

Django应用講座 2

(Login処理)

Class Based View (Viewを用いたログイン、ログアウト処理)

ログイン処理をViewを用いて、ログインをする場合、以下のように記述する

```
from django.views.generic import View
```

```
class LoginView(View):
```

```
    def post(self, request, *args, **kwargs):
```

```
        username = request.POST['username'] # usernameを取得
```

```
        password = request.POST['password'] # passwordを取得
```

```
        user = authenticate(username=username, password=password) # userが存在するか確認
```

```
        if user is not None:
```

```
            if user.is_active:
```

```
                login(request, user) # ログイン処理
```

```
        return render(request, "index.html")
```

```
class LogoutView(View):
```

```
    def get(self, request, *args, **kwargs):
```

```
        logout(request) # ログアウト処理
```

```
        return HttpResponseRedirect(settings.LOGIN_URL)
```

Class Based View (LoginRequiredMixin)

ログインが必要なViewに付与するLoginRequiredMixinやmethod_decoratorを用いる
(<https://docs.djangoproject.com/ja/3.1/topics/auth/default/#the-loginrequired-mixin>)

```
class MyView(LoginRequiredMixin, View): # ログインが必要なViewに継承させる
    login_url = '/login/'
    redirect_field_name = 'redirect_to'
```

<https://docs.djangoproject.com/ja/3.1/topics/class-based-views/intro/#decorating-the-class>

```
class ProtectedView(TemplateView):
    @method_decorator(login_required)
    def dispatch(self, request, *args, **kwargs):

    # ログインが必要なViewにデコレータを付ける
    @method_decorator(login_required, name='dispatch')
class ProtectedView(TemplateView):
    pass
```

Class Based View (LoginView, LogoutView)

<https://docs.djangoproject.com/en/3.1/topics/auth/default/>

django.contrib.auth.views.LoginView: **Login**をするためのView

template_name: 表示するテンプレートを指定

authentication_form: ログイン認証に用いられるFormを指定(デフォルトはAuthenticationForm)

django.contrib.auth.forms.AuthenticationForm: ログイン認証に用いられるForm

django.contrib.auth.views.LogoutView: **Logout**をするためのView

settings.pyに設定する内容

LOGIN_REDIRECT_URL: LoginViewで次に遷移する先が指定されていなかった場合に遷移する遷移先のURL

LOGIN_URL: ログインしていない状態でlogin_requiredが指定されているViewを表示しようとした場合にリダイレクトされるView

LOGOUT_REDIRECT_URL: LogoutViewで次に遷移する先が指定されていなかった場合に遷移する遷移先のURL

Class Based View（ログインを記憶するボタンを追加）

<https://docs.djangoproject.com/en/3.1/topics/http/sessions/>

ログインを記憶するボタンを追加するには、セッションの保存時間を変更するとよい。

つまり、デフォルトのセッション時間を10分などの短い時間にして

↓

「ログインを記憶する」ボタンを押すと、セッション時間を1日などに変更する

settings.py

SESSION_COOKIE_AGE: セッションの保存時間（秒）。デフォルト、1209600(2週間)

https://docs.djangoproject.com/ja/3.1/ref/settings/#std:setting-SESSION_COOKIE_AGE

request.session.set_expiry(value): セッションの保存時間を引数の時間(秒)に変更する。引数が0の場合、ブラウザを閉じるとセッションがなくなる。もし value が datetime または timedelta オブジェクトならば、指定された日時に破棄される

https://docs.djangoproject.com/ja/3.1/topics/http/sessions/#django.contrib.sessions.backends.base.SessionBase.set_expiry