzen dena; gai eta plugin ugari eskaintzen ditu, hau da, aukera ugari, gure webgunea pertsonalizatzeko eta itxura profesionala emateko. Adibidez, Wordpress-ek «Elementor» plugina eskaintzen du; bisualki editatzeko aukera ematen digu (*drag and drop* bidez), kodea idatzi beharrik gabe. WordPress webguneak sortzeko plataforma egokiena da webgune handiago eta konplexuetarako, halakoek malgutasun eta eskalagarritasun handiagoa behar baitute.

Pentsatzekoa denez, tresna horiei esker, hala partikularrek nola enpresek aukera izan dute online presentzia izateko, eta irismena handitu ahal izan dute trebetasun tekniko garestietan inbertitu edo web-garatzailleengana jo beharrik gabe.

### Informazio gehiago

Helbide interesgarriak:

- **Wix:** [es.wix.com](https://es.wix.com)
- **Weebly:** [weebly.com/es](https://weebly.com/es)
- **Squarespace:** [es.squarespace.com](https://es.squarespace.com)
- **Wordpress:** [wordpress.com](https://wordpress.com)
- **Visual Studio:** [visualstudio.microsoft.com/es](https://visualstudio.microsoft.com/es)
- **Eclipse:** [eclipse.org/ide](https://eclipse.org/ide)
- **NetBeans:** [netbeans.apache.org](https://netbeans.apache.org)
- **IntelliJ IDEA:** [jetbrains.com/idea](https://jetbrains.com/idea)

## Lanbide-inguruneak

Web-garatzaille profesionalek **garapen-ingurune**etan lan egin ohi dute (IDE, ingelesezko siglen arabera), eta aplikazio horiek kodea idazteko, probatzeko eta arazteko tresnak ematen dizkiete garatzaileei. Funtsean, IDE delakoa programa bat da, eta garatzaileei aplikazioak, webguneak, softwarea eta beste programa batzuk sortzeko aukera ematen die. IDEek honako hauek hartzen dituzte barne: testu-editorea, arazteko tresnak, konpiladorea, araztailea, kode-sortzailea eta garatzaileei modu eraginkorragoan lan egiten laguntzen dieten beste tresna batzuk. Nahiz eta CMSek baino muga gutxiago



dituzten, programazio-ezagutzak beharrezkoak dira haien funtzionaltasunak ahalik eta gehien aprobetxatzeko. Hona hemen garapen-ingurune ezagunen adibide batzuk: Visual Studio (Microsoft), Eclipse, NetBeans eta IntelliJ IDEA.

## Ondorioak

Gaur egun, askotariko aukerak daude webguneak sortzeko, eta kodeak hutsetik sortzea baino gehiago dira. Edukien kudeatzaileek (adibidez, WordPress) webguneak sortzeko modu irigarri eta errazagoa eskaintzen dute, programazioari buruzko ezagutza aurreraturik izan beharrik gabe. Gainera, badira beste garapen-ingurune aurreratuago batzuk ezagutza teknikoak izan eta webgune konplexu eta pertsonalatuagoak garatu nahi dituztenentzat. Azken batean, aukera horiek webguneak sortzeko malgutasun eta moldakortasun handiagoa ahalbidetzen dute, eta erabiltzailearen gogobetetasun eta esperientzia handiagoa dakar horrek.



### OHARRA

WordPress-n **gaia** zera da: webgune baten diseinua eta itxura zehazten duten fitxategiak. Gaiak WordPress plataforman deskargatu eta instalatzen dira, eta erraz pertsonalizatu daitezke, webgune baten premia eta lehenetasun indibidualetara egokitzeko.

**Pluginak** software-pieza txikiak dira, eta WordPress plataforman deskargatu eta instala daitezke webgune bati funtzionalitate eta ezaugarri gehigarriak eransteko, hala nola harremanetarako formularioak, irudi-galeriak, sare sozialen integrazioa, etab.

### Informazio gehiago

Wordpress sistema hobeto ulertzeko, jo ezazu plataformak ematen duen dokumentaziora, bai lehenengo zalantzak argitzeko, bai gai aurreratuagoetarako.

[es.wordpress.org/support](https://es.wordpress.org/support)





Eduki digitalak  
sortzea

**B2 maila** 3.1 Edukiak  
garatzea

# WCAGeko egokitze-mailak



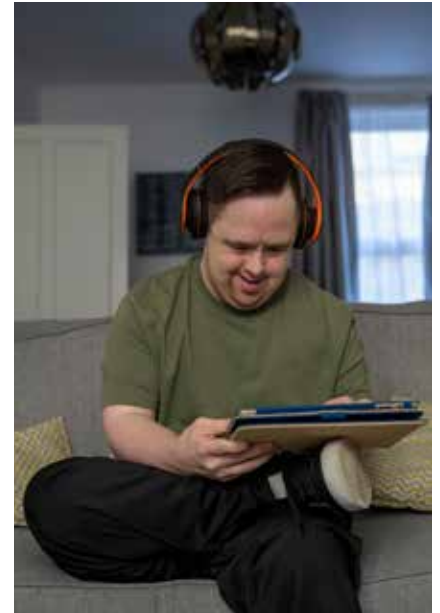


## WCAGeko egokitze-mailak

**WCAGeko ( Web Content Accessibility Guidelines) (e.digital.org.es/niveles-WCAG)** adostasun-mailak webgune baten irisgarritasuna sailkatzeko eta ebaluatzeko modu bat dira. 2008ko abenduan World Wide Web Consortium-ek (W3C) argitaratu zituen WCAGen jarraibideen 2.0 bertsioan sartu ziren. Bertsio eguneratu horrek 1.0 bertsioaren lehenetasunen ikuspegia ordeztu zuen, egokitzapen-mailen sistema batekin.

Jarraibide horiek gomendio zehatzak eta teknikoak ematen dituzte, web-edukia eskuragarriagoa izan dadin. WCAGen 2.0 bertsioan, guztira 12 jarraibide nagusi ezarri ziren, funtsezko lau printzipiotan multzokatuta. Hauek dira printzipioak:

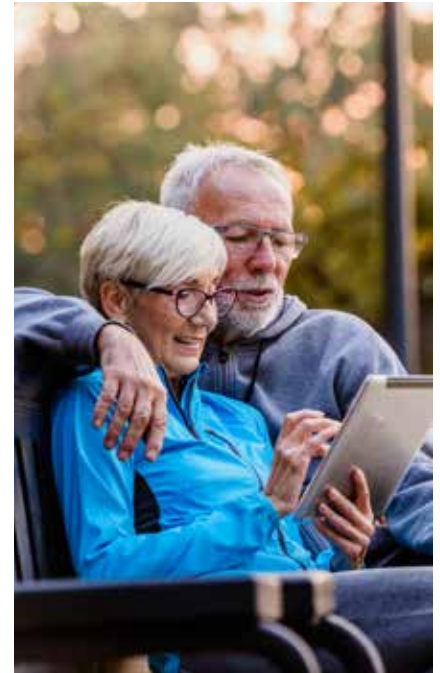
- 1 | Hautemangarria:** web-edukia erabiltzaile guztiek ikusteko moduan aurkeztu behar da, baita ikusteko eta entzuteko desgaitasuna dutenek ere. Printzipio honen jarraibideek honako alderdi hauek jorratzen dituzte, besteak beste: kolore-kontrastea, irudi eta baliabideetarako testu-alternatiben erabilera eta edukiaren aurkezpena kontrolatzeko gaitasuna.
- 2 | Erabilgarria:** web-edukiak erabiltzaile guztiek erabiltzeko modukoa izan behar du, baita mugitzeko eta nabigatzeko zailtasuna dutenek ere. Printzipio honen jarraibideak nabigazioan eta interakzioan oinarritzen dira, barne hartuta, besteak beste, teklatuaren irisgarritasuna, eduki errepikakorra ezkutatzeko gaitasuna eta edukia irakurtzeko eta erabiltzeko denbora nahikoa.
- 3 | Ulergarria:** web-edukiak ulertzeko modukoa izan behar du erabiltzaile guztientzat, baita desgaitasun kognitiboa eta ulertzeko desgaitasuna dutenentzat ere. Printzipio honen jarraibideek barne hartzen dituzte hizkuntzaren argitasuna, edukiaren egitura eta antolamendua, eta laguntza eta orientazio argiak emateko gaitasuna.
- 4 | Sendoa:** web-edukiak sendoa izan behar du, eta bateragarria teknologia eta nabigatzaile ugariekin, askotariko erabiltzaileek eskuratu dezaten. Printzipio honen jarraibideek oinarri hartzen dute, besteak beste, web-estandarren erabilera, laguntza-teknologiekiko bateragarritasuna eta teknologiaren akats eta aldaketekiko erresistentea den edukia sortzea.





Jarraibide horietako bakoitza adostasun-irizpide espezifikoetan banatzen da. Irizpide horiek adierazten dute zer betekizun tekniko eta funtzional bete behar diren egokitzapen-mailak (A, AA eta AAA) lortzeko. Kontuan izan behar da WCAGen jarraibideak eta irizpideak eboluzionatzen ari direla oraindik, eta geroagoko bertsioak abiarazi direla, hala nola 2.2 bertsioa (2022ko maiatza) eta 3.0 bertsioa. Bertsio bakoitzak jarraibide eta irizpide berriak gehitzen ditu sortzen ari diren irisgarritasun-erronkei aurre egiteko eta desgaitasuna duten erabiltzaileek webgunean duten esperientzia hobetzeko.

Egokitze-mailak lortzeko bete behar diren baldintza teknikoak eta funtzionalak adierazten dituzten adostasun-irizpide espezifikoak A, AA eta AAA mailak dira.



## Adostasun-mailak

Egokitze-mailak sartzearen helburu nagusia izan zen web-irisgarritasuna ebaluatzeko eta hobetzeko egitura argiagoa eta malguagoa ematea. Egokitzapen-mailek ikuspegi progresiboa ahalbidetzen dute, eta, hari esker, garatzaileek eta diseinatzaileek irisgarritasuna etengabe hobetzeko lan egin dezakete hainbat etapatan. Maila horiek hiru kategoriatan antolatuta daude: **A, AA eta AAA**. Maila bakoitzak bete beharreko gogobetetasun-irizpide batzuk ditu gogobetetasun-maila hori lor dadin. Jarraian, xehetasunez azaltzen dira maila bakoitza eta haren gogobetetasun-irizpideak:

### A adostasun-maila

A mailak bete beharreko gutxieneko irisgarritasun-irizpideak ezartzen ditu. Irizpide horiek oinarrizko betekizunak dira nolabaiteko irisgarritasun-neurria lortzeko eta desgaitasunen bat duten pertsonak dituzten oztopo arruntenak gainditzeko. A maila betetzeak esan nahi du funtsezko neurriak hartu behar direla irisgarritasuna hobetzeko eta edukia desgaitasunen bat duten pertsona batzuentzat irisgarria dela bermatzeko.



Hona hemen gogobetetasun-irizpide batzuk:

- Irudien ordeztuak ematea.
- Goiburu-egitura logikoa erabiltzea edukia antolatzeko.
- Bermatzea formularioak irisgarriak direla eta teklatu bidez betetzeko modukoak izatea.

## AA adostasun-maila

AA maila A mailatik harago doa, eta irizpide gehigarri zorrotzagoak ezartzen ditu. Irizpide horiek betetzeak irisgarritasun handiagoa ematen du, eta esperientzia hobetzen du desgaitasuna duten pertsonen sorta handiago batentzat. AA maila da gomendatutako eta zabalki onartutako estandarra, irisgarritasun-maila handia ematen baitu, eta jorratzen baititu, horrez gain, ikusteko desgaitasunak, entzuteko desgaitasunak, desgaitasun kognitiboak eta mugitzeko desgaitasunak dituzten pertsonen aurre egin beharreko oztopo komun asko.

Hona hemen gogobetetasun-irizpide batzuk:

- Testuaren eta funtsaren arteko kontraste nahikoa ematea.
- Multimedia edukietarako audio-transkripzioak ematea.
- Estekak deskribatzaileak eta esanguratsuak direla ziurtatzea.

## AAA adostasun-maila

AAA maila da adostasun-mailarik altuena, eta irisgarritasunerako irizpiderik zorrotzenak eta osoenak ezartzen ditu. AAA mailako irizpideak betetzeak goragoko maila batera darama irisgarritasuna, eta esperientzia inklusiboa bermatzen du desgaitasun mota askorentzat. AAA maila lortzeko, baliteke aldaketa esanguratsuak egin behar izatea web-edukiaren diseinuan, egituran eta funtzionaltasunean, eta baliteke irizpideetako batzuk lortzea zaila izatea zenbait testuingurutan.

Hona hemen gogobetetasun-irizpide batzuk:

- Webgune osoan nabigazio argia eta sendoa eskaintzea.
- Erabiltzaileei itzarote-denbora doitzeko aukera ematea edukia automatikoki aldatu aurretik.
- Edukiaren bertsio sinplifikatua ematea, errazago uler dadin.





## Ondorioak

AA maila, oro har, gomendatutako estandartzat hartzen da, eta oso onartuta dago. AA mailako gogobetetasun-irizpideak betetzeak irisgarritasun sendoa ematen du, eta esperientzia hobetzen du desgaitasunen bat duten erabiltzaile gehienentzat. AAA maila, zorrotzagoa bada ere, baliteke lortzen zailagoa izatea, eta haren irizpide batzuk betetzea baliteke zaila izatea zenbait testuingurutan. Egokitze-mailak ez dira legez nahitaezkoak oro har (bai, ordea, administrazio publikoko atarietan), baina haiek betetzea oso gomendagarria da, eta baliteke eska ditzatela lege- edo gobernu-testuinguru jakin batzuetan. Irakaskuntzan batez ere, egokitasun-mailak betetzea lagungarria da askotariko erabiltzaileei esperientzia inklusiboa bermatzeko; era horretan, eredia izaten dira eta onurak ematen dizkiete desgaitasuna duten pertsonei nahiz gainerako erabiltzaileei.

