Tarea 2: Web scraping

En esta tarea obtendremos los datos para rellenar una BD en la tarea siguente. Haremos scraping del sitio web del Museo Arqueológico para recoger la información de las 'obras singulares': título, imágen, descripción, procedencia, y comentario de cada una de ellas.

Scrapper

Para el scrapper usaremos la librería Playwright. Esta librería es mas completa que simplemente cargar el html: instancia un navegador y funciona tal cual este, esperarando asíncronamente a que se vayan cargando todos los elementos de la página. Además la vamos a utilizar también para testing mas adelante.

Instalamos e inicializamos `Playwright` :

```
> pnpm i -D playwright
> pnpm exec playwright install
```

'-D' es para que se instale como dependencia de desarrollo.

Siguiendo A Guide to Playwright Web Scraping in 2025, hacemos un script **scrap.mjs**:

```
import {chromium} from "playwright"  // o el que sea

const browser = await chromium.launch()
const page    = await browser.newPage()

// lista de páginas con enlaces a 'obras-singulares'
const obras_singulares = [
  "https://www.museosdeandalucia.es/web/museoarqueologicodegranada/obr
  ...
]

// Los métodos dePlaywright son asíncronos, devuelven promesas
(async () => {

  const enlaces_de_obras_singulares = []
  const lista_info_para_BD          = []

  for (const pag of obras_singulares) {
    const urls = await Recupera_urls_de(pag)
    enlaces_de_obras_singulares.push(...urls)  // ... operador spread
  }
  console.log("🚀 Hay ", enlaces_de_obras_singulares.length, ' página

  for (const url of enlaces_de_obras_singulares) {
    const info_obra = await Recupera_info_de(url)
    lista_info_para_BD.push(info_obra)
  }

  Guarda_en_disco('info_obras.json', lista_info_para_BD)
```

```
  await browser.close();
})();
```

Para recuperar la información que nos interese, están los locators de playwright. Para acceder a el/los elemento(s) que nos interesa(n), se usa la misma sintáxis de los selectores de css, p.e. para coger los enlaces de las páginas de 'obras-singulares':

```
const locators = page.locator('.descripcion > a')
```

ya que queremos enlaces debajo un elemento con clase 'descripcion' (ver código fuente de la página). La función queda:

```
async function Recupera_urls_de(pag) {
  const pags = []
  await page.goto(pag);
  const locators = page.locator('.descripcion > a')
  for (const locator of await locators.all()) {
    pags.push(await locator.getAttribute('href'))
  }
  return pags
}
```

Referencias:

- Playwright Getting Started
- Locator API

- CSS Selectors: A Visual Guide