

GetCourse — кнопка копирования абсолютной ссылки (userscript)

Что делает

Добавляет в админке GetCourse рядом со ссылками в таблицах маленькую иконку «копировать». Клик по ней копирует **абсолютный URL** страницы (с доменом) в буфер обмена. После копирования иконка на ~1 секунду превращается в зелёную галочку.

ID	Название	Адрес	Теги	Og Image	Ин
4547481	decembrie-keto-1200-IPSBZf	decembrie-keto-1200-IPSBZf	нет тегов	-	
4547480	decembrie-keto-1300-5pDIHu	decembrie-keto-1300-5pDIHu	нет тегов	-	
4547479	decembrie-keto-1400-mw2zyg	decembrie-keto-1400-mw2zyg	нет тегов	-	
4547478	decembrie-keto-1500-jgIPcV	decembrie-keto-1500-jgIPcV	нет тегов	-	
4547477	decembrie-keto-1600-iK5z21	decembrie-keto-1600-iK5z21	нет тегов	-	
4547476	decembrie-keto-1700-4Wq9dk	decembrie-keto-1700-4Wq9dk	нет тегов	-	
4547475	decembrie-keto-1800-MYAfpo	decembrie-keto-1800-MYAfpo	нет тегов	-	
4547474	decembrie-keto-1900-TEDIHv	decembrie-keto-1900-TEDIHv	нет тегов	-	
4547473	decembrie-keto-2000-C6zg9l	decembrie-keto-2000-C6zg9l	нет тегов	-	
4547472	decembrie-keto-2100-8zNDIR	decembrie-keto-2100-8zNDIR	нет тегов	-	
4547471	decembrie-keto-2200-nt4Wx9	decembrie-keto-2200-nt4Wx9	нет тегов	-	

Код

```
// ==UserScript==
// @name      GetCourse — копировать абсолютную ссылку
// @namespace  fitnessmama.school
// @version    1.3
// @description Кнопка копирования абсолютного URL рядом со ссылками в админке GetCourse
// @match     https://fitnessmama.school/pl/*
// @match     https://*.getcourse.ru/pl/*
// @run-at    document-idle
// @grant     none
// ==/UserScript==
```

```
(function () {
  'use strict';

  const BTN_CLASS = 'gc-copy-abs-link-btn';
  const PROCESSED_ATTR = 'data-gc-copy-added';

  const style = document.createElement('style');
  style.textContent = `
    td.w5 > a.not-pjax-link[${PROCESSED_ATTR}] {
      margin-right: 2px;
    }
    .${BTN_CLASS} {
      display: inline-flex;
      align-items: center;
      justify-content: center;
      width: 18px;
      height: 18px;
      padding: 0;
      margin: 0 0 0 2px;
      border: none;
      border-radius: 4px;
      background: transparent;
      color: #8a96a8;
      cursor: pointer;
      vertical-align: -4px;
      opacity: 0.7;
      transition: background .15s ease, color .15s ease;
    }
    .${BTN_CLASS}:hover { opacity: 1; background: #e8f0fe; color: #1a73e8; }
  `;
  document.head.appendChild(style);
})
```

```

.${BTN_CLASS}:active { background: #d2e3fc; }
.${BTN_CLASS}.copied { opacity: 1; background: #e6f4ea; color: #188038; }
.${BTN_CLASS} svg { width: 13px; height: 13px; display: block; }
`;
document.head.appendChild(style);

const SVG_NS = 'http://www.w3.org/2000/svg';

function svg(paths) {
  const s = document.createElementNS(SVG_NS, 'svg');
  s.setAttribute('viewBox', '0 0 24 24');
  s.setAttribute('fill', 'none');
  s.setAttribute('stroke', 'currentColor');
  s.setAttribute('stroke-width', '2');
  s.setAttribute('stroke-linecap', 'round');
  s.setAttribute('stroke-linejoin', 'round');
  paths.forEach(([tag, attrs]) => {
    const el = document.createElementNS(SVG_NS, tag);
    for (const k in attrs) el.setAttribute(k, attrs[k]);
    s.appendChild(el);
  });
  return s;
}

const iconCopy = () => svg([
  ['rect', { x: 9, y: 9, width: 13, height: 13, rx: 2 }],
  ['path', { d: 'M5 15H4a2 2 0 0 1-2-2V4a2 2 0 0 1 2-2h9a2 2 0 0 1 2 2v1' }],
]);

const iconOk = () => {
  const s = svg(['polyline', { points: '20 6 9 17 4 12' }]);
  s.setAttribute('stroke-width', '3');
  return s;
};

function makeButton(absUrl) {
  const btn = document.createElement('button');
  btn.type = 'button';
  btn.className = BTN_CLASS;
  btn.title = 'Скопировать ссылку';
  btn.setAttribute('aria-label', 'Скопировать ' + absUrl);
}

```

```

btn.appendChild(iconCopy());
btn.addEventListener('click', async (e) => {
  e.preventDefault();
  e.stopPropagation();
  try {
    await navigator.clipboard.writeText(absUrl);
  } catch {
    const ta = document.createElement('textarea');
    ta.value = absUrl;
    ta.style.position = 'fixed';
    ta.style.opacity = '0';
    document.body.appendChild(ta);
    ta.select();
    document.execCommand('copy');
    ta.remove();
  }
  btn.classList.add('copied');
  btn.replaceChildren(iconOk());
  setTimeout(() => {
    btn.classList.remove('copied');
    btn.replaceChildren(iconCopy());
  }, 1100);
});
return btn;
}

function process(root) {
  if (!root || root.nodeType !== 1) root = document.body;
  const links = root.querySelectorAll('td.w5 > a.not-pjax-link[href]');
  links.forEach((a) => {
    if (a.hasAttribute(PROCESSED_ATTR)) return;
    const href = a.getAttribute('href');
    if (!href || href.startsWith('javascript:') || href.startsWith('#')) return;
    let absUrl;
    try { absUrl = new URL(href, location.origin).href; } catch { return; }
    a.setAttribute(PROCESSED_ATTR, '1');
    const btn = makeButton(absUrl);
    if (a.nextSibling) a.parentNode.insertBefore(btn, a.nextSibling);
    else a.parentNode.appendChild(btn);
  });
}

```

```

function cleanupOld() {
  document.querySelectorAll('td.w5 > a.not-pjax-link').forEach((a) => {
    let n = a.nextSibling;
    while (n) {
      const next = n.nextSibling;
      if (n.nodeType === 3 && /^\\s*\\s*\\s*$/i.test(n.textContent)) {
        n.remove();
      } else if (n.nodeType === 1 && n.tagName === 'BUTTON' && (n.textContent.includes(' ') ||
n.textContent.includes('&#128203;'))) {
        n.remove();
      }
      n = next;
    }
  });
}

cleanupOld();
process(document.body);

const observer = new MutationObserver((mutations) => {
  for (const m of mutations) {
    m.addedNodes.forEach((node) => {
      if (node.nodeType === 1) process(node);
    });
  }
});
observer.observe(document.body, { childList: true, subtree: true });
})();

```

Revision #6

Created 28 April 2026 16:57:36 by David

Updated 28 April 2026 21:59:49 by David