### Informazio gehiago

WCAGeko edukiak zehatz-mehatz ezagutu nahi badituzu, berrikus itzazu haren webgunean.

[w3.org/TR/WCAG22](http://w3.org/TR/WCAG22)



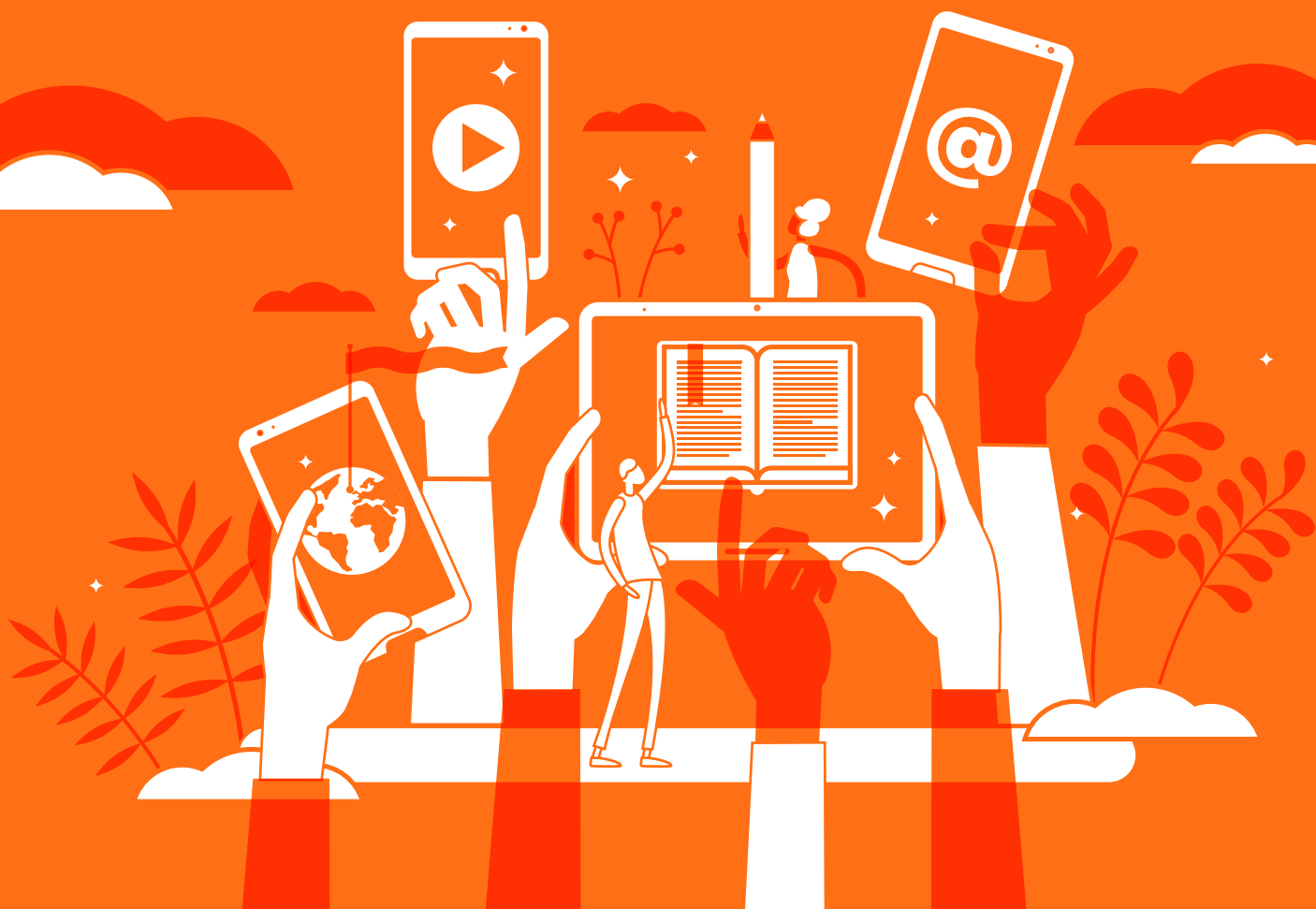


# DigitAll

Eduki digitalak  
sortzea

## 3.2

### EDUKI DIGITALAK INTEGRATZEA ETA BIRLANTZEA







Eduki digitalak  
sortzea

**B2 maila 3.2** Eduki digitala integratzea  
eta birlantzea

# Aurkezpenetako trantsizioak eta animazio aurreratuak





## Aurkezpenetako trantsizioak eta animazio aurreratuak

Aurkezpenak **eraginkorrak** izan daitezzen, diapositibetan egitura eta erritmoa ematen duten elementuak izan behar dituzte. Alde horretatik, aurkezpenak sortzeko tresnek hiru elementu nagusi jartzen dituzte erabiltzaileen eskura: **trantsizioak, animazioak eta ekintza-botoiak**.

### Trantsizioak

Aurkezpenetan diapositibaz aldatzean, ohikoa da diapositiba baten eta hurrengoaren artean mugimendu-efektuak ikusaraztea. Efektu horiek aurkezpenari eta edukiari jarraipena ematen diete.

Diapositiben **trantsizioa** zera da: diapositiba baten eta hurrengoaren arteko espazioan aplikatzen den **efektu bisuala**.

Oso erraza da PowerPoint aurkezpen batean diapositiba bati **trantsizio bat gehitzea**. Horretarako, erabiltzaileak urrats hauek egin beharko ditu:

- 1 | Aukeratu** trantsizioa zer diapositibari aplikatuko zaion.
- 2 | Egin klik **Trantsizioak**** fitxan. Aukeratu eskuragarri daudenen artetik zer trantsizio nahi duen. «Bat ere ez» trantsizioak diapositiba baten trantsizioa ezabatzea ahalbidetzen du.
- 3 | Aukeratu** trantsizioaren helbidea **Efektuen aukerak** botoian.
- 4 | Tarteak** taldean, doitu aukera gehiago hautatutako trantsizioaren gainean: trantsizioaren iraupena, aplikatu trantsizioa klik egitean edo denbora jakin baten buruan, gehitu soinu bat trantsizioari edo aplikatu trantsizioa aurkezpeneko diapositiba guztiei.





## Animazioak

Diapositiba baten barruko animazioak **A3C32B1D02**

«**Aurkezpenetako animazioak**» dokumentuan sartu ziren.

Diapositiba batean, baliteke objektu batek, hala nola testu, irudi, grafiko edo forma batek, animazio bat baino gehiago izatea.

Diapositiba batean animazio ugari daudenean, ohikoa da **animazio-panela** erabiltzea.



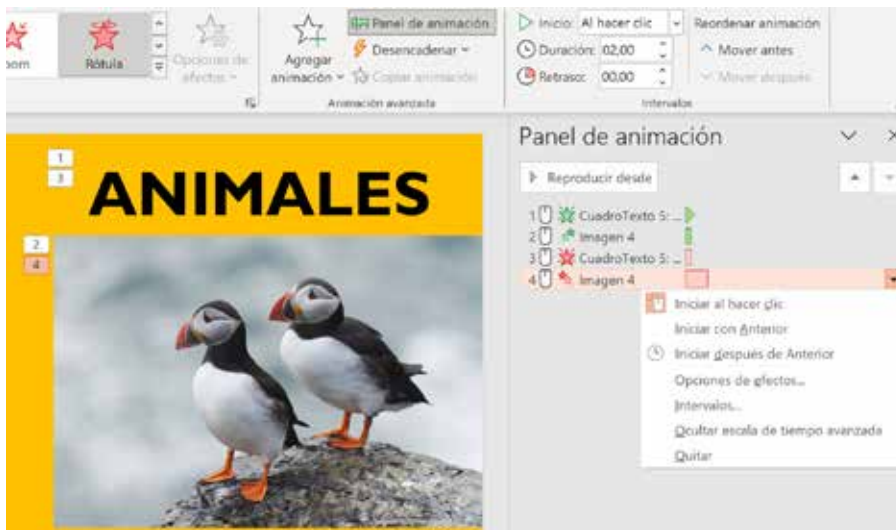
### AURKEZPENETAKO ANIMAZIOAK

Erreferentziako dokumentua:  
**A3C32B1D02**

**Animazio-panelean** aukera izango dugu diapositiba bati aplikatutako animazioen propietateak administratzeko.

Microsoft PowerPointeko **animazio-panelarekin lan egiteko**, urrats hauek egin behar dira (1. irudia):

- 1 | Hautatu** animatu nahi den objektua diapositibaren barruan.
- 2 | Animazioak** fitxan, hautatu **Animazio-panela** aukera.
- 3 |** Objektu baten gainean aplikatutako animazio bakoitza animazio-panelera **gehituko da**.
- 4 |** Animazio baten propietateak **administratzeko**, klik egin behar da dena delako animazioaren eskuinaldeko gezia.
  - a. Klik egitean:** animazioa erreproduzitzen du saguarekin klik egitean.
  - b. Hasi aurrekoarekin:** aurreko efektuarekin batera erreproduzitzen du animazioa.
  - c. Aurrekoaren ondoren hastea:** aurreko efektua erreproduzitu ondoren erreproduzitzen du animazioa.
  - d. Tarteak:** aukera ematen du animazioaren iraupena eta beste propietate batzuk doitzeko.



1. irudia. Animazio-panelaren aukera nagusiak.

## Ekintza-botoiak

Askotan, aurkezleak **diapositibekin elkarreraginean** aritu behar du aurkezpena egiten duen bitartean. Hori gertatuko da, adibidez, aurkezlearen fitxategi lokal bat irekitzeko, hiperesteka baten bidez webgune batera sartzeko, gailuko audio-pista bat erreproduzitzeko eta kanpoko bideo bat erreproduzitzeko. Ekintza-botoiak oso erabilgarriak dira aurkezlearen eta diapositibaren arteko komunikazioa eta interakzioa errazteko.

**Ekintza-botoia** objektu bat da; hain zuzen ere, **forma** bat; diapositiba batean txertatzen da, eta **ekintza** bat egiten du sagua pasatzean edo haren gainean klik egitean.

PowerPoint diapositiba baten barruan **ekintza-botoiak txertatzeko**, urrats hauek egin behar dira:

- 1 | Hautatu behar da** ekintza-botoia zer diapositibatan txertatu.
- 2 |** Klik egin behar da **Txertatu fitxan** eta, ondoren, **Formak** botoian.
- 3 |** Nahi den botoia **hautatu** behar da **Ekintza-botoiak** taldearen barruan. Forma marraztu behar da.



**4** | Elkarrizketa-koadro bat irekiko da. «**Egin klik saguarekin**» edo «**Pasatu sagua gainetik**» fitxa hautatu behar da, kontuan hartuta botoiari lotutako ekintza nola exekutatu nahi den.

**5** | **Botoiaren ekintza konfiguratu behar da** ekintzen zerrendaren barruan. Hauek dira garrantzitsuenak: hiperesteka bat sortzea (edozein diapositiba, kanpo-fitxategi edo web-helbidetara); programa bat exekutatzea eta soinu bat erreproduzitzea.

### **i** Informazio gehiago

PowerPointeko diapositiba-trantsizioak.

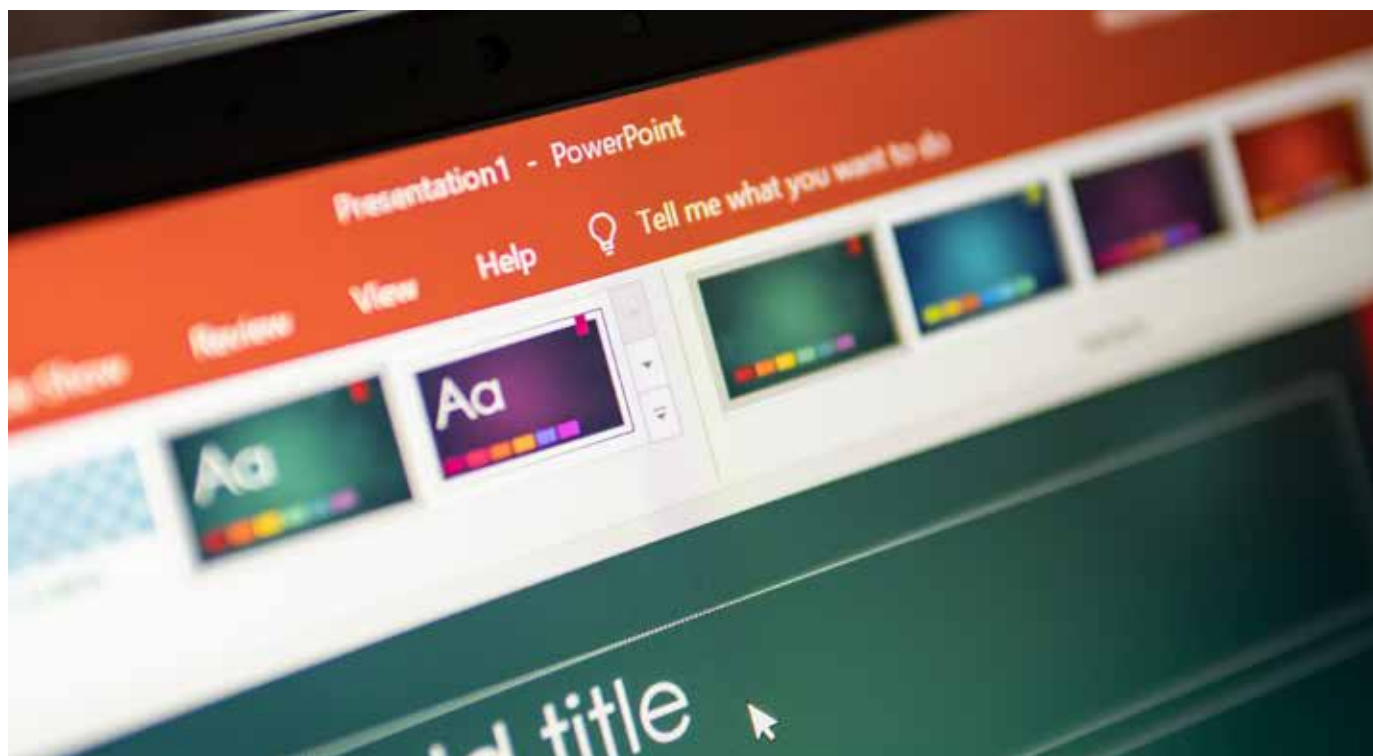
[e.digitall.org.es/transiciones-diapositivas](http://e.digitall.org.es/transiciones-diapositivas)

PowerPointeko animazio-panela.

