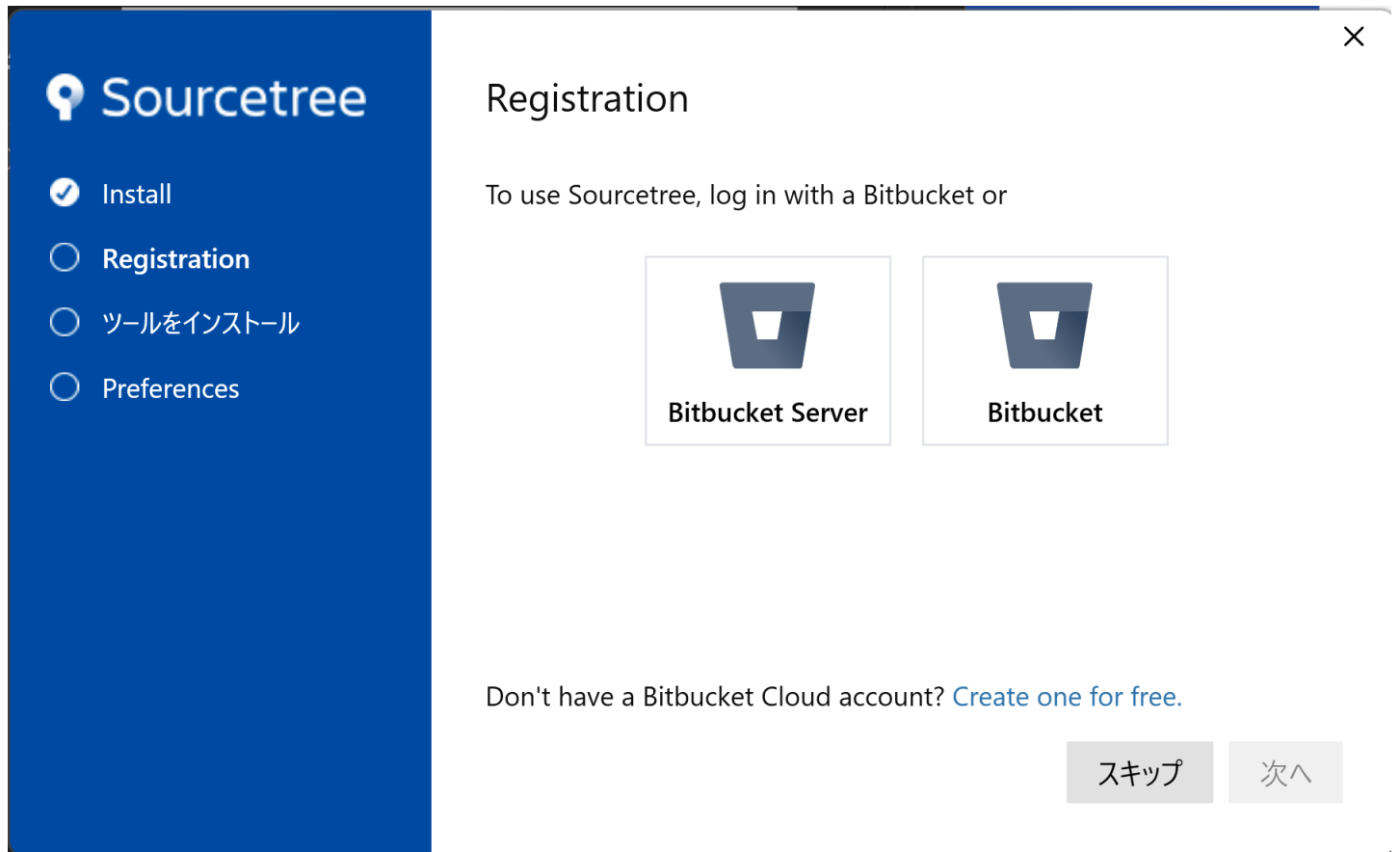


# 1. git 開始方法

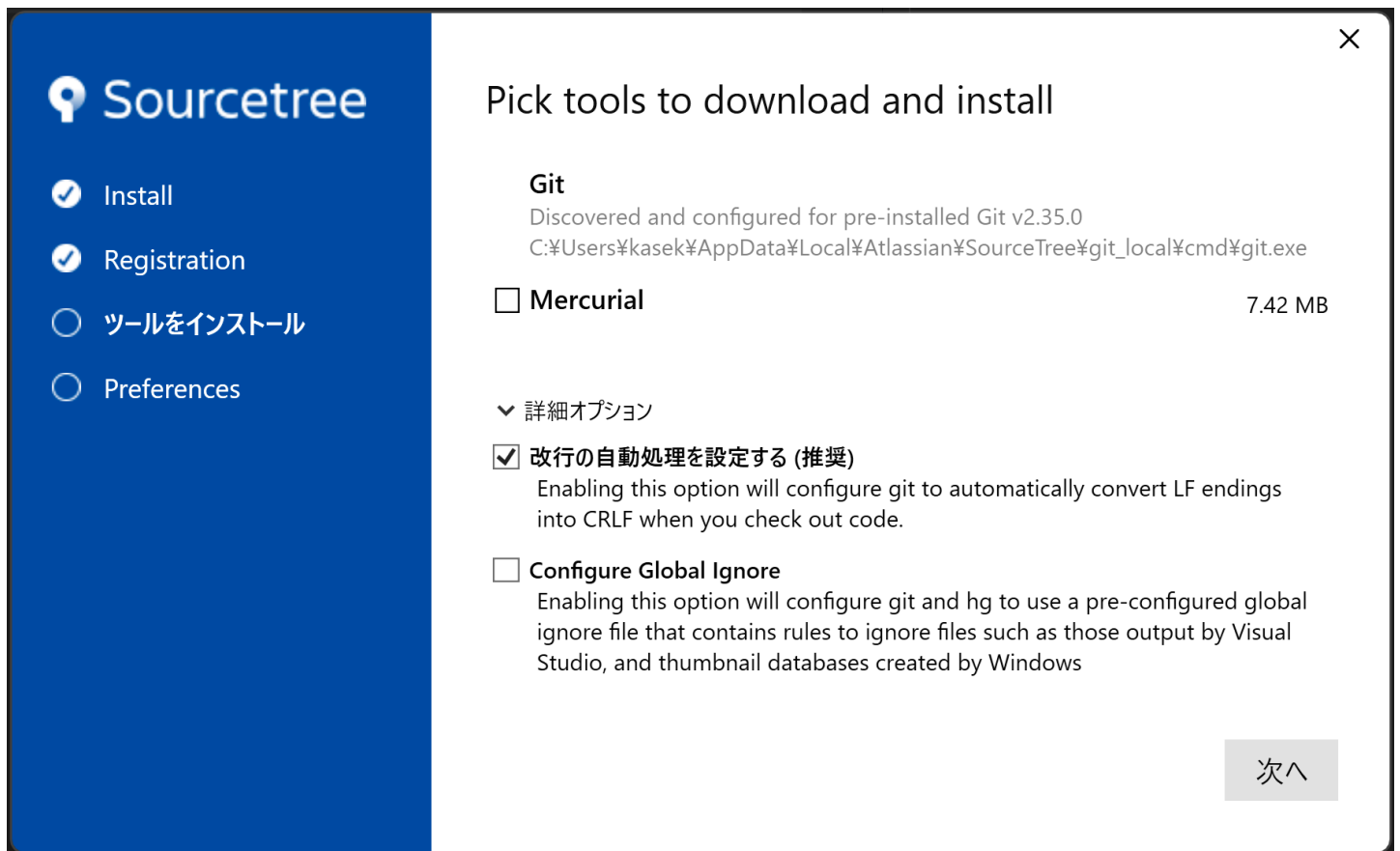
## 1.1. インストール

[git](#)をインストール

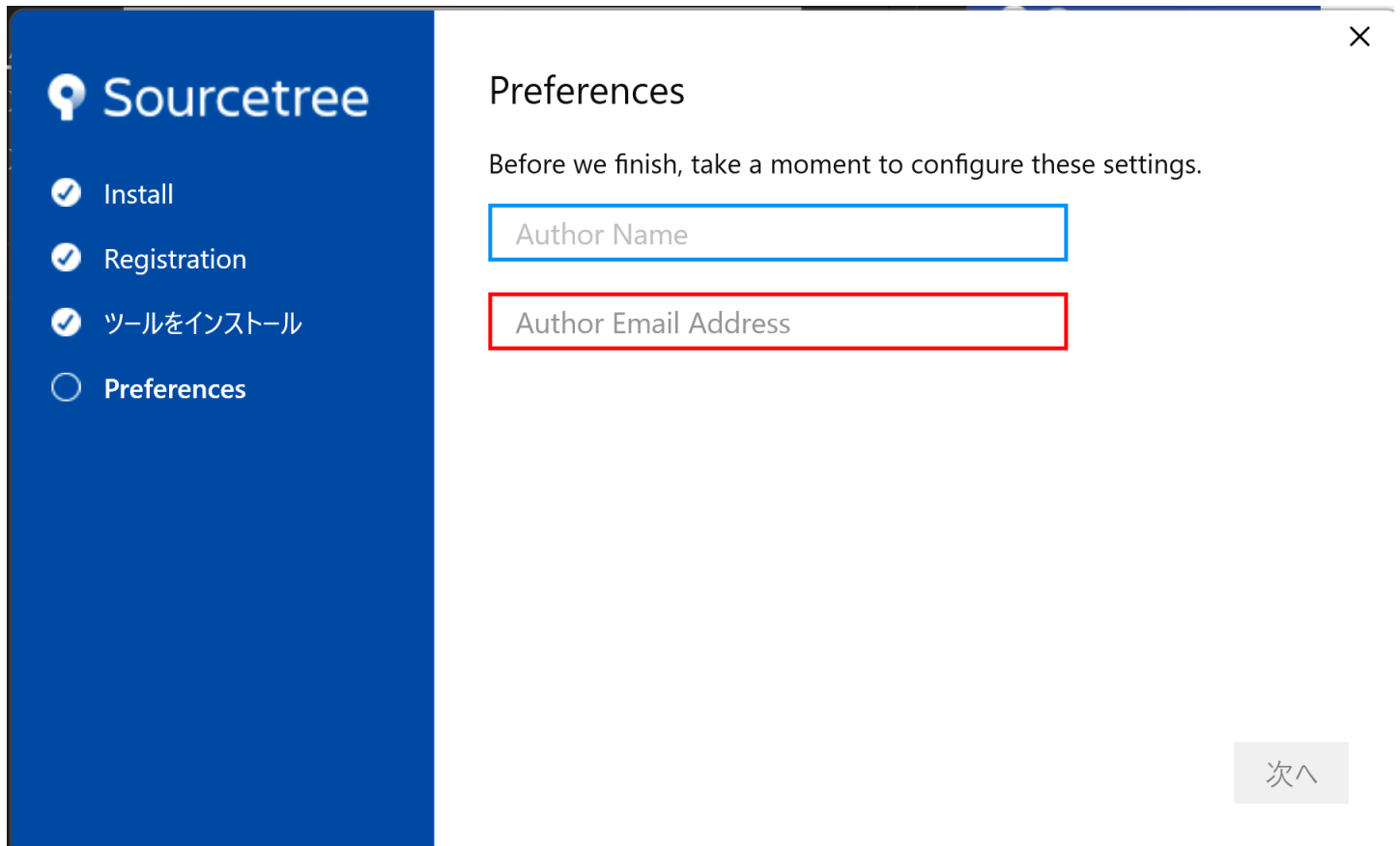
[sourcetree](#) をインストール



「スキップ」



写真のように設定し「次へ」



指示通り設定

## 1.2. アプリ設定

用語：本セクション中の

用語	意味
remote_user	リポジトリサーバ上のユーザー名 関口から指定されるユーザー名に置き換える
local_user	お使いのPC上のユーザー名。 この記事と異なっても特に意識しなくてOKな部分

### 1.2.1. SSHキー作成

インストール後

「ツール」=>「SSHキーの作成/インポート」=>「Generate」

マウスカーソルを適当に動かすとキーができる。

「Save private key」でprivatekey.ppkというファイル名で保存

