

# Lister les nouveautés de GitLab-CE

Entrer les lignes suivantes dans la console Javascript du navigateur

```
{
  const version = document.querySelector("h1").textContent;
  const wishesSections = [
    "top-feature",
    "primary-features",
    "secondary-features",
  ];
  const availableData =
    '[data-original-title="Available in GitLab self-managed Free"]';
  const notAvailable =
    '[data-original-title="Not available in GitLab self-managed Free"]';
  const [topFeatures, primaryFeatures, secondaryFeatures] = Array.from(
    document.querySelectorAll(".content > section")
  ).filter((el) => wishesSections.includes(el.id));

  const nestedElement = (el, overload, linkEl = "previousElementSibling") => {
    const paths = overload.split(".");
    let result = el.parentNode.parentNode.parentNode;
    const doc =
      result.parentNode.lastElementChild.firstElementChild.firstElementChild
        .href;
    const docLink = doc ? `[_(doc)](${doc})` : "";
    for (path of paths) {
      result = result[path];
    }
    const link = result[linkEl];
```

```

    return [\` - [`${result.textContent}`](`${link.href}`) `${docLink}`\`];
  };

  const hideElements = (el, overload, linkEl = false) => {
    const paths = overload.split(".");
    let result = el.parentNode.parentNode.parentNode;
    for (path of paths) {
      result = result[path];
    }
    linkEl && result.previousElementSibling.style.display = "none";
    result.style.display = "none";
  };

  Array.from(topFeatures.querySelectorAll(notAvailable)).forEach((el) =>
    hideElements(el, "parentNode")
  );

  Array.from(primaryFeatures.querySelectorAll(notAvailable)).forEach((el) =>
    hideElements(el, "parentNode.parentNode", true)
  );

  Array.from(secondaryFeatures.querySelectorAll(notAvailable)).forEach((el) =>
    hideElements(el, "parentNode")
  );

  const primaryAvailable = Array.from(
    primaryFeatures.querySelectorAll(availableData)
  ).map((el) =>
    nestedElement(
      el,
      "parentNode.parentNode.previousElementSibling.firstElementChild.lastElementChild"
    )
  );

  const secondaryAvailable = Array.from(
    secondaryFeatures.querySelectorAll(availableData)
  ).map((el) => nestedElement(el, "previousElementSibling.lastElementChild"));

  const finalResult = []

```

```

    [].concat(version, primaryAvailable, secondaryAvailable)
    [].join("\n");

    console.log(finalResult);

    const copyButton = document.createElement("button");
    const defaultOpacity = "0.4";
    copyButton.style.position = "fixed";
    copyButton.style.margin = "20px";
    copyButton.style.bottom = "0";
    copyButton.style.zIndex = "10";
    copyButton.style.backgroundColor = "teal";
    copyButton.style.color = "wheat";
    copyButton.style.fontSize = "1.5em";
    copyButton.style.fontWeight = "bold";
    copyButton.style.borderRadius = "10px";
    copyButton.style.transition = "all 0.4s 0.1s ease-in";
    copyButton.style.opacity = defaultOpacity;
    copyButton.onmouseover = function () {
        this.style.opacity = "1";
        this.style.boxShadow = "black -3px 2px 6px";
        this.style.transform = "translate(3px, -2px)";
    };
    copyButton.onmouseleave = function () {
        this.style.opacity = defaultOpacity;
        this.style.boxShadow = "unset";
        this.style.transform = "translate(-3px, 2px)";
    };
    copyButton.innerHTML = "Copier la liste<br><em>Community Edition</em>";
    document.body.insertBefore(copyButton, document.body.firstChild);
    const copyFunction = () => {
        navigator.clipboard.writeText(finalResult).then(
            () => {
                console.log("Données copiées dans le presse-papier.");
            },
            (e) => {
                console.error("Les données ne sont pas copiées...", e);
            }
        );
    };

```

```
document.execCommand("copy");
};
copyButton.addEventListener("click", copyFunction);
}
```

