

! Git & Obsidian Vault トラブルシューティングまとめ

🚫 Quick Fix チートシート

1. 最新を取り込んでPush

```
git pull origin main --allow-unrelated-histories --no-rebase
git add .
git commit -m "chore: resolve merge conflicts"
git push
```

2. 不要ファイル除外 & .gitignore

```
cat <<EOF > .gitignore
.DS_Store
.obsidian/workspace.json
EOF
git rm --cached .DS_Store .obsidian/workspace.json
git clean -f
git add .gitignore
git commit -m "chore: ignore DS_Store and workspace"
git push
```

3. 未Gitリポジトリ

```
git init
git remote add origin https://github.com/stellacodex/codex-collective-
archive.git
git add . && git commit -m "Initial commit"
git push -u origin main
```

4. Upstream設定

```
git push --set-upstream origin main
```

5. nanoが面倒なら一発生成

```
cat <<EOF > .gitignore
.DS_Store
```

```
.obsidian/workspace.json
EOF
```



Gitコマンド徹底チートシート：状態別リカバリパターン



git add の種類

コマンド	意味・用途
git add .	カレントディレクトリ以下の 変更・新規追加 を全てステージング
git add -A	全体 の変更・追加・削除を包括的にステージング（-A は . より強い）
git add -u	変更+削除のみ をステージング（新規ファイルは無視）
git add ファイル名	単体ファイルだけをステージング



git commit -m の使い分け（メッセージ例）

```
git commit -m "feat: initial codexvault structure"
git commit -m "fix: hard overwrite to sync from local"
git commit -m "chore: force overwrite to match local state"
git commit -m "fix: cleanup hidden files and sync fully"
git commit -m "chore: remove .DS_Store from backup-before-force"
git commit -m "chore: sync with local state (remove obsolete files)"
```



git push のバリエーション

コマンド	意味
git push	設定済のリモートブランチにPush（初回以外）
git push origin main	明示的にmainブランチをPush
git push -u origin main	upstream（追跡元）も設定する（初回Push）
git push -f	強制Push （注意）



.DS_Store削除 & キャッシュ解除（macOS）

```
find . -name '.DS_Store' -type f -delete
```

```
git rm --cached .DS_Store
```

Git 状態別リカバリ手順（トラブル時の処方箋）

⚠️ ローカル変更 vs リモート不一致 → "fetch first" エラー

```
git pull origin main --allow-unrelated-histories --no-rebase
# コンフリクトを手動で解消しつつ、下記で同期：
git add .
git commit -m "fix: sync local changes"
git push
```

📶 Git状態の強制リセット + 再アップロード（初期化的）

```
# ① 既存Gitを無効化（オプション）
rm -rf .git

# ② Git再初期化 & リモート再設定
git init
git remote add origin https://github.com/USERNAME/REPO.git

# ③ 不要ファイル整理
git clean -f -d
git add .
git commit -m "chore: full reset and reupload"
git push -u origin main
```

⚠️ リモート履歴を削除して完全上書きしたい時（要注意）

```
git push origin main --force
```

💬 注意：共同作業者がいる場合は事前共有！

よくある現象とショートFix集

症状	対処法
<code>.DS_Store</code> がリモートに出てしまった	<code>.gitignore</code> 追加 + キャッシュ解除 + 削除Push
<code>not a git repository</code>	<code>git init</code> してリモート再接続
<code>Your branch has no upstream branch</code>	<code>git push -u origin main</code> で初期追跡設定

症状	対処法
Merge conflict が多発する	<code>git pull</code> → 手動解決 or <code>git push -f</code> で強制上書き (注意)
ローカル変更を全部反映させたい	<code>git add -A && git commit && git push</code>

✓ 最終リカバリ・フロー（迷ったら）

```
rm -rf .git
find . -name '.DS_Store' -type f -delete

# 再初期化
cat <<EOF > .gitignore
.DS_Store
.obsidian/workspace.json
EOF

git init
git remote add origin https://github.com/YOURNAME/YOURREPO.git
git add .
git commit -m "feat: fresh start"
git push -u origin main
```

✓ これで「もう何が起きてもGitで詰まらない」が目標！ エラーや事例が増えたら、このチートシートに追加していくね ■