[e.digitall.org.es/efectos-animaciones](http://e.digitall.org.es/efectos-animaciones)

PowerPointeko ekintza-botoiak.

[e.digitall.org.es/botones-accion](http://e.digitall.org.es/botones-accion)





Eduki digitalak  
sortzea

**B2 maila 3.2** Eduki digitala integratzea  
eta birlantzea

# Bideoa eta audioa birmasterizatzea





## Bideoa eta audioa birmasterizatzea

Aurreko mailetan aztertu den bezala, audio- eta bideo-osagaien kalitatea oso garrantzitsua da eduki digitalen kontsumitzaileek sentsazio ona izan dezaten. Hala ere, A3C32B2V02 *Eduki digitalak hobetzeko kalitate-analisia* bideoan ikusi dugun bezala, baliteke edukiek zenbait gabezia izatea, eta haien ondorioz bideoaren kalitatea murrizten da: bereizmen txikiko bideoa, enkoadraketa desegokia, soinua hautematen zaila izatea, etab. Erremasterizazio-teknikek arazo horiek konpontzen lagun dezaketela erakutsiko dugu jarraian.

Erremasterizazioa zera da: aurrez grabatutako soinuko edo ikus-entzunezko lan bat aldatzeko prozesua, jatorrizko grabazioaren kalitatea hobetzeko.



EDUKI DIGITALAK  
HOBETZEKO  
KALITATE-ANALISIA

[e.digitall.org.es/A3C32B2V02](https://e.digitall.org.es/A3C32B2V02)

### ⚠ ADI

Ez dira **erremasterizazioa** eta **eduki digitala birlantzea** nahasi behar; izan ere, erremasterizazio-prozesuan edukia ez da aldatzen beste bertsio bat lortzeko. Helburu nagusia beti izango da ikus-entzunezko piezaren soinu- edo irudi-kalitatea hobetzea.

### Audio-erremasterizazioa

Antzina, estudioek zintak erabiltzen zituzten egindako grabazioak gordetzeko. Gero, zinta horiek prozesatu eta erremasterizatu egiten ziren, **master** zeritzon artxiboa sortzeko. Ondoren, fitxategi hori erabiltzen zen banatzen ziren kopiak sortzeko. Zintan egindako grabazioen arazoa zera da: mugak dituzte izan dezaketen soinu-kalitateari dagokionez, eta zarata ere sortzen da formatu horretan grabatzean. Hori dela eta, gaur egun, grabazio zaharrok digitalki gordetzen direnean, erremasterizazio-prozesua egiten zaie, eta haren bitartez, **ezabatu egiten da grabazioen hondoko zarata, murriztu egiten dira zintak narriatzearen ondoriozko akatsak, eta hobetu egiten da tresnen soinua. A3C32B1V04 Adimen artifizialean oinarritutako tresnak, eduki digitala aldatzeko bideoan** ikusi genuen grabazioetako hondoko zarata nola ezabatu **Podcastle. ai** tresna erabiliz([e.digitall.org.es/podcastle](https://e.digitall.org.es/podcastle)).





Gaur egun, estudioko albumak grabatzean pista bereziak erabiltzen dituzte tresna bakoitzerako, eta, horren ondorioz, askotan 48 pista baino gehiagoko piezak sortzen dira. Jarraian, estudioko teknikariek pista horietako bakoitza nahi bezala nahasten dute, bi pista baino ez dituen fitxategia lortzeko (ezkerraldean eta eskuinaldean kokatutako bozgorailuenak), estereo deritzon formatuan. Instrumentu bakoitzerako pista desberdinak dituzenez, grabazioa erremasteriza daiteke, **zenbait instrumentu nabarmentzeko, instrumentuen espazioko pertzepzioa aldatzeko**, edo seinalearen maiztasunak aldatu eta **soinua ekualizatzeko**. Online tresna oso baliagarri bat: **Audioalter** ([audioalter.com](http://audioalter.com)).

Azkenik, audio-erremasterizazioaren beste erabilera nagusi bat hauxe da: **audioa konprimatzean galdutako audioaren definizioa berreskuratzea**, eraginkortasunez gordetzeko. Teknika hori oso interesgarria da etxean erabiltzeko; izan ere, audio-fitxategi komertzialek, 2000ko lehen hamarkadan, konpresio-tasa handiko formatuak erabiltzen zituzten fitxategiaren tamaina ahalik eta gehien murrizteko (soinuaren definizioa sakrifikatuta). Konpresio haren arrazoia garai hartako erreproduktzio-gailuen memoria-edukiera txikia zen. **Podcastle.ai** ([e.digitall.org.es/magic-dust](http://e.digitall.org.es/magic-dust)) tresnak grabazioen kalitatea handitzea ere ahalbidetzen du.

## Bideo-erremasterizazioa

Bideo-erremasterizazioa asko erabiltzen da gaur egun, batez ere zinema-industrian. Askotariko erabilerak ditu, aukera ematen baitu jatorrizko grabazioaren alderdi asko aldatzeko. Adibidez, **hartualdien planoak alda** daitezke askotariko enkoadraketak lortzeko eta horizontearekiko gaizki lerrokatutako kamera bat zuzentzeko. Erabil daitezke, halaber, **kolorearen saturazioa**, kontrastea eta distira **aldatzeko**, hartualdi ilunegiak eta gehiegizko esposizioa dutenak zuzentzeko.



ADIMEN ARTIFIZIALEAN  
OINARRITUTAKO  
TRESNAK, EDUKI  
DIGITALA ALDATZEKO

[e.digitall.org.es/A3C32B1V04](http://e.digitall.org.es/A3C32B1V04)

### Informazio gehiago

Irudiak gehiegizko esposizioa izaten du kameraren sentsoreak hartzen duen argi kantitatea handiegia denean, eta horren ondorioz, alderdirik argituenetako xehetasunak ezin dira garbi ikusi.





Bideo-erremasterizazioaren beste erabilera oso interesgarri bat hauxe da: **zuri-beltzeko grabazio zaharrak koloreztatzeko** tresnak aplikatzea. Adimen artifizialean oinarritutako tekniken bidez lortzen da «asmatzea» zein den irudi-zati bakoitzaren kolorea bideoko fotograma bakoitzean. Teknika horiek oso deigarriak dira, gure garaira ekartzen baitituzte oso film zaharren eszenak. Nabarmentzekoa da, adibidez, **DeOldify** ([deoldify.ai](https://deoldify.ai)) tresna; sare neuronal sakonak erabiltzen ditu irudiak eta bideoak koloreztatzeko. Teknika horren erabileraren adibide bat ikusiko dugu 1. irudian.



1. irudia. «Toffs and Toughs» by Jimmy Sime (1937). DeOldify erabiliz lortutako irudia.

Ohikoa da, bestalde, **bideoen bereizmena handitzeko** teknikak aplikatzea, bereizmen txikiko grabazioak FullHD edo 4K bereizmenetara egokituz. Horri esker, jatorrizko bideoa askoz ere bereizmen handiagoko pantailetan proiektatu ahal izango litzateke, emaitza pixelatu gabe. Teknika horien barruan, Topaz Labs-en **Gigapixel AI** ([topazlabs.com/gigapixel-ai](https://topazlabs.com/gigapixel-ai)) tresnak erabil daitezke, esaterako. Tresna honek neurona-sare sakonak erabiltzen ditu bideoen bereizmena handitzeko. Aplikazio hori, besteak beste, Netflix, Disney, Apple eta Warner Bros enpresek erabiltzen dute.



Bideo-erremasterizazioko tekniken beste erabilera batzuek **fotograma-interpolatzea** hartzen dute barne. Horri esker, bideoei leuntasuna ematen zaie, segundoko 24 edo 30 fotograma izatetik segundoko 60 fotograma izatera igarotzen baitira. **Smoother AI** ([e.digitall.org.es/smoothier-ai](https://e.digitall.org.es/smoothier-ai)) tresna nabarmendu behar dugu, teknika hori erraz aplikatzeko aukera ematen baitu. Ohikoa ere bada **zarata murrizteko** teknikak aplikatzea, galerak edo kalteak dituzten biltegiatze-sistemetan bideo zaharrak biltegiatzean sortutako zarata saihesteko. Zarata murrizteko tresna ugari daude, hala nola **Neat Video** ([neatvideo.com](https://neatvideo.com)).

Ikusten dugun bezala, aukera oso zabalak daude esparru honetan. Gainera, adimen artifizialeko teknikek azken urteotan izan duten gorakadak biderkatu egin ditu audioa eta bideoa erremasterizatzeko dauden tresnak eta aukerak. Irakurlea bultzatu nahi dugu bere bideoekin tresna horiek saiatzera, kalitate handiagoko materialak lortzeko eta bideo honetan zehaztutako gabeziak konpontzeko: **A3C32B2V02 Eduki digitalak hobetzeko kalitate-analisia**.



EDUKI DIGITALAK  
HOBETZEKO KALITATE-  
ANALISIA

[e.digitall.org.es/A3C32B2V02](https://e.digitall.org.es/A3C32B2V02)





# DigitAll

Eduki digitalak  
sortzea

## 3.3

**EGILE-ESKUBIDEAK  
ETA JABETZA  
INTELEKTUALEKO  
LIZENTZIAK**





Eduki digitalak  
sortzea

**B2 maila** 3.3 Egile-eskubideak eta jabetza  
intelektualeko lizentziak

# Obra bati copyrighta ematea





## Obra bati copyrighta ematea

Dokumentu honen edukiak jabetza intelektuala erregistratzeko mekanismoari buruzko ezagutzak ematen ditu.

Jabetza Intelektualaren Erregistroa bakarra da Estatuko lurralde osoan. Eskumenak estatuaren eta autonomia-erkidegoen artean banatuta daude. Horregatik, erregistro guztietan egindako inskripzioek balio bera dute Espainiako lurralde osoan.

Erregistro Zentrala Estatuko Administrazio Orokorraren parte da, eta Kultura eta Kirol Ministerioaren mende dago. Erregistro Zentralaren bulego bat dago Balear Uharte, Kanaria, Kantabria, Gaztela eta Leon, Gaztela-Mantxa eta Nafarroako autonomia-erkidegoetako probintzia-hiriburuetan, bai eta Ceuta eta Melilla hiri autonomoetan ere.

Lurralde-erregistroak daude autonomia-erkidego eta hiri autonomo hauetan: Andaluzia, Aragoi, Asturias, Katalunia, Valentziako Erkidegoa, Extremadura, Galizia, Errioxa, Madril, Murtzia eta Euskal Autonomia Erkidegoa, eta Ceuta eta Melillako hirietan.

Erregistro Zentralean eta haren mendeko probintzia-bulegoen bidez aurkezten dituzten inskripzio-eskaerak honela egin daitezke:

- **Aurrez aurre**, honako hauek aurkeztuz:
  - Inskripzioa eskatzeko inprimaki ofizialak.
  - Lanaren, emanaldiaren edo ekoizpenaren alea, inprimaki ofizialetan adierazitako moduan.
  - Honako hau da, kasuen arabera, legeriaren arabera eskatzen den ohiko dokumentazioa:
    - ◇ **Eskatzailearen eta egileen datuak:** NANaren fotokopia edo nortasuna egiaztatzeko agiria, atzerritarra bada. Adingabea edo ezindua izanez gero, familia-liburuaren fotokopia edo guraso-ahala, tutoretza edo kuradoretza egiaztatzen duen agiria aurkeztu beharko da.
    - ◇ **Ordezkariaren datuak, halakorik badago:** Ordezkaritza Ahalordea.
  - Dagokion tasa ordaindu izanaren frogagiria. Erregistroak inprimaki ofizialean likidazioa egin ondoren egiten da ordainketa.

### ⚠ ADI

Lurralde Erregistro bakoitzak bere jarraibideak ditu eskaerak aurkezteko. Hori dela eta, eskaera bat Lurralde Erregistro batean aurkezteko, harekin harremanetan jarri beharko dugu aldez aurretik, hori egiteko moduari buruzko informazioa jasotzeko.