「Save public key」で自分の名前+".pub"（私の場合sekiguchi.pub）をファイル名にして保存。

「ツール」=>「オプション」=>「全般」=>「SSHクライアントの設定」=>「SSHキー」

に上で作ったprivatekey.ppk を登録

上で作った.pubファイルを関口に送り、サーバーに登録してもらう。

### 1.2.2. ppk登録

public key 登録後、研究室LAN内でコマンドプロンプトで以下を実行

```
> cd path_to_ppk
> "%UserProfile%\AppData\Local\SourceTree\app-3.4.14\tools\putty\plink.exe" -ssh -C -i "privatekey.ppk" remote_user@192.168.100.
```

「path\_to\_ppk」は上でppkファイルを保存したフォルダへのパス

「app-3.4.14」の部分はインストールしたsourcetree のバージョン

二行目のコマンドは長くPDFでは見切れてしまうので以下に同じものを掲載する。1行のコマンドとして実行する必要がある。

```
"%UserProfile%\AppData\Local\SourceTree\app-3.4.14\tools\putty\plink.exe" -ssh -C -i "privatekey.ppk"
```

[remote\\_user@192.168.100.90](#)

実行時にy/n と聞かれるのでy と入力しenter

```
remote_user@phoebe:~$
```

と表示されたら成功。表示されなければ関口に相談する。

### 1.2.3. リモートリポジトリのクローン

sourcetreeの「Clone」で以下を設定し「クローン」を押す

# Clone

Cloning is even easier if you set up a remote account

[参照](#)

リポジトリタイプ:  これは Git リポジトリです

[参照](#)

Local Folder:

