

# GET /api/resources/ — list resources

## Purpose

Return a paginated list of curated external resources (articles, videos, PDFs). Public endpoint — no auth required.

**URL :** GET http://localhost:8000/api/resources/

(Use your production base URL in production.)

## Permissions

- `AllowAny` — public.

## Query parameters

- `q` — optional. Free-text search applied to `title` OR `description` (case-insensitive, partial match).
  - Example: `?q=stress`
- `type` — optional. Filter by resource type. Allowed values (case-insensitive): `article`, `video`, `pdf`.
  - Example: `?type=video`
  - Any other value is ignored (no filter applied).
- Pagination:
  - `page` — optional page number (1-indexed).
  - `page_size` — optional items per page.
  - The default and maximum `page_size` are defined by your `ResourcePagination` class (commonly `page_size=10`, `max_page_size=50`). If not customized, follow your project's pagination defaults.

## Sorting

- Results are ordered by `-created_at` (newest first) as implemented in `get_queryset()`.

## Response format (200 OK)

Paginated response. Example shape (fields returned by `ResourceSerializer`):

```
{  
    "count": 6,  
    "page": 1,  
    "page_size": 10,  
    "next": "http://localhost:8000/api/resources/?page=2",  
    "previous": null,  
    "results": [  
        {  
            "id": 12,  
            "title": "Mindfulness for Beginners",  
            "resource_type": "video",  
            "thumbnail_url": "https://img.youtube.com/vi/abcd1234/0.jpg",  
            "url": "https://www.youtube.com/watch?v=abcd1234",  
            "description": "A 12-minute introduction to mindfulness.",  
            "tags": ["mindfulness", "intro"],  
            "source": "YouTube",  
            "published_at": "2021-06-01",  
            "created_at": "2025-10-01T10:00:00Z"  
        },  
        {  
            "id": 13,  
            "title": "CBT Workbook (PDF)",  
            "resource_type": "pdf",  
            "thumbnail_url": "https://example.com/thumbs/cbt.png",  
            "url": "https://example.com/resources/cbt-guide.pdf",  
            "description": "Downloadable CBT workbook.",  
            "tags": ["CBT", "workbook"],  
            "source": "ExampleOrg",  
            "published_at": "2019-02-15",  
            "created_at": "2025-09-30T11:00:00Z"  
        }  
        // ...  
    ]  
}
```

## Example requests

- List all (first page):

GET /api/resources/

- Search for "stress":

GET /api/resources/?q=stress

- Videos only:

GET /api/resources/?type=video

- Search + type:

GET /api/resources/?q=stress&type=article

- Pagination:

GET /api/resources/?page=2&page\_size=5

## Frontend (browser console) examples

- Basic:

```
fetch("http://localhost:8000/api/resources/")
  .then(r => r.json())
  .then(console.log);
```

- Search:

```
fetch("http://localhost:8000/api/resources/?q=stress")
  .then(r => r.json()).then(console.log);
```

- Videos only:

```
fetch("http://localhost:8000/api/resources/?type=video")
  .then(r => r.json()).then(console.log);
```

- Combined + pagination:

```
fetch("http://localhost:8000/api/resources/?q=cbt&type=pdf&page=1&page_size=5")
  .then(r => r.json()).then(console.log);
```

## Errors / status codes

- **200 OK** — request successful (may be empty `results` if no matches).
- **400 Bad Request** — invalid query parameters (for example if your pagination class rejects a non-integer `page` or an out-of-range `page_size`). Response will contain a JSON error message.
- **500 Internal Server Error** — server-side error (rare).

## Notes & frontend tips

- Because the `type` filter is case-insensitive and only accepts `article|video|pdf`, send `type` in lowercase for clarity.
- If a `special` parameter or additional filters are later added to the view, the frontend should gracefully ignore unknown query params.
- Use the `next/previous` links from the response to implement “load more” or page navigation (they are full URLs).
- Always handle the case `results: []` and show a friendly “No resources found” UI.
- `url` points outwards — open in a new tab `target="_blank"` with `rel="noopener"`.