- **Telematikoki, RePI** aplikazio telematikoa bidez ([e.digitall.org.es/repI](http://e.digitall.org.es/repI)); eskuragarri dago Kultura eta Kirol Ministerioaren egoitza elektronikoa. Eskatzaileak sinadura elektronikoa izan beharko du erregistro-prozedura telematikoki egiteko. Identifikazio- eta autentifikazio-prozesua egin ondoren, urrats hauek egin behar dira:
  - **Eskaera betetzea.** Horretarako, eskatzailearen eta ordezkatuaren datuak bete behar dira, halakorik badago. Erregistratu nahi den lanaren datuak ere sartu behar dira, adibidez: lan mota, obraren izenburua eta argitalpen- edo dibulgazio-data, baldin eta lan dibulgatu edo argitaratua bada. Gainera, egileari edo egileei buruzko datuak jaso behar dira, obra egile bati edo gehiagori badagokie. Azken kasu horretan, berariaz adierazi beharko da. Azkenik, eskaerari dokumentu gehigarriak erantsi ahal izango zaizkio, eskaeran bertan berariaz hala adierazten bada edo interesdunak beharrezkotzat jotzen badu.
  - **Tasak ordaintzea.** Lehenengo inskripzioa egiteko, tasa bat ordaindu behar da eskaera sinatu eta bidali aurretik. Ordainketa telematikoki edo bankuan egin ahal izango da.
  - **Eskaera sinatzea eta bidaltzea.** Aplikazioan beharrezkoak diren datuak sartu ondoren, «Bidali» eta, ondoren, «Sinatu» egin beharko da, sinadura elektronikoa erabiliz.

Izapidetze telematikoa dena ondo joan bada, eskaera behar bezala erregistratu dela adieraziko da, eta eskaeran sartutako datu guztiak agertuko dira, barne hartuta sortutako espediente-zenbakia, erregistro-zenbakia eta data.

Gainera, lanaren identifikazio-alearen kopia elektronikoa bat aurkeztu beharko da. Horretarako, lan bat erregistratu ondoren, sistemak web-helbide bat eskainiko du identifikazio-alearen kopia aurkezteko.

#### ADI

Erregistro bat egiten denean, LAN mota argi eta garbi adierazi behar da. Lan mota eta azpimota hautatu behar dira. Mota batzuetarako, aukera dago lanaren xehetasun espezifikoa zehazteko. Xehetasun horiek ez dira sartu behar obra mota hauetarako: koreografia eta pantomima; datu-baseak; ordenagailu-programak; web-orrria eta multimedia; argazki hutsa; datu-base baten gaineko *sui generis* eskubidea.



**OHARRA**

Egile bat baino gehiago dagoenean, «Egileak» adieraztean adierazi beharko da bakoitzari zer egiletza mota dagokion atal honetan adierazitako egiletza motetatik.

**i Informazio gehiago****Lurralde-erregistroen mendeko bulegoak:**

[e.digitall.org.es/registro-territorial](https://e.digitall.org.es/registro-territorial)

**Erregistro Zentralaren mendeko bulegoak:**

[e.digitall.org.es/registro-central](https://e.digitall.org.es/registro-central)

**Estatuko legeria:**

- 1/1996 Legegintzako Errege Dekretua, apirilaren 12koa, Jabetza Intelektualari buruzko Legearen testu bategina onartzen duena, arloan indarrean dauden lege-xedapenak erregularizatu, argitu eta harmonizatzen dituena ([e.digitall.org.es/se 8930](https://e.digitall.org.es/se-8930)).
- 281/2003 Errege Dekretua, martxoaren 7koa, Jabetza Intelektualaren Erregistro Orokorraren Erregelamendua onartzen duena ([e.digitall.org.es/boe-6247](https://e.digitall.org.es/boe-6247)).
- **Erkidegoko legeria:** [e.digitall.org.es/legislacion-autonomica](https://e.digitall.org.es/legislacion-autonomica)





Eduki digitalak  
sortzea

**B2 maila 3.3** Egile-eskubideak eta jabetza  
intelektualeko lizentziak

**Software librea,  
freeware,  
shareware,  
adwarea**







## Software librea, freewarea, sharewarea, adwarea

Softwarea ere bada babestu daitekeen eta babestu behar den produktu intelektual bat. Software baliotsu eta erabilgarri bat garatu duen programatzaile batek azter dezake bere softwarearekin eta programa informatikoarekin zer egin, hirugarrenek erabili ahal izan dezaten. Beste pertsona batek erabiltzeko baimen hori **softwarearen lizentziaren** bidez formalizatzen da, eta barne hartzen du, orobat, kopiatu, aldatu, banatu eta/edo saltzea, zuk ezarri nahi dituzun muga eta baldintzekin. Lizentzia hori kontratu bat da, eta haren bidez jabetza intelektualak babestutako elementu baten (adibidez, gure softwarea) titularrak hura erabiltzeko baimena ematen dio beste pertsona bati, hirugarren bati, klausuletan ezarritako baldintzekin. Baina programaren titulartasuna saldu nahi badu, softwarea LAGATZEKO kontratua sinatu beharko du. Softwarearen titularra edo egilea lizentzia-emailea da, eta erabiltzailea, berriz, lizentzia-hartzailea.

**Zer motako lizentzia aplika dakioke softwareari?** Kontuan hartuta lizentzia-hartzaileei zer baldintza edo baimen ezarri nahi zaizkien, eta **zer plataformatan jarri nahi den**, honako lizentzia mota hauek bereizten dira:

- **GPL lizentzia (GNU LGPL).** Software librea da, eta modulu ez-libreekin lotzeko aukera ematen du. Erabilgarri eta eskuragarri egon behar du, kopiak mugarik gabe egiteko eta eskatzen duen edonorentzat. Lehen bertsioan, «GNUko liburutegietarako lizentzia publiko orokorra» esaten zitzaion.
- **BSD lizentzia.** Lizentzia permisiboa da, ia ez baitu baldintzarik ezartzen erabiltzaileek egin dezaketinari dagokionez: softwarea sal daiteke, eta ez da nahitaezkoa iturburu-kodea sartzea.
- **MPL lizentziak.** Halakoak dira Mozilla Firefox, Mozilla Thunderbird eta Mozillaren beste software gehienak, eta ez dira aurrekoak bezain permisiboagoak.





- **Debian lizentzia** (Debian Free Software Guidelines-DFSG). Hortik dator Open Source Initiative lizentzia. Ezartzen du iturburu-kodea argitaratzeko eskakizuna, birbanaketa askea (baita jatorrizkoaren lizentzia beraren ondoriozko lanarena ere), eta ezin duela pertsona eta talderik diskriminatu, ez softwareak, ez hura erabiltzeko ezein moduk ere.
- **Open Source Initiative lizentzia**. Lizentziarik ezagunenetako eta erabilienetako bat da; iturburu irekikoa edo kode irekikoa ere esaten zaio, eta zera ahalbidetzen du: banaketa askea (iturburu-kodearena ere bai), hura aldatzea, garapen eratorriak eta jatorrizko softwarearen baldintza berberetan birbanatzea. Gainera, Open Sourcek egilearen iturburu-kodearen osotasuna ezartzen du, eta ez duela ezen pertsona edo talde diskriminatu behar, ez eta softwarea erabiltzea mugatu domeinu- edo jardueraren eremuetara ere, eta neutrala izan behar du teknologiari dagokionez.
- **X.org lizentzia** (X Windows System). X Partzuergoak banatzen du, haren garapena kontrolatzen duen erakunde gisa, eta Microsoften tankerako sistema grafikoen alternatiba gisa sortutako Linux eta UNIX sistema eragileetan erabiltzen da. Software librea da, baina copyleft-arekin bat egin gabe.
- **Freeware lizentzia**. Softwarea askatasunez eta doan erabiltzeko baimena ematen du, baina, oro har, partikularrei, eta ez enpresa edo erakunde ofizialei. **Donationware lizentzia** antzekoa da, baina dohaintza bat eskatzen du trukean, nahitaezkoa ez bada ere. **Postcardware lizentzia** antzekoa da, baina postal bat bidaltzeko eskatzen du, dohaintzaren orde.
- **Proba-lizentziak edo sharewarea**. Aurretik erregistratzea eskatzen dute, eta programa bat erabiltzeko baimena ematen dute, erabiltzaileak hura ebaluatu eta ondoren eros dezan. Demo lizentzia programa bat ebaluatzeko saioa da. Oso ezagunak dira joko-probetan, adibidez.

Azkenik, ez badugu nahi softwarearen erabiltzaileak hura aldatu, partekatu edo banatzea sortzailearen baimena eduki ezean, **lizentzia pribatiboa izango da**, hala nola Appleren macOS sistema eragilearen edo Windows lizentziak.



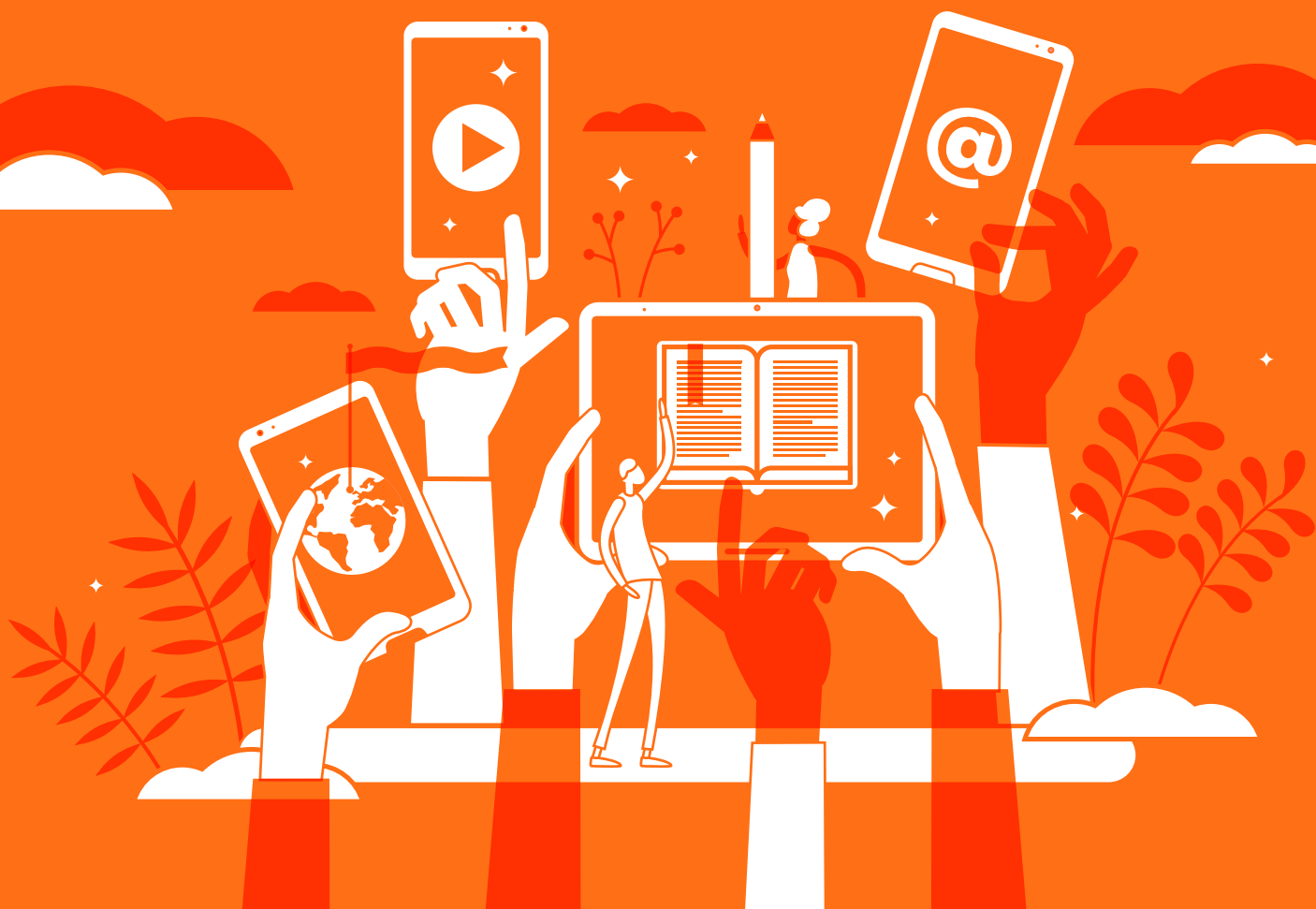


# DigitAll

Eduki digitalak  
sortzea

## 3.4

### PROGRAMAZIOA





Eduki digitalak  
sortzea

**B2 maila 3.4** Programazioa

# Python lengoiaren alderdi orokorrak





# Python lengoaiaren alderdi orokorrak

## Sarrera

Programazioaren munduan ikasteko bideari ekitean hartu beharreko erabakietako bat hauxe da: «zer programazio-lengoaia erabili». Python aukera ona da zenbait arrazoiengatik: lengoaia moldakorra, malgua da, eta ikaste-kurba nahiko leuna du beste lengoaia batzuekin alderatuta; plataforma anitzekoa eta erabat doakoa da.

Atal honetan Python lengoaiaren alderdi lexikoen, sintaktikoen eta semantikoen hastapenak aurkituko dituzu. Python-eko programa baten egitura orokorra aurkeztuko dugu, eta ikasleari aurkeztu eta azalduko dizkiogu Python kodean ikusi ohi diren kontzeptu-mailak. Azkenik, Python-eko indentazioaren kontzeptua izango dugu hizpide, baita kontrol-jarraibide edo -egituren irismena mugatzeko giltza/sententziarik eza ere.



## Python-en alderdi lexikoak, sintaktikoak eta semantikoak

Hizkuntza bat erabiltzen dugunean, ematen dituen tresnak erabiltzen ditugu, hala nola alfabetoa, hiztegia (hura osatzen duten hitzen multzoa eta haien esanahia barne) eta gramatika, hau da, hitzek zer ordenatan agertu behar duten zehazten duten arauen multzoa. Programazio-lengoaia baten kasuan, antzeko elementuak daude erabilgarri.

Python-en eta halako programazio-lengoaien **lexikoak** zehazten du zer sinbolo eta hitz erabil ditzakegun programetan jasotako diren sententziak eraikitzeke. Python-en **azterketa lexikoak** fitxategi bat (idatzi dugun programarekin) *tokenetan* banakatzen du, lengoaia horretan programak idazteko modua zuzentzen duten oinarriko arauak jarraikiz.

