🧠 Exercise: Article Pagination in PHP

🎯 Learning Objective

Learn how to paginate results from a database using PHP:

- Part 1: Create an HTML page displaying paginated articles.
- Part 2: Create a REST API returning paginated articles in JSON format.

📦 Database Used

We will work on the `article` table from the beer database, already used in previous SQL exercises.

🧩 Part 1: HTML Page with Pagination

🔧 Instructions for the Page

1. Write a PHP script `articles.php` that:

   - Connects to the database.
   - Retrieves a defined number of articles per page (e.g., 10).
   - Displays the articles in an HTML table.
   - Displays pagination links (previous page, next page, page numbers...).

2. Add the following features:

   - Indicate the current page.
   - Handle edge cases (negative page numbers, page too high, etc.).
   - Protect against SQL injections (prepared statements required).

💡 Tips

- Use `LIMIT` and `OFFSET` in SQL.
- Offset calculation: `(page - 1) * articles_per_page`.
- Example link: `articles.php?page=2`.

🌐 Part 2: Paginated API (JSON format)

🔧 API Instructions

Create a file `api/articles.php` that:

- Returns paginated articles as JSON.
- Accepts `page` and `limit` parameters via `GET`.
- Also returns metadata in the JSON (e.g., `total`, `current_page`, `last_page`).

🗒️ Example Output

```
{
  "current_page": 2,
  "per_page": 10,
  "total": 58,
  "last_page": 6,
  "data": [
    {
      "ID_ARTICLE": 11,
      "NOM_ARTICLE": "Vin rouge doux",
      "PRIX_ACHAT": 12.5
    }
  ]
}
```

✅ Bonus (optional)

- Allow filtering articles by `ID_TYPE` or `ID_MARQUE` (via additional GET parameters).
- Return HTTP 400 status code for invalid parameters (e.g., negative page).

✳️ Tips

- First make sure SQL pagination works before integrating it into PHP.
- Test your URLs manually in the browser (`?page=1&limit=20`).
- Use `json_encode()` with `JSON_PRETTY_PRINT` to test.

🧪 Testing Your API

Use Postman or your browser to test the endpoints:

```
GET http://localhost/api/articles.php?page=1&limit=10
```

📂 Suggested Structure

```
/project-root
├── articles.php
└── api/
    └── articles.php
```