Puoi creare uno script Python che esegue automaticamente le operazioni di reimpostazione del nome utente e della password per un account specifico. Questo script può essere eseguito utilizzando il comando python manage.py runscript se utilizzi il pacchetto django-extensions. Ecco come puoi farlo:

**Passi per creare lo script**

1. **Installa django-extensions**: Se non lo hai già fatto, installa il pacchetto django-extensions nel tuo ambiente Django:

bash

Copia codice

pip install django-extensions

1. **Aggiungi django-extensions alle tue app installate**: Apri il file settings.py del tuo progetto e aggiungi 'django\_extensions' alla lista INSTALLED\_APPS:

python

Copia codice

INSTALLED\_APPS = [

# altre app

'django\_extensions',

]

1. **Crea la directory per gli script**: Crea una directory chiamata scripts nella tua applicazione principale (può essere la stessa directory dove si trovano i file models.py e views.py).
2. **Crea lo script Python**: All'interno della directory scripts, crea un file chiamato reset\_admin.py e aggiungi il seguente codice:

python

Copia codice

from django.contrib.auth.models import User

def run():

# Configura le nuove credenziali

nuovo\_nome\_utente = 'nuovo\_nome\_utente'

nuova\_password = 'nuova\_password'

nuova\_email = 'nuova\_email@example.com'

# Trova o crea l'utente amministratore

user, created = User.objects.get\_or\_create(username=nuovo\_nome\_utente)

# Aggiorna le informazioni dell'utente

user.set\_password(nuova\_password)

user.email = nuova\_email

user.is\_superuser = True

user.is\_staff = True

user.save()

if created:

print(f"Creato nuovo superuser: {nuovo\_nome\_utente}")

else:

print(f"Superuser aggiornato: {nuovo\_nome\_utente}")

**Esecuzione dello script**

Ora puoi eseguire questo script utilizzando il comando runscript fornito da django-extensions. Apri il terminale e naviga nella directory del tuo progetto Django, quindi esegui il comando:

bash

Copia codice

python manage.py runscript reset\_admin

**Spiegazione del codice**

* **Importa il modello User**: Importa il modello User di Django.
* **Configura le nuove credenziali**: Imposta le nuove credenziali dell'utente amministratore.
* **Trova o crea l'utente amministratore**: Utilizza get\_or\_create per trovare l'utente con il nome utente specificato o crearne uno nuovo se non esiste.
* **Aggiorna le informazioni dell'utente**: Reimposta la password, l'email e i permessi dell'utente.
* **Stampa un messaggio**: Indica se l'utente è stato creato o aggiornato.

Questo script ti permetterà di reimpostare facilmente il nome utente e la password per l'account amministratore specificato ogni volta che lo esegui.