Jarraian, Python-eko **sintaxi-analizatzaileak** programaren egitura sintaktikoa aztertzen du. Lexiko-analizatzailearen irteera erabiltzen du sintaxi abstraktuko zuhaitz bat (AST) eraikitzeke; zuhaitzak sarrera-programaren egitura sintaktikoa irudikatzen du. Egitura horrekin, zeregin batzuk egin daitezke: akats sintaktikoak detektatzea eta objektu-kodea sortzea.



**Analizatzaile semantikoak**, azkenik, sintaxi-analizatzaileak sortutako irteera erabiltzen du hizkuntzaren semantikari eragiten dioten akatsak detektatzeko, hala nola motak egiaztatzea, aldagai eta metodoetarako sarbidea, balio-tartea, esleipen baliodunak, definitu gabeko aldagaiak eta legez kanpoko eragiketak, besteak beste.

Python-eko programa bat exekutatzeko, programak ez du akats lexikorik, sintaktikorik eta semantikorik izan behar.

## Python-eko programa baten egitura orokorra

Python-eko programa batek egitura orokor hau izan ohi du:

**1 | Iruzkinak:** testu-lerroak dira, eta programaren helburua eta nola funtzionatzen duen azaltzen dute. Aukerakoa da, programatzailearen irizpidearen arabera, eta Python-eko interpretatzaileak ez du kontuan hartzen. Horrenbestez, ez du inolako akatsik sortzen.

```
# Hau lerro bateko iruzkin bat da
"""
Hau zenbait lerroko iruzkin bat da,
komatxo hirukoitzak erabilia
"""
```

**2 | Inportazio-deklarazioa:** kode-lerroak dira, eta aukera ematen dute programarentzat erabilgarriak diren funtzioak eta klaseak dituzten moduluak edo liburutegiak inportatzeko.

```
import math
print (math.pi) # Pi balioa inprimatzen du (3.141592653589793)
```

**3 | Funtzioen definizioa:** zeregin espezifiko bat egiten duten kode-blokeak dira, eta programan behin baino gehiagotan dei daitezke. Funtzioak definitzeko, «def» gako-hitza erabiltzen da, eta, ondoren, funtzioaren izena eta parentesi arteko parametroak.

```
def batuketa(a, b):
    emaitza = a + b
    return emaitza
```



**4 | Aldagai-deklarazioa:** kode-lerroak dira, eta programan erabiliko diren aldagaiei balioak deklaratzeko eta esleitzen dizkiete. Aldagai horiek sinpleak dira (balio bakarra biltegitzen dute) edo datu-egitura konplexuagoak, hala nola arrayak, bektoreak, zerrendak eta hiztegiak, besteak beste.

```
# Esleitu balioak aldagaiei
zenbakia = 10
izena = "Jose Manuel"
es_aktibo = True
```

**5 | Bloke nagusia:** programaren gorputz nagusia da, eta bertan eragiketak egiten dira eta aurrez definitutako funtzioei deitzen zaie. Bloke nagusian zein funtzioetan, fluxua kontrolatzeko egiturak erabili ohi dira, hala nola baldintzazkoak eta begiztak.

## Python-eko indentazioa

Python-eko **indentazioa** kode-lerro baten hasieran zuriuneak edo tabulazioak erabiltzea da, programaren hierarkia eta egitura adierazteko. Python-en, indentazioak erabiltzen dira fluxua kontrolatzeko egituren (baldintzazkoak eta begiztak) eta funtzioen irismena adierazteko.

Python **lengoaiaren ezaugarri bereizgarri** bat da giltzarik edo sententziarik ez egotea kontrol-jarraibide edo -egituren irismena mugatzeko. Giltzak irismena mugatzeko erabili beharrean, beste programazio-lengoaia askotan egiten den bezala, indentazioak erabiltzen dira. Jarraian, adibide batean ikusiko dugu, lehenik, kodearen zati xume bat, giltzak blokea mugatzeko erabilia, eta bigarrenik, Python-eko haren balioak, indentazioak erabilia:

```
if x > 0: {
    print ("x zero baino handiagoa da")
}
```

Python-eko kodea:

```
if x > 0:
    print ("x zero baino handiagoa da")
```

### ⚠ ADI

Python-eko indentazioak ez dira alderdi estetiko hutsa, eta, errespetatzen ez badira, sintaxi-akatsak egongo dira. Python-en ezaugarri berezi horren ondorioz kodea irakurterazagoa eta ulertterazagoa izaten da, eta horrek programen hierarkia eta egitura argi eta garbi ikustea errazten du.



Eduki digitalak  
sortzea

**B2 maila** 3.4 Programazioa

# Python-eko datu-motak







# Python-eko datu-motak

## Sarrera

Beste edozein programazio-lengoaian bezala, Python-ek datuak aldagaietan biltegitratzea onartzen du, eta aldagai bakoitzak datu-mota bat izan dezake lotuta. Hurrengo atalean, Python-ek eskaintzen dituen oinarrizko datu-moten ikuspegi orokorra emango dugu, baita haietako bakoitzaren erabileraren adibide bat ere.

## Python-eko oinarrizko datu-motak

**1 | Boolean Type (bool):** Python-en, 'bool' datu-motak balio logiko bat adierazten du, hau da, egiazkoa (True) edo gezurrezkoa (False). Horrelako datuak baldintzapeko adierazpenetan eta eragiketa logikoetan erabili ohi dira. Hona hemen adibide xume bat:

```
x = True
y = False
if x:
    print("x egiazkoa da")
else:
    print("x gezurrezkoa da")
```

**2 | Text Type (str):** Python-eko 'str' datu-motak karaktere-sekuentzia bat adierazten du (testua) eta komatxo bakun edo bikoitzen artean definitzen da. Jarraian, adibide simple bat erakusten da; bertan testu-kate bat sortzen da eta pantailan erakusten da:

```
mezua = "Kaixo, Mundua"
print(mezua)
```

**3 | Numeric types (int, float):** Python-ek zenbakizko datu-motak irudikatzeko aukera ematen du. Ohikoenak hauek dira: 'int' (osoko zenbakiak adieraztea ahalbidetzen du), eta 'float' (zenbakiak dezimalekin adierazten ditu). Datu-mota horri esker eragiketa matematikoak egin daitezke, hala nola batuketak, kenketak, biderketak eta zatiketak. Hona hemen bi datu-mota horien erabileraren adibide xume batzuk:





```
num1 = 5
num2 = 10
emaitza = num1 + num2
print (emaitza)
```

```
num1 = 5.5
num2 = 10.1
emaitza = num1 + num2
print (emaitza)
```

Osoko zenbaki baten eta koma mugikorreko zenbaki baten arteko eragiketa bat eginez gero, emaitza koma mugikorreko zenbaki bat izango da.

**4 | Sequence Types (list, tuple, range):** Python-eko sekuentziako datu-motak, izenak berak adierazten duen bezala, elementu-sekuentziak dira. Mota horiek *list*, *tuple* eta *range* barne hartzen dituzte.

- **List** sekuentzia motak era guztietako elementuen sekuentzia ordenatua biltegitratzea ahalbidetzen du. Zerrendak kako zuzenen [] artean definitzen dira, eta sortu ondoren alda daitezke.

```
frutak = ["sagarra", "banana", "udarea"]
print(frutak)
```

- **Tuple:** **list**en antzekoa da, baina tupla bat ezin da aldatu sortu ondoren. Tuplak parentesi artean definitzen dira ().

```
aste_egunak = ("astelehena", "asteartea", "asteazkena")
print(aste_egunak)
```

- **Range** datu-motak osoko zenbakien sekuentzia bat sortzeko aukera ematen du, eta maiz erabiltzen da fluxua kontrolatzeko egituretan; adibidez, *for* begiztak. Hurrengo adibideak letik 10era bitarteko osoko zenbakien sekuentzia sortu eta inprimatuko du.

```
zenbakiak = range (10)
print(list(zenbakiak))
```

Kodearen azken lerroan, «zenbakiak» tartea zerrenda bihurtzen da, sortutako elementuak inprimatzeko. Biltegitratu ere egin daitezke, manipulatzeko, hala nahi izanez gero. Hona hemen adibide xume bat:

```
tartea = range (1, 6)
zerrenda = list(tartea)
```



**5 | Mapping Type (dict):** Python-en, mappinga objektu mota bat da, eta elementu-bilduma bat gako-balio pareetan biltegitratzen du, eta haietan, gako bakoitza balio bati lotuta dago. Beste programazio-lengoaia batzuetan, objektu mota horri «hiztegi» edo «hash taula» esaten zaio batzuetan. Gakoak bakarrak (ezin ditugu gako berak erabili bikote desberdinetarako) eta aldaezinak (ezin dira aldatu) dira, eta gako bakoitza balio bati lotuta dago. Gako horiek erabiltzen dira, hain zuzen ere, lotuta dauden balioetara iristeko. Adibidez, hiztegi bat sor genezake, eta bertan gakoak herrialdeen izenak izatea, eta lotutako balioak, berriz, herrialde horietako hiriburuak. Horrela, hiriburuaren izena erraz berreskuratuko dugu herrialdearen izenetik abiatuta:

```
hiriburuak = {'Espainia': 'Madril', 'Frantzia': 'Paris', 'Alemania': 'Berlin', 'Italia': 'Erroma'}
```

Azpiratu behar da balio gisa askotariko elementuak erabil ditzakegula, hala nola zerrendak, osoko zenbakiak eta beste edozein objektu aldakor.

Hona hemen gakoak eta lotutako balioak pantailan erakusteko modu erraz bat:

```
print(list(hiriburuak.keys()))  
print(list(hiriburuak.values()))
```

**6 | Set Types (set):** Python-en, set datu-mota bat da, eta aukera ematen du elementu bakarren eta ordenatu gabeen bilduma bat biltegitratzeko. Zerrenda edo hiztegi baten antzekoa da, baina gakorik gabe eta elementu bakarrekin. Setak sortzeko, {} giltzak edo set() funtzioa () erabiltzen dira. Adibidez, honela defini genezake, gehienetan, aldatu behar ez diren balioak dituzten koloreen multzoa:

```
koloreak = {'gorria', 'berdea', 'urdina'}  
koloreak = set(['gorria', 'berdea', 'urdina'])
```

Datu mota horrek eragiketa matematiko-logikoak jasaten ditu, hala nola bilketa, ebaketa eta kenketa, besteak beste.

### Informazio gehiago

Python-eko datu-motei buruz gehiago jakiteko, kontsulta ezazu gaztelaniazko dokumentazio ofiziala: [docs.python.org/es](https://docs.python.org/es)



Eduki digitalak  
sortzea

**B2 maila** 3.4 Programazioa

# Python-eko aldagaien eta konstanteen alderdi orokorrak





## Python-eko aldagaien eta konstanteen alderdi orokorrak

Aldagaiak, hitzarmen bidez, letra xehez izendatzen dira, errazago identifikatze aldera. Gainera, haien izenak edo identifikatzaileak letra edo azpimarra batez (`_`) hasi behar dira, eta ondoren, letra, zenbaki edo azpimarra gehiago jarriko dira. Identifikatzaileek ezin dute zuriunerik izan, eta ezin da erabili Python-en lehendik dauden funtzioen izenik. Identifikatzaile bat zenbait hitzez osatuta badago, hitzarmena da haiek gidoez banantzea.

Aldagai bat definitutzat jotzen da balio bat esleitu zaionean, eta, beraz, hasita dagoenean. Hona hemen ondo definitutako aldagaien adibide bat:

```
prezioa_ordu = 5
egunak = 10
kilometroak = 60
```

Ez dago konstanterik bere horretan Python-en. Horregatik, programatzaileek aldagai bat baino ez dute aitortzen, hasierako balio bat esleitzen diote eta espero dute balio hori inork ez aldatzea. Programatzaileen arteko konbentzio bat dago konstanteak izendatzeko, haiek erraz identifikatzeko eta ez direla aldatu behar gogorarazteko. Aldagaiak ez bezala, eta hartu ohi den hitzarmenaren arabera, haien izen osoa letra larriz idatzi behar da.

Hona hemen ondo definitutako konstanteen adibide bat:

```
URTE_EGUNAK = 365
ORDU_MINUTUAK = 60
ORDU_SEGUNDOA = 60
```



## Esleipen anizkoitza

Aldagai bat definitzeko, “=” operadorearekin hasieratu behar da, lehen ikusi dugun bezala. Gainera, aldagaiak batera hasieratzeko moduak daude. Adibidez, balio bat aldagai bati baino gehiagori esleitu ahal zaie aldi berean:

```
a = b = c = 1 # Hasieratu a, b eta c, 1 balioarekin
```

Aldagai bat baino gehiago aldi berean baina balio desberdinekin hasieratu nahi baditugu, sintaxia honelakoa izango da:

```
a, b, c = 10, 50, 75 # Hasieratu a 10ekin, b 50ekin eta c 75ekin
```





Eduki digitalak  
sortzea

**B2 maila** 3.4 Programazioa

# Python-eko funtzioak eta parametro- pasatzeak





## Python-eko funtzioak eta parametro-pasatzeak

Python programazio-lengoaia moldakorra da, kodearen irakurgarritasuna eta sintaxi sinplea ditu ezaugarri.

Programazio-lengoaia guztietan, baita Python-en ere, **funtzioen** definizioa funtsezkoa da. Ondoren, Python-en funtzioak aztertuko ditugu eta xehetasunez azalduko dugu nola sortu eta erabiltzen diren. Gainera, argumentuen eta parametroen arteko aldea aztertuko dugu, eta parametro lehenetsien kontzeptua aurkeztuko dugu.

