

PythonのフレームワークであるDjangoのAdminサイトのプロパティについて説明します。

実行環境

- python3.6
- Django 2.0
- postgresql 10

登場するクラス

- AdminSite
 - Adminサイトを作成できます。
 - このクラスを拡張することでAdminサイト自身の挙動を変更することができます。
- ModelAdmin
 - Modelと紐付いたAdminサイトの画面を作成できます
 - このクラスを拡張することで、Adminサイトの挙動を変更することができます。

AdminSiteのカスタマイズ方法

```
from django.contrib.admin import AdminSite
from .model import MyModel

class MyAdminSite(AdminSite):
    # サイトのヘッダーを設定できます。
    # 初期値 : "Django administration"
    site_header = 'Monty Python administration'

    # サイトのタイトルを設定できます。
    # 初期値 : "Django site admin"
    site_title = 'Python title'

    # Admin画面の右上"View site"のリンクになります。
    # Noneを設定することで右上"View site"のリンクを削除することができます。
    # 初期値 : /
    site_url = None

    # 画面上部のタイトルを変更できます。
    # 初期値 : "Site administration"
    index_title = 'custom administration'

    # ログイン後の画面のテンプレートを設定できます。
    index_template = '...'

    # アプリ毎の画面のテンプレートを設定できます。
    app_index_template = '...'

    # データリスト画面で値がない時に表示する文字を設定できます。
```

```
# ModelAdminでも設定できますが、AdminSiteで設定することで
# すべてのModelで共通値とすることができます。
# 初期値 : ”_“
empty_value_display = ‘?’

# ログイン画面のテンプレートを設定できます。
login_template = ‘...’

# ログイン画面のFormクラスを設定できます。
# 初期値 : AuthenticationForm
login_form = AuthenticationForm

# ログアウト画面のテンプレートを設定できます。
logout_template = ‘...’

# パスワード変更画面のテンプレートを設定できます。
password_change_template = ‘...’

# パスワード変更後画面のテンプレートを設定できます。
password_change_done_template = ‘...’

admin_site = MyAdminSite(name=‘myadmin’)
admin_site.register(MyModel)
```

admin.py hosted with ♥ by GitHub

[view raw](#)

```
from django.urls import path

from myapp.admin import admin_site, advanced_site

urlpatterns = [
    path('myadmin/', admin_site.urls),
    # このように複数のAdminSiteを作成することもできます。
    path('advanced_admin/', advanced_site.urls),
]
```

urls.py hosted with ♥ by GitHub

[view raw](#)

yuyan7sh [2年前](#)