

# 課題提出の仕方

---

- [課題提出の仕方](#)
  - [課題提出の流れ](#)
  - [コマンド一覧](#)
  - [注意事項](#)
  - [課題の一例](#)
  - [参考資料](#)

## 課題提出の流れ

1. LMS上の招待リンクをクリックしてリモート課題リポジトリを作成
2. リモート課題リポジトリをローカルにクローンする
3. VS Codeでローカル課題リポジトリを開く
4. READMEファイルの内容に沿って課題に取り組む
5. プログラムの動作確認をする
6. 変更内容をステージングする
7. ステージングした変更内容をコミットする
8. コミットした変更履歴をリモート課題リポジトリにプッシュする
9. ブラウザでリモート課題リポジトリが変更されているか確認して完了

## コマンド一覧

コマンド	作業内容
git clone <b>repository</b>	課題リポジトリ <b>repository</b> をローカルにコピーする
git status	リポジトリの状態を確認する
git add -A	全ての変更をステージングする
git commit -m "message"	メッセージ付きでコミットする
git push	リモートをローカルに同期する
git fetch	リモートとローカルの差分をダウンロードする
git pull	ローカルをリモートに同期する

## 注意事項

- gitのコマンド操作は、**ローカルリポジトリ上でgit bashを開いて**行ってください。
  - クローンするときとはディレクトリが違うので注意!
- リモートリポジトリとローカルリポジトリの**どちらの内容が新しいのか**、常に気にするようにしてください。
  - どちらが新しいか確認するには、fetchしてからstatusを調べてください。
  - リモートの方が新しければプル、ローカルの方が新しければプッシュするようにしましょう。
- 提出は期限内ならば何度でもやり直せます。
- 状態がよく分からなくなったらローカルリポジトリを削除してクローンからやり直してください。

## 課題の一例

例えば、「"Hello, World!"という文字列を出力するPythonコードを作成しなさい」という課題が出された場合、次のように**セル形式**で記述してください。

```
# %%  
print("Hello, World!")
```

必ずローカルで実行して、動作確認をしてからプッシュするようにしてください。

## 参考資料

より詳しい内容については[こちらのウェブサイト](#)を参照のこと。