### Funtzioak sortzea eta erabiltzea

Python-en, funtzio bat kode berrerabilgarriko bloke bat da, eta ekintza espezifiko bat egiten du. Funtzioek kodea bloke logikoetan antolatzeko modua ematen dute, eta horrek argitasun handiagoa eta mantentzeko erraztasuna ahalbidetzen du. Funtzioak sortzen dira **def** gako-hitza erabiliz, eta, ondoren, funtzioaren izena eta parentesiak jarritz. Adibidez:

```
def nire_funtzioa():  
    print("Kaixo, mundua!")
```

Definitu ondoren, funtzioa deitu daiteke, exekutatzeko, izena eta, haren ondoren, parentesiak erabiliz:

```
nire_funtzioa()
```

Funtzio hori exekutatu gero, pantailan «Kaixo, mundua!» testua erakutsiko da. Ikusten dugun bezala, funtzio batek beste adierazpen asko ere izan ditzake osagai, eta ordenari jarraikiz exekutatu dira funtzioa deitzen denean. Kasu honetan, definitu den funtzioak adierazpen bakar bat exekutatu du; azken hori **print** izeneko beste funtzio bat deitzeak osatua dago, testua pantailan erakusteko.







## Argumentuak eta parametroak

Argumentu eta parametro terminoak askotan bereizi gabe erabiltzen dira programazioan, baina funtsezko desberdintasun bat dago. Parametro bat funtzioaren definizioko aldagai bat da, eta argumentu bat, berriz, deitzean funtzioari pasatzen zaion balioa da.

Adibidez, hurrengo funtzioan, **izena** parametro bat da:

```
def agurtu(izena):  
    print(f"Kaixo, {izena}!")
```

Eta funtzioa deitzean, pasatzen zaion balioa, **Pedro**, argumentu bat da:

```
agurtu("Pedro")
```

Baina, zer gertatuko litzateke adibidearen funtzioa argumenturik gabe deituz gero? Adibidez, honela:

```
agurtu()
```

Kasu honetan, **izena** parametroan argumentu baten zain dagoen funtzio bat definitu da; horrenbestez, funtzioa deitzen bada argumentu hori pasatu gabe, Python-eko interpretatzaileak honako errore-irteera honekin erantzungo du pantailan:

```
Traceback (most recent call last):  
  File "<stdin>", line 1, in <module>  
TypeError: saludar() missing 1 required positional argument: 'izena'
```

Erroreak adierazten duen bezala, **agurtu()** funtzioaren deiarri argumentu bat falta zaio, hark deitzea espero duen bezala.

### OHARRA

Python-ek aukera ematen du testu-kateetan aldagaiak zuzenean txertatzeko, aurreko adibideko **print()** funtzioa deitzean erakusten den bezala. Horretarako, katea **f** karakterearekin aurrez finkatu behar da, eta testu-katean sartu nahi ditugun aldagaiak edo adierazpenak giltzen artean **{}** bildu behar dira.



## Parametro lehenetsiak

Amaitzeko, Python-ek aukera ematen du funtzioetan balio lehenetsiak dituzten parametroak definitzeko. Horrek esan nahi du ezen, funtzioa deitzean parametro horretarako baliorik eman ezean, funtzioa definitzean adierazitako balio lehenetsia erabiliko dela. Adibidez:

```
def agurtu(izena="Mundua"):  
    print(f"Kaixo, {izena}!")
```

Kasu honetan, funtzioa `agurtu()` bidez inolako argumenturik gabe deitzen bada, «Kaixo, Mundua!» testua agertuko da pantailan, izenaren parametroak «Mundua» testu-katea baitu lehenetsita. Baina argumentu bat pasatzen bazaio, hala nola `agurtu("Pedro")` funtzioan, haren ordean «Kaixo, Pedro!» erakutsiko da. Teknika hau oso erabilgarria da funtzioei portaera lehenetsia emateko eta, horrela, aurreko atalean erakutsitakoa bezalako errorerik gerta ez dadin. Esate baterako, hona hemen erabilera-kasu batzuk:

- **Funtzioei portaera malgua ematea:** parametro lehenetsiek aukera ematen dute funtzioak modu desberdinean jokatzeko, ematen den informazio kantitatearen arabera. Adibidez, `berretura(berrekizuna, berretzailea=2)` funtzioak zenbaki baten karratua kalkulatu du argumentu bakarra ematen bada, baina zehaztutako berretura kalkulatu du bi argumentu emanez gero.
- **Funtzioaren deiak sinplifikatzea:** parametro asko behar dituzten funtzioetan, balio lehenetsiak emateak funtzioa deitzea sinplifikatu du, parametro kritiko edo nahitaezko gutxi batzuetarako baino ez baitira argumentuak behar.



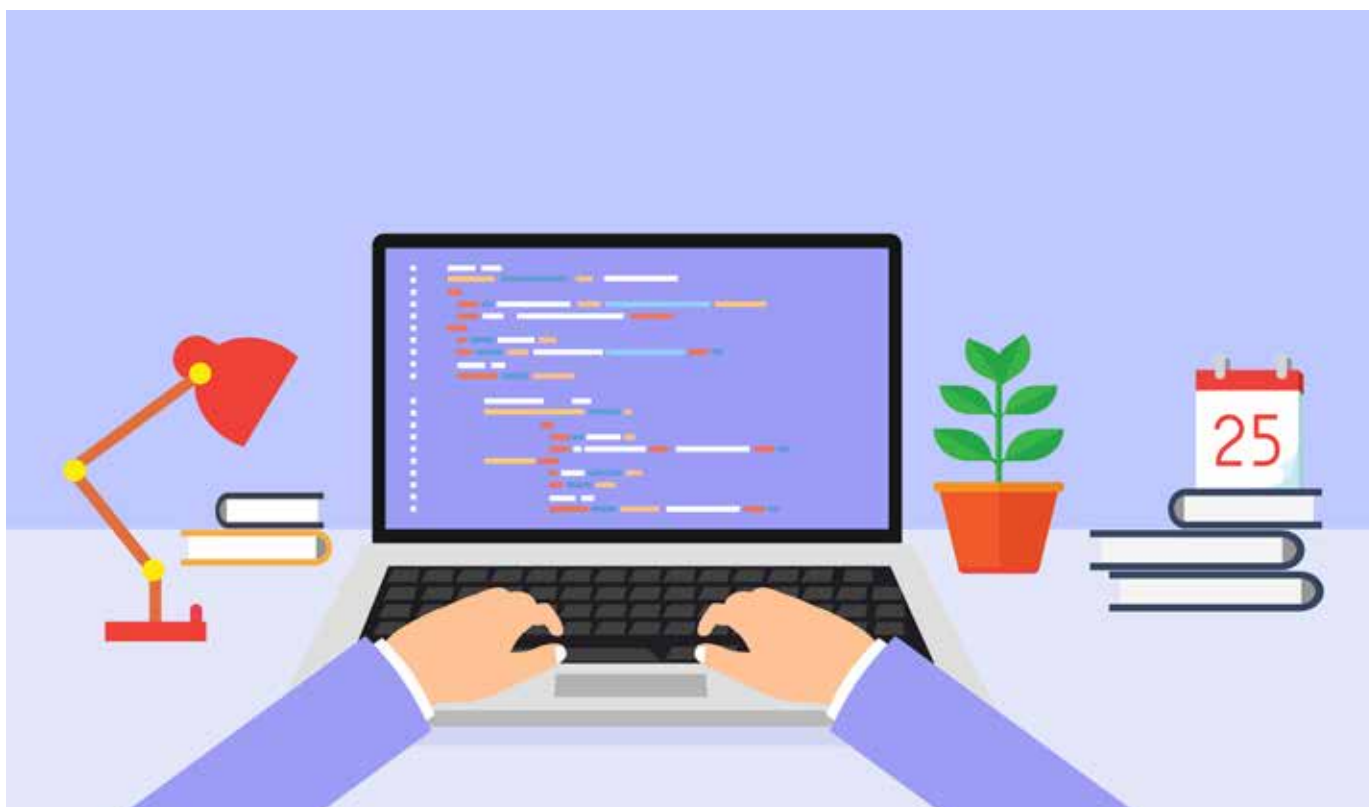
## Ondorioa

Python-en funtzioak erabiltzeak askotariko onurak dakartza. Lehenik eta behin, kodea berrerabiltzeko aukera ematen dute, eta horrek eraginkortasuna eta koherentzia hobetzen ditu; kode bera programaren zati desberdinetan errepikatu beharrik gabe, funtzio bat behin definitu daiteke, eta gero deituko dugu eragiketa espezifiko hori egin behar dugun bakoitzean.

Gainera, Python-eko funtzioek kodea antolatzen eta irakurtzen laguntzen dute; funtzioetan erlazionatutako kode-segmentuak elkartzen direnez, programak errazago ulertzen eta mantentzen dira. Onura hori are handiagoa da Python-ek gaitasuna baitu parametro lehenetsiak definitzeko eta funtzioei beren izenaren arabera argumentuak emateko, eta horrek hobetu egiten du funtzioekiko interakzioen malgutasuna eta argitasuna.

### **i** Informazio gehiago

Python-eko funtzioei buruz gehiago jakiteko, kontsulta ezazu gaztelaniazko dokumentazio ofiziala: [e.digitall.org.es/definir-funciones](https://e.digitall.org.es/definir-funciones)





Eduki digitalak  
sortzea

**B2 maila 3.4** Programazioa

# Hiztegiak: definizioa eta erabilera nagusiak





## Hiztegiak: definizioa eta erabilera nagusiak

Python-eko hiztegiak datu-egitura egokia dira, halaber, arazo jakin batzuetarako. Zerrenda batean elementuak atzitzen dira zerrendaren barruan duten posizioaren bidez; hiztegietan, berriz, gako baten bidez atzitzen dira. Puntu honetan, garrantzitsua da azpimarratzea Python-eko hiztegi baten gakoek aldaezinak izan behar dutela (*int*, *float*, *decimal*, *bool*, *string*, *tuple* eta *range*). Normalean, hiztegi baten gakoen mota *int* edo *string* izango da.

Python-en hiztegiak erabiltzea aztertzeke orduan, klabe-balio pareen multzotzat har daitezke. Horrela, hiztegi baten gakoek bakarrak izan behar dute. Adibidez, hiztegi bat erabiltzen badugu pertsona batzuen informazioa maneiatzeko, eta gakoa NANA bada, ezin izango ditugu hiztegian bi pertsona gorde NAN berarekin.

Irakurleak puntu honetan imajinatuko duen bezala, hiztegi batekin gehienbat bi eragiketa hauek erabiltzen dira: i) beste balio bat eta hari dagokion gakoa gehitzea, eta ii) hiztegiko balio bat ateratzea, hura indexatzeko aukera ematen duen gakoa erabiliz. Hiztegi bateko elementuak ere ezaba daitezke, jakina.

Hurrengo kode-zatiak hiztegi baten oinarritzko erabileraren adibide bat erakusten du, Python-en interpretatzaile interaktiboaren bitartez:

```
>>> d = {}
>>> d['David'] = 39
>>> d['Alberto'] = 34
>>> d
{'David': 39, 'Alberto': 34}
>>> d['David']
39
>>> del d['David']
>>> d
{'Alberto': 34}
```





Hiztegi bateko elementuen gainean iteratzeko aukera dago, eta hiztegiako elementu bakoitzaren gakoa eta balioa aldi berean ere berreskura daitezke. Hurrengo adibideak ezaugarri hori erakusten du, eta, aldi berean, Python-en hiztegi bat sortzeko beste modu laburtuago bat azaltzen du:

```
>>> d= {'David': 39, 'Alberto': 34, 'Javier': 33, 'Cristina': 30, 'Elena': 23}
>>> for k, v in d.items():
...     print(k, v)
...
David 39
Alberto 34
Javier 33
Cristina 30
Elena 23
```

Aurretik adierazi dugun bezala, hiztegiak datuak biltegitratzeko datu-egitura idealak dira, eta haietan erlazio argia dago egiturako elementu bakoitzaren erreferentzia edo propietate bakar baten eta dagokion balioaren artean.

#### ⚠ ADI

Python-en multzo bateko elementuak gordetzeko datu-egitura bat aukeratzeko orduan, zalantzan izango dugu zer erabili; alegia, zerrenda bat edo hiztegi bat. Oro har, garrantzitsua da gogoratzea Python-en hiztegia azkarragoa dela zerrenda baino. Gainera, bai elementuen bilaketa, bai haien ibilbidea eraginkorragoa da hiztegi batean. Hala ere, zerrendan elementuek ordena izaten dute, eta hiztegi batean ez da hori gertatzen.

## Python-eko hiztegi motaren erabileraren adibidea

Atal honetan, Python-eko hiztegien erabileraren adibide xume bat ikusiko dugu. Funtsean, hiztegi bat erabiltzen da munduko hiriburuei buruzko informazioa biltegitratzeko eta erabiltzeko.





```
>>> hiriburuak = {'Espainia': 'Madril', 'Portugal': 'Lisboa', 'Frantzia': 'Paris', 'Italia':  
'Erroma'}  
>>> print(len(hiriburuak))  
4  
>>> if 'Espainia' in hiriburuak:  
...     print(hiriburuak ['Espainia'])  
...  
Madril  
>>> print(hiriburuak ['Erresuma Batua'])  
Traceback (most recent call last):  
  File "<stdin>", line 1, in <module>  
KeyError: 'Erresuma Batua'  
>>> 'Grezia' in hiriburuak  
False  
>>> hiriburuak.pop('Italia')  
'Erroma'  
>>> print(hiriburuak)  
{'Espainia': 'Madril', 'Portugal': 'Lisboa', 'Frantzia': 'Paris'}  
>>> hiriburuak ['Portugal'] = 'Oporto'  
>>> print(hiriburuak)  
{'Espainia': 'Madril', 'Portugal': 'Oporto', 'Frantzia': 'Paris'}
```

Hiztegi batera sartu nahi denean bertan ez dagoen gako baten bidez —halaxe gertatzen da aurreko adibidean *hiriburuak Greziaren* bidez indexatzean—, interpretatzaileak salbuespen bat ematen du, eta gako horrekin errore bat dagoela adierazten du. Gako bat hiztegi batean dagoen egiazta daiteke **in** operadorearen bidez.

