*// requires conf.js loaded before this script*

if (typeof SUPABASE\_URL === 'undefined' || typeof SUPABASE\_ANON\_KEY === 'undefined') {

console.error('Please set SUPABASE\_URL and SUPABASE\_ANON\_KEY in conf.js');

}

async function api(path, opts = {}) {

const url = `${SUPABASE\_URL}/rest/v1/${path}`;

const headers = Object.assign({'apikey': SUPABASE\_ANON\_KEY, 'Authorization': 'Bearer ' + SUPABASE\_ANON\_KEY, 'Content-Type': 'application/json'}, opts.headers || {});

const res = await fetch(url, Object.assign({headers}, opts));

return res;

}

async function loadTOC() {

try {

*// fetch metadata table*

const res = await api('files\_meta?select=\*&order=uploaded\_at.desc');

if (!res.ok) throw new Error('Failed to load files');

const files = await res.json();

const toc = document.getElementById('toc');

toc.innerHTML = '';

files.forEach((f, idx) => {

const li = document.createElement('li');

const link = document.createElement('a');

const gated = f.gated === true || idx >= 20;

if (!gated) {

link.href = `${SUPABASE\_URL}/storage/v1/object/public/${STORAGE\_BUCKET}/${encodeURIComponent(f.storage\_name)}`;

link.target = '\_blank';

} else {

link.href = `subscribe.html?file=${encodeURIComponent(f.id)}`;

}

link.textContent = f.title || f.storage\_name;

li.appendChild(link);

if (gated) {

const lock = document.createElement('span'); lock.textContent = ' 🔒'; lock.className = 'lock';

li.appendChild(lock);

}

toc.appendChild(li);

});

} catch (err) {

console.error(err);

document.getElementById('toc').innerHTML = '<li>Unable to load content</li>';

}

}

window.addEventListener('load', loadTOC);