



GitHub入門：みんなで「すごいお絵かき」を完成させよう！

みんなの力を合わせて、大きな作品を作り上げよう



ブランチ



プッシュ



プル



マージ

1. GitHub（ギットハブ）ってなに？

GitHubは、

「みんなで1つの大きな作品（お絵かき）を完成させる」ための、

「魔法の本棚」

みんなが作った「お絵かき」のデータ（プログラムのこと）を、インターネット上に置いておく場所だよ。



セーブできる

「ここまでできた！」という場所を何回でも保存できるよ！



元に戻せる

「あ、失敗した！」と思ったら、前の保存場所に戻れるよ！



みんなで見れる

チームのみんなが、最新の「お絵かき」を見たり、合体させたりできるよ！



誰がやったかわかる

「いつ」「だれが」「どこを」変えたのか、全部記録してくれるよ！

なにがうれしいの？

チーム開発を円滑にするGitHubの**4つの主要な利点**だよ！

1



セーブできる

「ここまでできた！」という場所を何回でも**保存**できるよ！
お絵かきを途中まで描いても、後で続けられるようになるんだ！

2



元に戻せる

「あ、失敗した！」と思ったら、**前のバージョン**に戻れるよ！
ミスをしても心配しないで大丈夫！

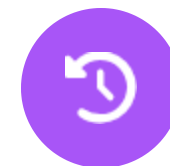
3



みんなで見れる

チームのみんなが、いつでも**最新の絵**を見たり、
自分の作業を合体させたりできるよ！

4



誰がやったかわかる

「いつ」「だれが」「どこを」変えたのか、**全部記録**してくれるよ！
誰かがどんな仕事をしたのか、きれいにわかるんだ！

2. チーム開発の「3つの合言葉」

GitHubを使ってチームでお絵かき（開発）をするとき、**覚えておくべき3つの重要な命令**があります。これを「合言葉」と呼んで、チームで作業するときに使います。

①



コミット (Commit)

「セーブする！」

自分の作業を「保存」するときに使います

お絵かきを「保存する」動作に相当します

②



プッシュ (Push)

「みんなに見せる！（送る）」

自分の「保存」した変更を「魔法の本棚」に送ります

作った絵を「魔法の本棚」に「送る」動作に相当します

③



プル (Pull)

「みんなのをもらう！（引っぱる）」

「魔法の本棚」にある他の人の変更を自分のパソコンにもらいます

他の人が作った絵を「もらってくる」動作に相当します

① コミット (Commit) : 「セーブする！」



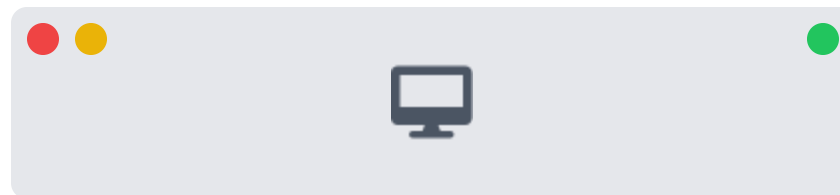
コミットってなに？

自分の「お絵かき」（作業したファイル）を、

「こんな変更をしたよ！」というメモをつけて

保存することだよ！

💡 ポイント：まずは自分のパソコンの中だけに保存されるよ。チームのみんなにはまだ見えないんだ！



お絵かきを
作業する



メモをつける



コミットする

“ 「太陽を描いた」 ”



まだチームのみんなには見えない
「自分のパソコンだけにある保存」だよ

② プッシュ (Push) : 「みんなに見せる！ (送る)」

プッシュとは、

「自分のパソコン」の変更を、
「魔法の本棚 (GitHub)」に送ることだよ！

💡 これで、チームのみんなが「お、太陽が追加されたな」と見られるようになった！

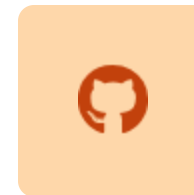


自分のパソコン

📄 太陽を描いた



📶 プッシュ！



魔法の本棚
(GitHub)

📖 太陽が追加された



インターネットに送る

自分のパソコンの変更をインターネット上のGitHubに送るよ



チームと共有

チームのみんなが見られるように共有するよ

③ プル (Pull) : 「みんなのをもらう ! (引っぱる) 」

なにをすること ?