› 詳細オプション

クローン

クローンするフォルダ名はわかりやすいように変更してもOK

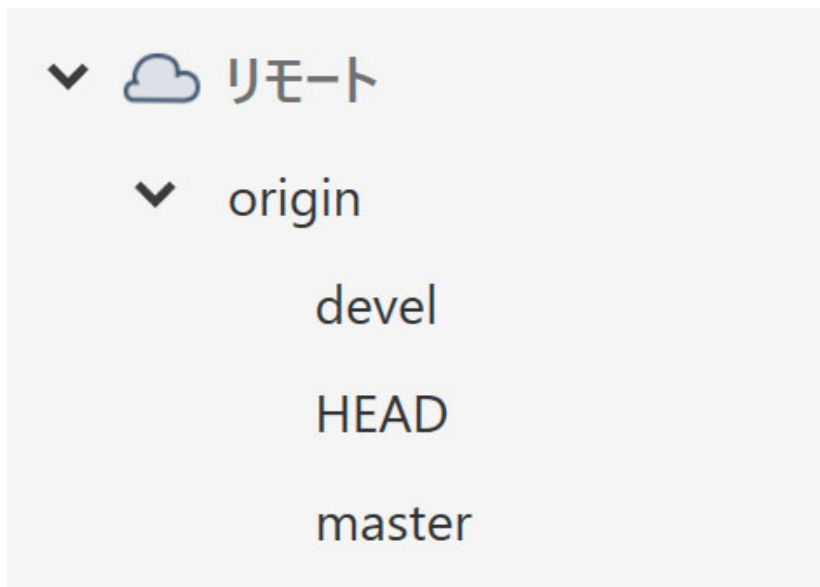
例 : Documents\program => Documents\collab

「クローン」ボタンが押せない場合はppkの設定がちゃんとできているか確認する。

注意 : SSHキーに設定するppkとpubはペアなので、SSHキーをGenerateし直した場合、pubキーを登録しなおす（関口送りなおす）必要がある。

## 1.2.4. devel ブランチへ切り替え

写真のように開き, develをダブルクリックしてdevelブランチに切り替える.

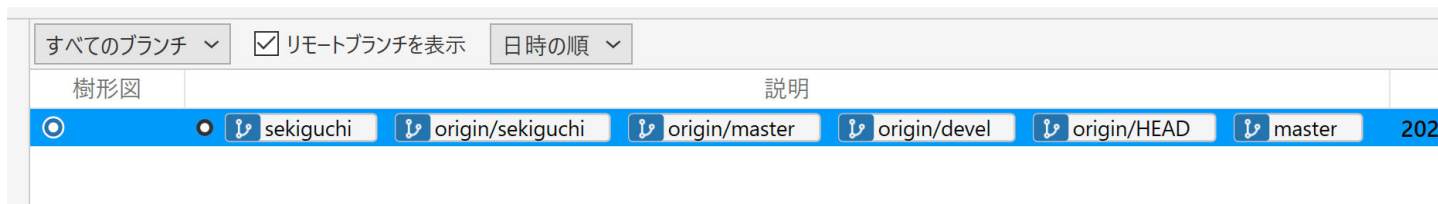


### 1.2.5. 個人開発用ブランチ作成

次に「ブランチ」を押し、新規ブランチに自分の名前（私の場合sekiguchi）を設定し「ブランチを作成」

「プッシュ」を押し、今作ったブランチを対象にセットし「プッシュ」

「リモートブランチを表示」のチェックを入れ、写真のようにorigin/名前（私の場合 origin/sekiguchi）が見えていればOK



以上で開発のための準備が完了

## 1.3. 運用ルール

各ブランチの運用ルールは以下の通り

個人名ブランチは変更の塊ごとにcommitし、毎日pushする。

devel はエラー無く回る状態のものを週 1 程度を目処に個人名ブランチからmergeする。

1 週以上エラーが取れないときは相談すること。

master は大きな区切り（学会発表や公聴会など）毎にdevelからmergeする。

## 1.4. 使い方

ファイルを変更すると写真のように作業ツリーのファイルに変更したファイルが表示される。



1. 「全てインデックスに追加」を押すとIndexにステージしたファイルに移動する。
2. この状態で「コミット」を押すとコミットのためのコメント記入欄が出てくるので、どんな変更をしたのかメモを残し「コミット」ボタンを押す。
3. 「プッシュ」でサーバーに変更を登録

## 2. 関口用備忘録

config/manual 参照

```
ssh-keygen -i -f id_rsa.conan >> authorized_keys
```

とすることでputty形式のキーを追加できる。

リポジトリを設定するときは

```
> mkdir program
> git init --bare
```

ユーザー登録

```
sudo adduser usern
```

パスワード生成はmanualフォルダ参照

## ユーザー削除

```
sudo userdel -r usern
```