

付録 演習問題

前準備

STEP1：リモートリポジトリの作成

GitHub にアクセスして、演習問題用のリモートリポジトリを作成してください。リポジトリ名は「git-train」としてください（Private で作成）。

GitHub

<https://github.com/login>

STEP2：リモートリポジトリのクローン

STEP1 で作成したリモートリポジトリを、Visual Studio Code でクローンしてください。保存するフォルダは「C:¥git」としてください。

リモートリポジトリへの反映

STEP1：ファイルの追加

Visual Studio Codeにて、以下の内容のファイル（local_made.txt）を作成してください。

```
<< Git Training >>
ファイルの追加
```

作成できたらステージ→コミット→プッシュを行い、リモートリポジトリにファイルが追加されていることを確認してください。コミットメッセージは「local_made.txt の作成」としてください。

STEP2：ファイルの変更

STEP1で作成したファイル（local_made.txt）の内容を以下のように変更してください。

```
<< Git Training >>
ファイルの追加
ファイルの変更
```

変更できたらステージ→コミット→同期を行い、リモートリポジトリのファイルが変更されていることを確認してください。コミットメッセージは「local_made.txt の変更」としてください。

STEP3：ファイルの削除

STEP2で変更したファイル（local_made.txt）を削除します。削除できたらコミット→同期を行い、リモートリポジトリのファイルが削除されていることを確認してください。コミットメッセージは「local_made.txt の削除」としてください。

リモートリポジトリからの受け入れ

STEP1：リモートリポジトリの内容を受け入れる

GitHub にて、以下の内容をファイル（remote_made.txt）を作成してください。

```
<< Git Training >>
GitHub にてファイルの追加
```

作成できたら、Visual Studio Code の Git Graph でローカルがリモートに対してひとつ古い状態であることを確認します。確認できたら同期を行い、ローカルとリモートが同じ状態であることを確認します。

STEP2：コンフリクト（競合）の発生

GitHub にて、STEP1 で作成したファイル（remote_made.txt）の内容を以下のように変更してください。

```
<< Git Training >>
GitHub にてファイルの追加
GitHub にてファイルの変更
```

変更できたら、Visual Studio Code にて、同ファイル（remote_made.txt）の内容を以下のように変更してください。

```
<< Git Training >>
GitHub にてファイルの追加
Visual Studio Code にてファイルの変更
```

変更できたら、ステージ→コミット→同期を行います。コミットメッセージは「remote_made.txt をローカルで変更」としてください。

ここでコンフリクト（競合）が発生します。

STEP3：コンフリクト（競合）の解決

Visual Studio Codeにて、同ファイル（remote_made.txt）の内容を以下のように変更してください（リモートでの変更点とローカルでの変更点を手動で統合）。

```
<< Git Training >>  
GitHubにてファイルの追加  
GitHubにてファイルの変更  
Visual Studio Codeにてファイルの変更
```

変更できたらステージ→コミット→同期を行い、リモートリポジトリのファイルの内容が上記のようになっていることを確認してください。コミットメッセージは「remote_made.txt のコンフリクト解決」としてください。

ブランチの操作

STEP1：ブランチの作成

Visual Studio Code の Git Graph で新しいブランチ（develop）を作成してください。
作成できたらブランチのプッシュを行い、リモートリポジトリに反映されていることを確認してください。

STEP2：作成したブランチにファイルを追加

Visual Studio Code にて、STEP1 で作成したブランチ（develop）に切り替え（チェックアウト）、以下の内容のファイル（dev.txt）を作成してください。

```
<< Git Training >>  
ブランチにファイルを追加
```

作成できたらステージ→コミット→プッシュを行い、Git Graph にて develop ブランチが一歩先に行っていることを確認してください。

STEP2：ブランチのマージ

Visual Studio Code にて、main ブランチに切り替え（チェックアウト）ます。
develop ブランチで「Marge into current branch（カレントブランチにマージ）」を行い、Git Graph で develop ブランチが main ブランチと合流していることを確認してください。
確認できたら同期を行い、リモートリポジトリでもブランチがマージされていることを確認してください。

STEP3：ブランチの削除

Visual Studio Code の Git Graph にて、develop ブランチで「Delete Branch（ブランチの削除）」を行い、リモートリポジトリでもブランチが削除されていることを確認してください。

演習問題 参照先

前準備 参照先

STEP1：リモートリポジトリの作成

テキスト（Git 入門）P38 から P39 を参照

STEP2：リモートリポジトリのクローン

テキスト P48 から P49 を参照

リモートリポジトリへの反映 参照先

STEP1：ファイルの追加

テキスト P53 から P54 を参照

STEP2：ファイルの変更

テキスト P50 から P53 を参照

STEP3：ファイルの削除

テキスト P54 から P55 を参照

リモートリポジトリからの受け入れ 参照先

STEP1：リモートリポジトリの内容を受け入れる

テキスト P59 から P61 を参照

STEP2：コンフリクト（競合）の発生

テキスト P63 から P64 を参照

STEP3：コンフリクト（競合）の解決

テキスト P64 から P65 を参照

ブランチの操作 参照先

STEP1：ブランチの作成

テキスト P66 から P68 を参照

STEP2：作成したブランチにファイルを追加

テキスト P68 から P69 を参照

STEP2：ブランチのマージ

テキスト P70 から P72 を参照

STEP3：ブランチの削除

テキスト P88 を参照