「魔法の本棚 (GitHub) 」にある、**自分"以外"**の人がプッシュした「新しい変更」を、自分のパソコンに持ってくること。

最新の状態を自分のパソコンに反映させるんだよ !

💡 例え :

友だちが「お家」の絵をGitHubにプッシュして、本棚のデータが新しくなった !

「プル ! 」と唱えて、そのデータを自分のパソコンに引っ張ってくる。

これで、自分の絵に「お家」が追加されるよ !

プルの仕組み



✔️ これで、最新のデータが自分のパソコンに届くよ !

3. まとめ：チーム開発のいつもの流れ



💡 これだけで、チームで「お絵かき」を作り上げることができるよ！



GitHub応用編：安全に「合体」させるワザ

⚠️ プッシュとプルだけでは、危険なことがあります

なぜ危険なのか



直接絵を描くの like push



友達が「太陽」を描いてる最中、
自分が「太陽」を消しちゃうと...



ブランチ

完成品の絵をコピーして、自
分の作業場を作るよ



プルリク

「できたからチェックして！
合体させて！」とお願いする
よ



マージ

みんなで確認して、OKなら
メインの絵に安全に合体させ
るよ

これさえあれば、みんなの力を最大限合わせて、安全にお絵かきが完成するよ！

① ブランチ (Branch) : 「コピーして、自分の作業場でやる！」

ブランチとは、

「メインの完成品のお絵かき」を
コピーして、自分の作業場を作る方法

木の枝のように分かれて、安全に作業できる仕組みだよ。



コピーする

メインの絵 (プログラム) の completa なコピーを作れるよ！



自分の作業場

自分の好きなように編集しても、メインには影響しないよ！



安全に作業

失敗しても心配ない！元の絵はそのまま残ってるよ。



完成ったら合体

作業が終わったら、メインの絵に「マージ」して完成！

ブランチの例とポイント

💡 例：「町」の絵に「学校」を追加する



mainブランチ

完成品のお絵かき



学校追加用ブランチ

自分の作業場



失敗を恐れずに作業できる

メインの絵に直接影響を与えることなく、自分の作業を安全に進めることができます。



複数人が同時に働ける

他の人が「公園」や「病院」を描いている間に、自分が「学校」を描くことができるよ。







自分の作業場が必要

そっくりそのままコピー（複製）して、自分の作業場を作ることができます。

「ブランチ」は「枝」の意味で、木の幹から新しい枝が分かれるイメージ

② プルリクエスト (Pull Request / プルリク) : 「できた！みんな、チェックお願い！」

プルリクエストは、「**完成品をメインに合体させる前に、みんなにチェックしてもらうためのお願い**」だよ。

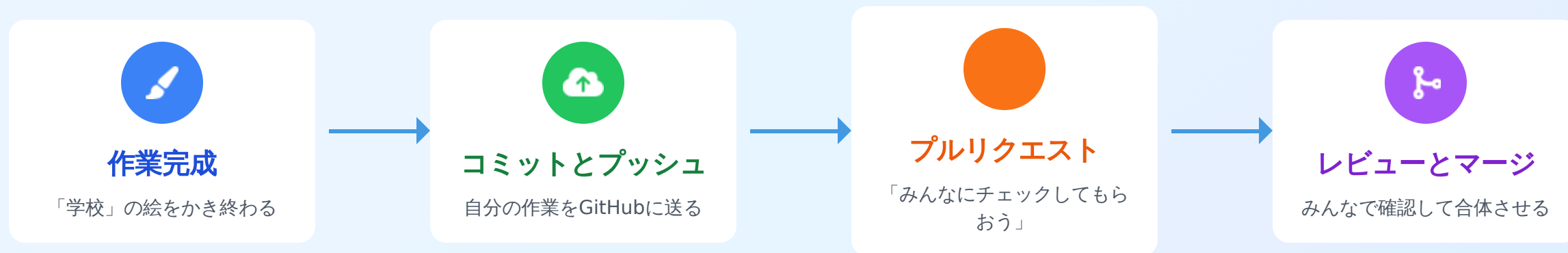
-  • 自分の「学校」の絵が完成した！
-  • 自分の作業用ブランチで作業完了
-  • GitHub上で「プルリクエスト」ボタンを押す
-  • 仲間たちに「変なところがないか、チェックして」とお願いする