Bestalde, aurreko adibidean **pop()** eragiketaren erabilera ere erakusten da. Eragiketa hori erabiltzen da gako baten bitartez hiztegiko sarrera bat ezabatzeko.

Hiztegiko sarrera bat ere eguneratu daiteke. Horretarako, aski da lehendik dagoen gakoaren bidez indexatzea, eta balio berria adieraztea. Kasu zehatz hori erakusten da, adibidean, *Portugalgo* hiriburua eguneratuz.

Azkenik, hiztegi batean elementuak ordenatzeko eragiketak egiteko aukera dago. Adibidez, **sorted()** funtzioa erabil daiteke hiztegi bateko gakoak dituen zerrenda bat, goranzko ordenan, lortzeko:

```
>>> sorted(hiriburuak)  
['Espainia', 'Frantzia', 'Portugal']
```



Eduki digitalak  
sortzea

**B2 maila** 3.4 Programazioa

# Python-eko fitxategien oinarrizko erabilera







## Python-eko fitxategien oinarrizko erabilera

Python-ek baditu fitxategiak erabiltzeko zenbait ezaugarri integratu. Funtzio horien bitartez programatzaileek aukera dute testu-fitxategiak modu erraz eta eraginkorrean sortzeko, irakurtzeko, idazteko eta erabiltzeko. Jarraian, Python-en fitxategiak sortzeko eta kudeatzeko prozesua aztertuko dugu, eta oinarrizko eragiketak azpimarratuko ditugu, hala nola fitxategiak irekitzea, irakurtzea, idaztea eta ixtea.

### Testu-fitxategi bat sortzea zenbait eduki-lerroekin

Python-ek `open()` funtzioa erabiltzen du fitxategi bat irekitzeko. Zehaztutako fitxategia ez badago, Python-ek automatikoki sortuko du. Fitxategi batean idazteko, `write()` metodoa erabiltzen da. Adibidez, zenbait lerroko testu-fitxategi bat sortuko litzateke:

```
# Sortu 'adibidea.txt' deitutako fitxategia
fitxategia = open('adibidea.txt', 'w')

# Idatzi zenbait lerro fitxategian
fitxategia.write('Hau da lehen lerroa.\n')
fitxategia.write('Eta hau da bigarren lerroa.\n')
fitxategia.write('Azkenik, hau da hirugarren lerroa.\n')

# Itxi fitxategia
fitxategia.close ()
```

Testu-editore batekin *adibidea.txt* fitxategia irekitzen bada, eduki hau erakutsiko da:

```
Hau da lehen lerroa.
Eta hau da bigarren lerroa.
Azkenik, hau da hirugarren lerroa.
```

`open()` funtzioko `'w'` argumentuak fitxategia idazteko (*write*) irekitzen dela adierazten du. Ikuspegi hori erabilita fitxategi horretan berriro idazten bada, artxiboa eduki berriarekin gainidatziko da, eta lehendik dagoen edukia ezabatuko da. Portaera hori eragozteko, fitxategiaren irekiera konfiguratu daiteke lehendik dagoen fitxategi batera edukia gehitzeko, `'a'` argumentua erabiliz (*append*). Adibidez:



```
# Ireki 'adibidea.txt' deitutako fitxategia
fitxategia = open("adibidea.txt", "a")

# Gehitu bi lerro fitxategiari
fitxategia.write('Hau lerro berri bat da.\n')
fitxategia.write('Eta hau fitxategiari erantsitako beste lerro bat da.\n')

# Itxi fitxategia
fitxategia.close ()
```

Orain, fitxategia testu-editore batekin irekitzean, honako eduki hau agertuko da:

```
Hau da lehen lerroa.
Eta hau da bigarren lerroa.
Azkenik, hau da hirugarren lerroa.
Hori lerro berria da.
Eta hau da fitxategiari erantsitako beste lerro bat.
```

## Oinarrizko eragiketak: Irekitzea, irakurtzea, idaztea eta ixtea

### Irekitzea

Lehen aipatu bezala, `open()` funtzioa erabiltzen da fitxategi bat irekitzeko. Funtzio horrek bi argumentu ditu: fitxategiaren izena eta modua:

```
# Ireki fitxategi bat irakurtzeko
fitxategia = open('adibidea.txt', 'r')
```

Nabarmentzekoa da `open()` funtzioaren bigarren argumentua aukerakoa dela eta, zehazten ez bada, `'r'` balioa hartuko duela lehenetsita.

### Irakurtzea

Fitxategi bat ireki ondoren, zenbait metodo erabil daitezke haren edukia irakurtzeko. `read()` metodoak fitxategiaren eduki guztia irakurtzen du eta adierazten dugun aldagai batean gordetzen du:

```
# Irakurri fitxategiaren eduki guztia
edukia = fitxategia.read()
print(edukia)
```



## Idaztea

Fitxategi batean idazteko, lehendabizi idazteko ('w') edo eransteke ('a') moduan ireki behar da. Ondoren, `write()` metodoa erabiltzen da:

```
# Ireki fitxategi bat idazteko
fitxategia = open('adibidea.txt', 'w')

# Idatzi fitxategian
fitxategia.write('Kaixo, mundua!')
```

## Ixtea

Fitxategi batekin lan egin ondoren, garrantzitsua da hura ixtea, sisteman baliabideak askatzeko eta fitxategia sisteman behar bezala gordetzen dela ziurtatzeko. Hartarako, `close()` metodoa erabiltzen da:

```
# Itxi fitxategia
fitxategia.close ()
```

Lana amaitzen denean fitxategia ixtea ez ahazteko, `with` sententzia erabiltzea gomendatzen da. Sententzia horrek aukera ematen du fitxategi bat irekitzeko, eta, harekin lan egiten amaitu ondoren, behar bezala ixten dela ziurtatzeko, baita prozesuan erroreak gertatzen direnean ere. Adibidez:

```
# Fitxategi bat with erabiliz irekitzeko adibidea
with open("fitxategia.txt", "r") as fitxategia:
    edukia = fitxategia.read()
    print(edukia)
```

`with` sententziak indentatutako blokearen exekuzioa amaitzen denean, fitxategia automatikoki itxiko da, `close()` metodoa deitu beharrik gabe.

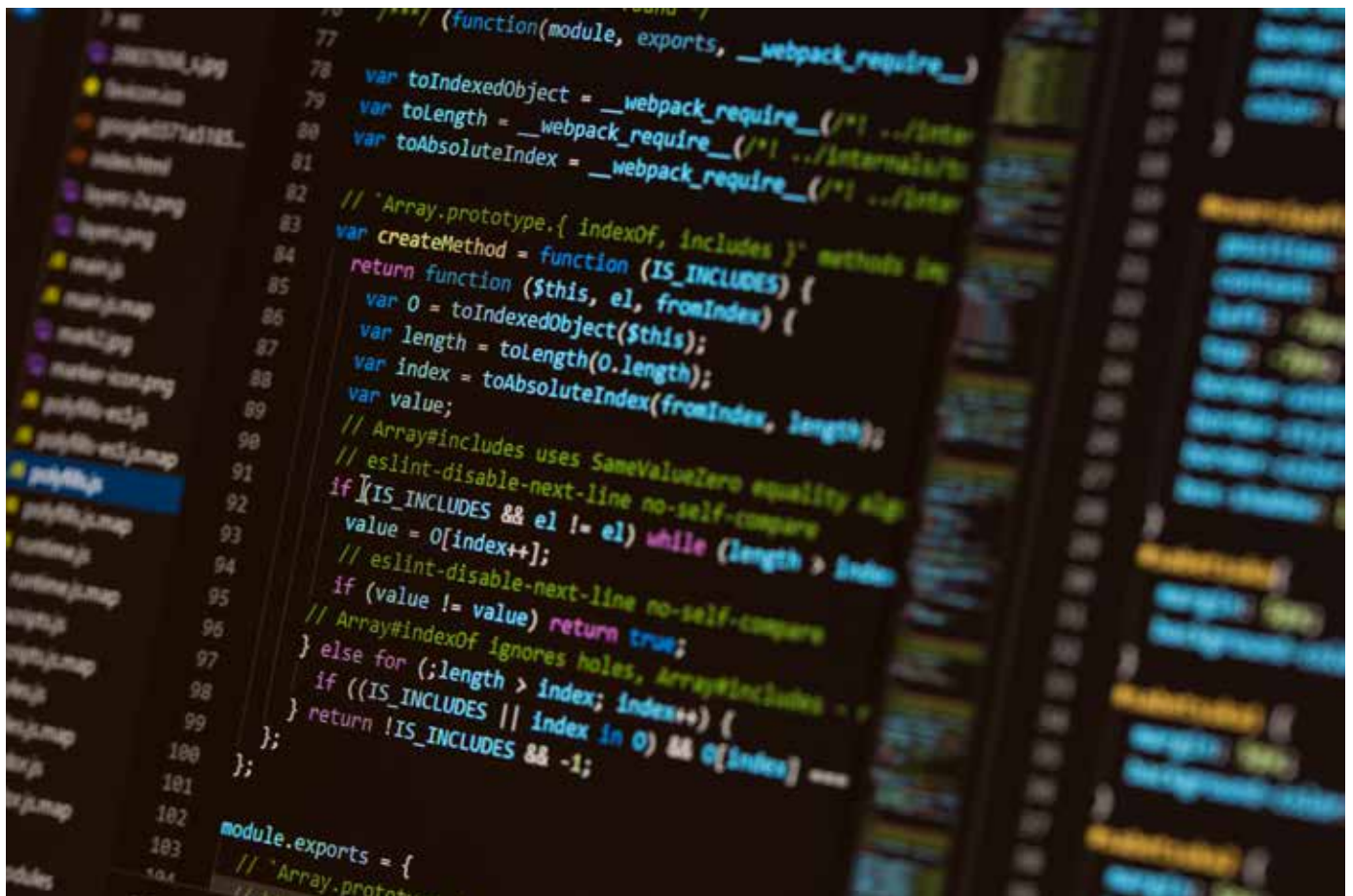


## Ondorioa

Fitxategiak erabiltzea funtsezko eta ohiko zeregina da programazioan. Python-ek funtzio eta metodo integratuak eskaintzen ditu fitxategiak errazago irakurri, idatzi, ireki eta ixteko. Aurretik aztertutako alderdiek oinarritzko eragiketak bakarrik barne hartzen badituzte ere, Python-ek askoz funtzio aurreratu gehiago eskaintzen ditu fitxategiekin lan egiteko. Python-en dokumentazio ofiziala sakonago aztertu beharko duzu haren funtzioak hobeto ulertzeko.

### Informazio gehiago

Python-en fitxategiak erabiltzeari buruz gehiago jakiteko, kontsulta ezazu gaztelaniazko dokumentu ofiziala: [e.digitalall.org.es/archivos-phyton](https://e.digitalall.org.es/archivos-phyton)



```
77  /** (function(module, exports, __webpack_require__)
78  var toIndexedObject = __webpack_require__(/* .../inter
79  var toLength = __webpack_require__(/* .../internals/to
80  var toAbsoluteIndex = __webpack_require__(/* .../inter
81
82  // "Array.prototype.{ indexOf, includes }" methods impl
83  var createMethod = function (IS_INCLUDES) {
84  return function ($this, el, fromIndex) {
85  var O = toIndexedObject($this);
86  var length = toLength(O.length);
87  var index = toAbsoluteIndex(fromIndex, length);
88  var value;
89  // Array#includes uses SameValueZero equality alg
90  // eslint-disable-next-line no-self-compare
91  if (IS_INCLUDES && el !== el) while (length > index)
92  value = O[index++];
93  // eslint-disable-next-line no-self-compare
94  if (value !== value) return true;
95  // Array#indexOf ignores holes, Array#includes - not
96  } else for (;length > index; index++) {
97  if ((IS_INCLUDES || index in O) && O[index] ===
98  ) return !IS_INCLUDES && -1;
99  };
100 };
101
102 module.exports = {
103 // "Array.prototype"
```



Eduki digitalak  
sortzea

**B2 maila 3.4** Programazioa

**Zer da  
software 2.0?**





## Zer da software 2.0?

Software Ingeniaritzaren diziplinak zenbait metodo, teknika eta tresna ditu softwarea garatzea ahalbidetzeko. Orain arte, programatzailea izan da software-garapen horren protagonista bakarra. Garatutako programen kalitatea eta eraginkortasuna beti egon da programatzaileen artearen eta jardun onaren mende. Software-garapenaren ikuskera horrek aditu programatzaileak ditu ardatz, eta 1.0 softwarea esaten zaio. Hala ere, azken urteotan *adimen artifizialaren* diziplinak –bereziki, besteak beste, *Deep Learning* eta *Ikaste Sakona* arlo zehatzetan– izandako iraultzagatik, aldatzen ari da softwarea garatzeko ikuspegi hori, eta bide eman zaie programa informatikoen garapenean lagunduko duten eragile berriei. Eragile horiek makinak dira; gauza dira softwarea modu autonomoan garatzeko, edo, gutxienez, garapen-zeregin batzuetan laguntzeko, eta programatzaile klasikoaren lana errazten dute.