# Version Bookmarklet

Pour une utilisation régulière, il faut ajouter un marque-page, puis remplacer l'URL par le code suivant :

```
javascript: const version = document.querySelector("h1").textContent; const wishesSections = [
"top-feature", "primary-features", "secondary-features", ]; const availableData = '[data-
original-title="Available in GitLab self-managed Free"]'; const notAvailable = '[data-
original-title="Not available in GitLab self-managed Free"]'; const [topFeatures,
primaryFeatures, secondaryFeatures] = Array.from( document.querySelectorAll(".content >
section") ).filter((el) => wishesSections.includes(el.id)); const nestedElement = (el,
overload, linkEl = "previousElementSibling") => { const paths = overload.split("."); let
result = el.parentNode.parentNode.parentNode; const doc =
result.parentNode.lastElementChild.firstElementChild.firstElementChild.href; const docLink =
doc ? `[_(doc)](${doc})` : ""; for (path of paths) { result = result[path]; } const link =
result[linkEl]; return [`${result.textContent}(${link.href}) ${docLink}`]; }; const
hideElements = (el, overload, linkEl = false) => { const paths = overload.split("."); let
result = el.parentNode.parentNode.parentNode; for (path of paths) { result = result[path]; }
if (linkEl) {result.previousElementSibling.style.display = "none"; result.style.display =
"none"; }; Array.from(topFeatures.querySelectorAll(notAvailable)).forEach((el) =>
hideElements(el, "parentNode" ));
Array.from(primaryFeatures.querySelectorAll(notAvailable)).forEach((el) => hideElements(el,
"parentNode.parentNode", true) );
Array.from(secondaryFeatures.querySelectorAll(notAvailable)).forEach((el) => hideElements(el,
"parentNode" )); const primaryAvailable = Array.from(
primaryFeatures.querySelectorAll(availableData) ).map((el) => nestedElement( el,
"parentNode.parentNode.previousElementSibling.firstElementChild.lastElementChild" )); const
secondaryAvailable = Array.from( secondaryFeatures.querySelectorAll(availableData) ).map((el)
=> nestedElement(el, "previousElementSibling.lastElementChild")); const finalResult = []
.concat(version, primaryAvailable, secondaryAvailable) .join("\n"); console.log(finalResult);
const copyButton = document.createElement("button"); const defaultOpacity = "0.4";
copyButton.style.position = "fixed"; copyButton.style.margin = "20px"; copyButton.style.bottom
= "0"; copyButton.style.zIndex = "10"; copyButton.style.backgroundColor = "teal";
copyButton.style.color = "wheat"; copyButton.style.fontSize = "1.5em";
```

```
copyButton.style.fontWeight = "bold"; copyButton.style.borderRadius = "10px";
copyButton.style.transition = "all 0.4s 0.1s ease-in"; copyButton.style.opacity =
defaultOpacity; copyButton.onmouseover = function () { this.style.opacity = "1";
this.style.boxShadow = "black -3px 2px 6px"; this.style.transform = "translate(3px, -2px)"; };
copyButton.onmouseleave = function () { this.style.opacity = defaultOpacity;
this.style.boxShadow = "unset"; this.style.transform = "translate(-3px, 2px)"; };
copyButton.innerHTML = "Copier la liste<br><em>Community Edition</em>";
document.body.insertBefore(copyButton, document.body.firstChild); const copyFunction = () => {
navigator.clipboard.writeText(finalResult).then( () => { console.log("Donn%C3%A9es
copi%C3%A9es dans le presse-papier."); }, (e) => { console.error("Les donn%C3%A9es ne sont pas
copi%C3%A9es%E2%80%A6", e); } ); document.execCommand("copy"); };
copyButton.addEventListener("click", copyFunction);
```

Ensuite, le fait de cliquer sur le marque-page activera le *script*.

En ajoutant comme seul mot-clé : `bm_gitlab`, cela me permet d'entrer ce mot-clé lorsque je suis sur la page de *release* de *GitLab* pour lancer directement le *script*.

## Résultat

Exemple avec la page [GitLab 16.0 released](#).

Les éléments ne correspondant pas à `Self-Managed` `Free` sont masqués sur la page.

Un bouton apparaît permettant de copier la liste suivante dans le presse-papier :

### GitLab 16.0 Release

- [Comment templates \(doc\)](#)
- [Update your fork from the GitLab UI \(doc\)](#)
- [New Web IDE experience now generally available \(doc\)](#)
- [Token rotation API \(doc\)](#)
- [Authenticate with Jira personal access tokens \(doc\)](#)
- [Import collaborators as an additional item to import \(doc\)](#)
- [Opt in to a new navigation experience \(doc\)](#)
- [Add or resolve to-do items on tasks, objectives, and key results \(doc\)](#)
- [GitLab Pages unique subdomains \(doc\)](#)
- [Pick custom colors for labels \(doc\)](#)

- [Real-time merge request updates \(doc\)](#)
- [Configurable maximum number of included CI/CD configuration files \(doc\)](#)
- [Create an instance runner in the Admin Area as a user \(doc\)](#)
- [GitLab Runner 16.0 \(doc\)](#)
- [REST API endpoint to create a runner \(doc\)](#)
- [Trigger job mirror status of downstream pipeline when cancelled \(doc\)](#)
- [Import Maven/Gradle packages by using CI/CD pipelines \(doc\)](#)
- [Placeholder for issue description in Service Desk automated replies \(doc\)](#)
- [Faster, easier Scala scanning in SAST \(doc\)](#)
- [SAST analyzer updates \(doc\)](#)
- [Option to disable followers \(doc\)](#)
- [Self-managed GitLab uses two database connections \(doc\)](#)
- [Filter GitHub repositories to import \(doc\)](#)
- [Limit session length for users \(doc\)](#)
- [Add emoji reactions on tasks, objectives and key results \(doc\)](#)
- [Change work item type from quick action \(doc\)](#)
- [Reorder child records for tasks, objectives and key results \(doc\)](#)
- [CI/CD components \(doc\)](#)
- [Create a group runner as a user \(doc\)](#)
- [Create project runners as a user \(doc\)](#)
- [Per-cache fallback cache keys in CI/CD pipelines \(doc\)](#)
- [Rate Limit for the projects/:id/jobs API endpoint reduced \(doc\)](#)
- [Download packages from the Maven Registry with Scala \(doc\)](#)
- [Display message when deploy freeze is active \(doc\)](#)
- [Secret Detection updates \(doc\)](#)
- [GitLab chart improvements \(doc\)](#)
- [Omnibus improvements \(doc\)](#)
- [Mark to-do items completed by other group or project owners Done \(doc\)](#)
- [Rate limit for unauthenticated users of the Projects List API \(doc\)](#)
- [Additional Registration Features available to Free users \(doc\)](#)

Révision #12

Créé 24 mai 2023 14:48:10 par Mickaël G.

Mis à jour 5 août 2023 21:22:27 par Mickaël G.