💡 ポイント

- ✓ プルリクは「合体させてください！」という「お願い (Request)」だよ
- ✓ チームの他の人たちは、このプルリクを見てコメントすることができる
- ✓ 「いいね！」とか「あ、窓の色がちがうよ」とコメントしてもらえるよ

プルリクの例とポイント



💡 プルリクの例

「学校追加用」ブランチから
「mainブランチ」にプルリク

メッセージ: 「学校を描きました！変なところがないか、レビューをお願いします！」

✅ プルリクは「合体させてください！」という「お願い (Request)」です

★ レビュープロセスのポイント

- 💬 チームの他の人たちは、プルリクを見て「いいね！」とか「あ、窓の色がちがうよ」とコメントをすることができます
- 👍 プルリクが出たら、他の人がレビューするまで待つ時間ができます
- 👍 みんなのレビューが終わったら、リーダーが「マージ (合体)」ボタンを押すことで、メインの絵に safely 合体します

③ マージ (Merge) : 「OK！合体させよう！」

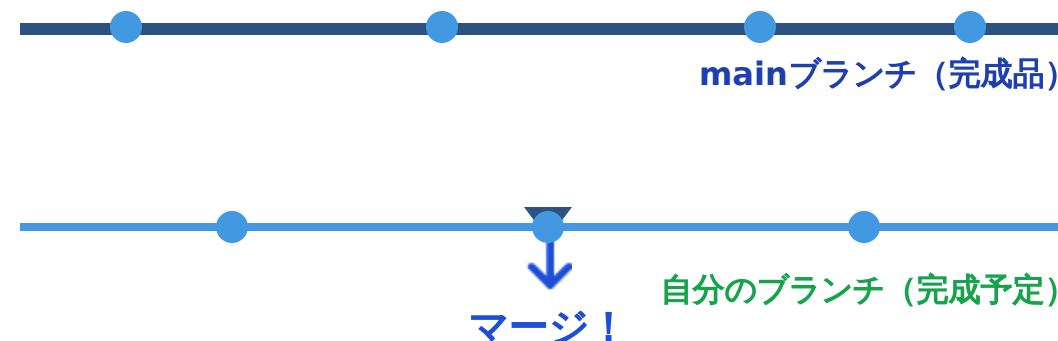
マージってなに？

プルリクでみんなのチェックが終わり、「OK」が出たら、

いよいよ「自分のブランチ (コピー)」を

「メインの絵 (完成品)」に合体させるよ！

💡 たとえば、「学校」の絵を描き終わったら、それを「町」の絵に合体させるんだよ！



✓ マージの手順

1. みんなが「いいね！」とOKを出す
2. 必要に応じて修正をプッシュする
3. リーダーが「マージ」ボタンを押す

マージの例とポイント

例え話

たとえば、みんなで作っている「町」の絵（mainブランチ）に、



+



「学校」を描いた人がいたとしよう。プルリクでレビューが通ったら、



リーダー →



マージ

リーダーが「マージ（合体）」ボタンを押すことで、安全に「町」の絵に「学校」が追加されるよ！

マージのポイント



レビューが終わったら

チームのみんなが「うん、OK!」と承認したらマージするよ



安全に合体できる

プルリクで確認が終わってからマージするので、ミスが少なくなるよ



リーダーがマージする

プルリクが承認されると、リーダーが「マージ」ボタンを押して完成させるよ

main

マージの仕組み

feature/school

マージ

4. まとめ：チーム開発の安全な流れ

GitHubを使うと、安全にチームで「お絵かき」を完成させることができます。以下の4つのステップで、チーム開発の基本的な流れを覚えましょう！



1. ブランチを作る

メインの絵をコピーして、自分の作業場を確保するよ！



2. 作業する

自分のブランチで絵を描き、コミットしてプッシュするよ！



3. プリクを送る

「できたからチェックして！合体させて！」とお願いするよ！



4. レビュー&マージする

みんなでチェックして、OKならメインの絵に合体させるよ！

★ これで、チーム開発の基本がマスターできました！ ★

みんなの力を合わせて、もっともっと大きな「お絵かき」を作り上げよう！

GitHubでチーム開発を楽しく学ぼう！