**Software 2.0** terminoa Adrej Karpathyk sortu zuen, hau da, Tesla enpresako Adimen Artifizialaren arloko arduradunak. Softwarea garatzeko era tradizionala motela eta neketsua da, eta horren ondorioz, garatzaileek akats asko egiten dituzte. Alabaina, adimen artifizialaren barruko ikaste automatikoko teknikek garapenaren zati bat automatizatzea ahalbidetzen dute. Hartarako, ordenagailuek adibideen bidez programatzen ikasten dute, hau da, lehendik ezarrita dauden programak oinarri hartuta, haien egitura ulertzeko eta kopiatzeko gai dira. Ikaste horrek programazio-zereginak automatizatzea ahalbidetzen du; garapen-prozesua bizkortzen du, eta egin ohi diren akatsak nabarmen murrizten ditu.

Laburbilduz, 1.0 softwareak oinarri hauek ditu: arazoa aztertzea, beharrezko diren algoritmoak eta osagaiak diseinatzea, eta garatzaileak azken horiek implementatzea eta probatzea. Bestalde, 2.0 softwareak garapen-atazen adibideak biltzen ditu, ordenagailu batek, haietan oinarrituta eta Ikaste Automatikoko tekniken bidez, haiek kopiatzen ikas dezan.



## Adibidea

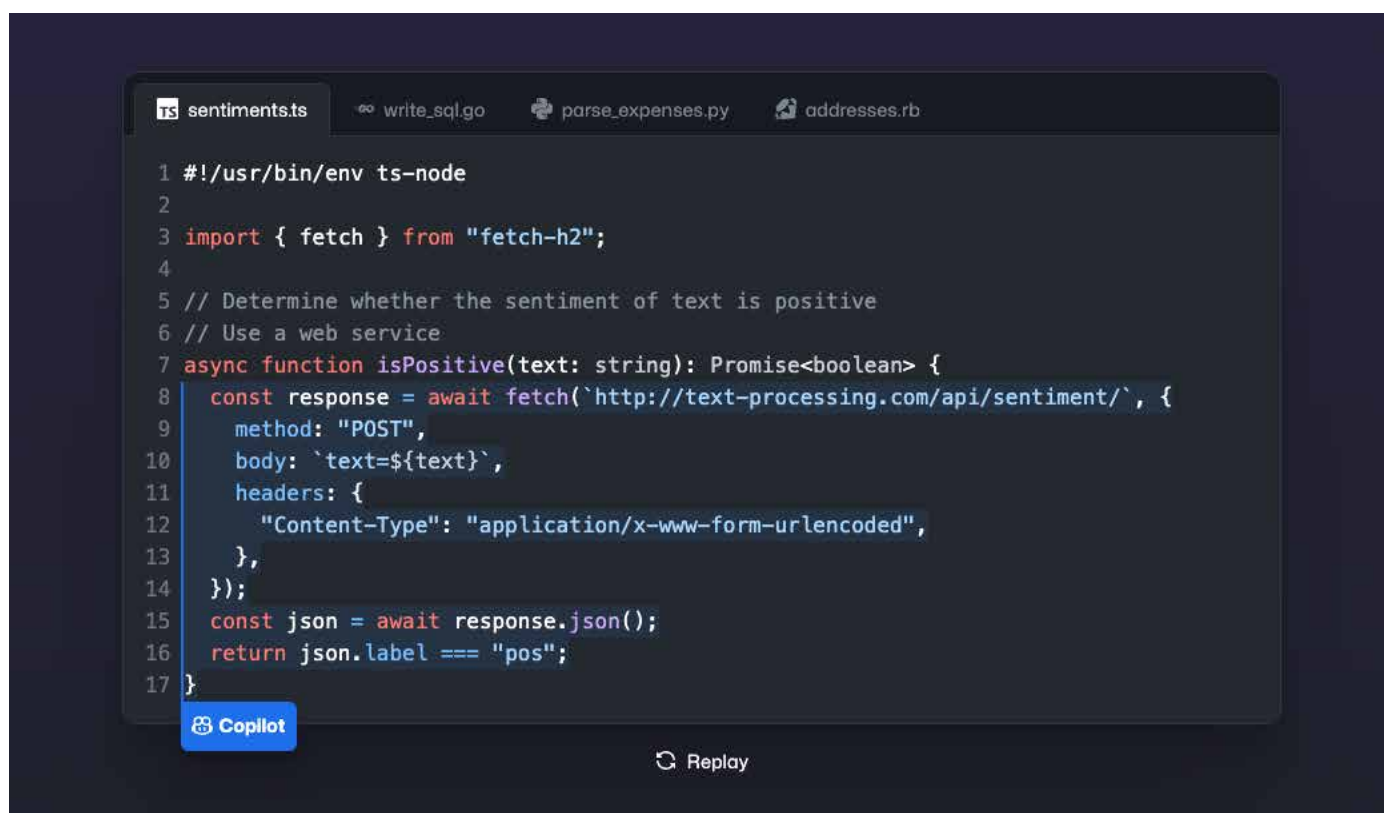
Hurrengo urteetan aplikazio asko aurkituko ditugu 2.0 softwarearekin lotuta, askotariko arloetan, hala nola ikusizko eta ahotsezko ezagutzan, itzulpen automatikoan, jokoetan eta datu-basetan.

Adibide bat **Copilot** da ([github.com/features/copilot](https://github.com/features/copilot)). Tresna honek erabiltzaileari aukera ematen dio ezarri behar duen funtzionalitate bat lengoia naturalean deskribatzeko, eta adierazitako funtzionalitatea iturburu-kodera itzultzen du eskatzen diogun programazio-lengoian.


### Informazio gehiago

**Ikaste sakona.** A. Shrestha eta A. Mahmood: "Review of Deep Learning Algorithms and Architectures" *in* IEEE Access, 7. lib., 53040-53065. or., 2019, doi: 10.1109/ACCESS.2019.2912200.

**Software 2.0aren definizioa:** [e.digitall.org.es/software-2](https://e.digitall.org.es/software-2)



```
1 #!/usr/bin/env ts-node
2
3 import { fetch } from "fetch-h2";
4
5 // Determine whether the sentiment of text is positive
6 // Use a web service
7 async function isPositive(text: string): Promise<boolean> {
8   const response = await fetch(`http://text-processing.com/api/sentiment/`, {
9     method: "POST",
10    body: `text=${text}`,
11    headers: {
12      "Content-Type": "application/x-www-form-urlencoded",
13    },
14  });
15  const json = await response.json();
16  return json.label === "pos";
17 }
```

 Copilot Replay

Copilot-en pantaila-argazkia. Iturria: [github.com/features/copilot](https://github.com/features/copilot).



# DigitAll

Gaitasun  
digitaletan  
prestakuntza





## Coordinación General

**Universidad de Castilla-La Mancha**  
Carlos González Morcillo  
Francisco Parreño Torres

## Coordinadores de área

### Área 1. Búsqueda y gestión de información y datos

**Universidad de Zaragoza**  
Francisco Javier Fabra Caro

### Área 2. Comunicación y colaboración

**Universidad de Sevilla**  
Francisco Javier Fabra Caro  
Francisco de Asís Gómez Rodríguez  
José Mariano González Romano  
Juan Ramón Lacalle Remigio  
Julio Cabero Almenara  
María Ángeles Borrueco Rosa

### Área 3. Creación de contenidos digitales

**Universidad de Castilla-La Mancha**  
David Vallejo Fernández  
Javier Alonso Albusac Jiménez  
José Jesús Castro Sánchez

### Área 4. Seguridad

**Universidade da Coruña**  
Ana M. Peña Cabanas  
José Antonio García Naya  
Manuel García Torre

### Área 5. Resolución de problemas

**UNED**  
Jesús González Boticario

## Coordinadores de nivel

### Nivel A1

**Universidad de Zaragoza**  
Ana Lucía Esteban Sánchez  
Francisco Javier Fabra Caro

### Nivel A2

**Universidad de Córdoba**  
Juan Antonio Romero del Castillo  
Sebastián Rubio García

### Nivel B1

**Universidad de Sevilla**  
Francisco de Asís Gómez Rodríguez  
José Mariano González Romano  
Juan Ramón Lacalle Remigio  
Montserrat Argandoña Bertran

### Nivel B2

**Universidad de Castilla-La Mancha**  
María del Carmen Carrión Espinosa  
Rafael Casado González  
Víctor Manuel Ruiz Penichet

### Nivel C1

**UNED**  
Antonio Galisteo del Valle

### Nivel C2

**UNED**  
Antonio Galisteo del Valle

## Maquetación

**Universidad de Salamanca**  
Fernando De la Prieta Pintado  
Pilar Vega Pérez  
Sara Alejandra Labrador Martín

# Creadores de contenido

## Área 1. Búsqueda y gestión de información y datos

### 1.1 Navegar, buscar y filtrar datos, información y contenidos digitales

#### Universidad de Huelva

Ana Duarte Hueros (coord.)  
Arantxa Vizcaíno Verdú  
Carmen González Castillo  
Dieter R. Fuentes Cancell  
Elisabetta Brandi  
José Antonio Alfonso Sánchez  
José Ignacio Aguaded  
Mónica Bonilla del Río  
Odriel Estrada Molina  
Tomás de J. Mateo Sanguino (coord.)

### 1.2 Evaluar datos, información y contenidos digitales

#### Universidad de Zaragoza

Ana Belén Martínez Martínez  
Ana María López Torres  
Francisco Javier Fabra Caro  
José Antonio Simón Lázaro  
Laura Bordonaba Plou  
María Sol Arqued Ribes  
Raquel Trillo Lado

### 1.3 Gestión de datos, información y contenidos digitales

#### Universidad de Zaragoza

Ana Belén Martínez Martínez  
Francisco Javier Fabra Caro  
Gregorio de Miguel Casado  
Sergio Ilarri Artigas

## Área 2. Comunicación y colaboración

### 2.1 Interactuar a través de tecnología digitales

Iseazy

### 2.2 Compartir a través de tecnologías digitales

#### Universidad de Sevilla

Alién García Hernández  
Daniel Agüera García  
Jonatan Castaño Muñoz  
José Candón Mena  
José Luis Guisado Lizar

### 2.3 Participación ciudadana a través de las tecnologías digitales

#### Universidad de Sevilla

Ana Mancera Rueda  
Félix Biscarri Triviño  
Francisco de Asís Gómez Rodríguez  
Jorge Ruiz Morales  
José Manuel Sánchez García  
Juan Pablo Mora Gutiérrez  
Manuel Ortigueira Sánchez  
Raúl Gómez Bizcocho

### 2.4 Colaboración a través de las tecnologías digitales

#### Universidad de Sevilla

Belén Vega Márquez  
David Vila Viñas  
Francisco de Asís Gómez Rodríguez  
Julio Barroso Osuna  
María Puig Gutiérrez  
Miguel Ángel Olivero González  
Óscar Manuel Gallego Pérez  
Paula Marcelo Martínez

### 2.5 Comportamiento en la red

#### Universidad de Sevilla

Ana Mancera Rueda  
Eva Mateos Núñez  
Juan Pablo Mora Gutiérrez  
Óscar Manuel Gallego Pérez

### 2.6 Gestión de la identidad digital

Iseazy

## Área 3. Creación de contenidos digitales

### 3.1 Desarrollo de contenidos

#### Universidad de Castilla-La Mancha

Carlos Alberto Castillo Sarmiento  
Diego Cordero Contreras  
Inmaculada Ballesteros Yáñez  
José Ramón Rodríguez Rodríguez  
Rubén Grande Muñoz

### 3.2 Integración y reelaboración de contenido digital

#### Universidad de Castilla-La Mancha

José Ángel Martín Baos  
Julio Alberto López Gómez  
Ricardo García Ródenas

### 3.3 Derechos de autor (copyright) y licencias de propiedad intelectual

#### Universidad de Castilla-La Mancha

Gabriela Raquel Gallicchio Platino  
Gerardo Alain Marquet García

### 3.4 Programación

#### Universidad de Castilla-La Mancha

Carmen Lacave Rodero  
David Vallejo Fernández  
Javier Alonso Albusac Jiménez  
Jesús Serrano Guerrero  
Santiago Sánchez Sobrino  
Vanesa Herrera Tirado

## Área 4. Seguridad

### 4.1 Protección de dispositivos

#### Universidade da Coruña

Antonio Daniel López Rivas  
José Manuel Vázquez Naya  
Martíño Rivera Dourado  
Rubén Pérez Jove

### 4.2 Protección de datos personales y privacidad

#### Universidad de Córdoba

Aida Gema de Haro García  
Ezequiel Herruzo Gómez  
Francisco José Madrid Cuevas  
José Manuel Palomares Muñoz  
Juan Antonio Romero del Castillo  
Manuel Izquierdo Carrasco

### 4.3 Protección de la salud y del bienestar

#### Universidade da Coruña

Javier Pereira Loureiro  
Laura Nieto Riveiro  
Laura Rodríguez Gesto  
Manuel Lagos Rodríguez  
María Betania Groba González  
María del Carmen Miranda Duro  
Nereida María Canosa Domínguez  
Patricia Concheiro Moscoso  
Thais Pousada García

### 4.4 Protección medioambiental

#### Universidad de Córdoba

Alberto Membrillo del Pozo  
Alicia Jurado López  
Luis Sánchez Vázquez  
María Victoria Gil Cerezo

## Área 5. Resolución de problemas

### 5.1 Resolución de problemas técnicos

Iseazy

### 5.2 Identificación de necesidades y respuestas tecnológicas

Iseazy

### 5.3 Uso creativo de la tecnología digital

Iseazy

### 5.4 Identificar lagunas en las competencias digitales

Iseazy



El material del proyecto DigitAll se distribuye bajo licencia CC BY-NC-SA 4.0. Puede obtener los detalles de la licencia completa en: <https://creativecommons.org/licenses/by-nc-sa/4.0/deed.es>