

[🏠 ACCUEIL](#)[APPRENDRE PYTHON ▼](#)[CERTIFICATIONS](#)[WEB ▼](#)[JAVA](#)[TICE-MULTIMÉDIA](#)[GLOSSAIRE INFORMATIQUES](#)[BOUTIQUE](#)[CONTACT](#)[RECHERCHER](#)

**Hébergement
web 0.99€ /
mois**

Hebergeur Discount

Où



Authentification Utilisateur En Django

📅 23 MAI 2021 👤 PYTHONTECH 📁 LE FRAMEWORK DJANGO 💬 LEAVE A COMMENT

1 - UserCreationForm

Le framework Django fournit un formulaire d'authentification grâce à une classe nommé UserCreationForm (qui hérite de la classe ModelForm) pour gérer la création de nouveaux utilisateurs. Il comporte trois champs à savoir **nom d'utilisateur**, **mot de passe1** et **mot de passe2** (pour la confirmation du mot de passe). Pour utiliser UserCreationForm, vous devez d'abord l'importer depuis django.contrib.auth.forms comme suit:

```
1 from django.contrib.auth.forms import UserCreationFo
```

2 - Création d'une application users

Avant de commencer, nous devons:

1. Créer un **projet django**, appelons le **mysite** à titre d'exemple
2. Créer une **application**, nommons là **users**
3. Créer une zone **admin** (**superuser**)
4. **Enregistrer** l'application users au niveau du fichier **settings.py**

Hébergeur wo 0.99€ .fr inclu

Stockage SSD | Wordpress 1 clic |
Adresses Emails | Société Française



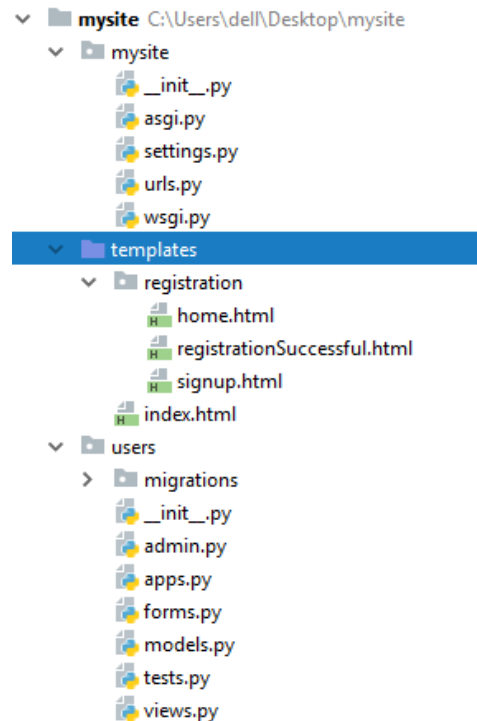
Hebe



Cours Python

5. Créer un nouveau répertoire **template/registration** au sein duquel 3 fichiers: **home.html**, **registrationSuccessful.html**, **signup.html**

◆ Aperçu du projet



◆ Mettre à jours le fichier settings.py

Dans **settings.py**, nous ajouterons l'application utilisateurs et utiliserons la configuration **AUTH_USER_MODEL** pour indiquer à Django d'utiliser notre nouveau modèle utilisateur personnalisé à la place du modèle

Les bases en Python

1. Introduction au langage Python
2. Ide Python
3. Premier programme avec IDLE
4. Premier programme avec Wing
5. Les variables en Python
6. Les commentaires en Python
7. Les opérateurs Python
8. Les fonctions en Python
9. Structure If else... Python
10. For While Python
11. Les Chaines De Caractères En Python
12. Formatage des chaines Python
13. Les listes en Python
14. Les tuples en Python

Autres Cours

Programmation & Database

1. Liste des cours Java
2. Base de données

utilisateur intégré. Nous appellerons notre modèle d'utilisateur personnalisé **CustomUser**.

Dans **INSTALLED_APPS**, ajoutez **'users.apps.UsersConfig'**. Ensuite, en bas du fichier , ajoutez la configuration **AUTH_USER_MODEL**.

```
1 # mysite/settings.py
2 INSTALLED_APPS = [
3     'users.apps.UsersConfig',
4     'django.contrib.admin',
5     'django.contrib.auth',
6     'django.contrib.contenttypes',
7     'django.contrib.sessions',
8     'django.contrib.messages',
9     'django.contrib.staticfiles',
10 ]
```

3 - Création du formulaire d'enregistrement via la classe UserCreationForm

Créons maintenant un fichier **users/forms.py** et insérons au sein duquel le code suivant:

```
1 # users/forms.py
2 from django import forms
3 from django.contrib.auth.models import User
```

Nous rejoindre sur Facebook

Hébergement web 0.99€ / mois

Hebergeur Discount

```
4 from django.contrib.auth.forms import UserCreationF
5
6 class UserRegisterForm(UserCreationForm):
7     email = forms.EmailField()
8
9     class Meta:
10         model = User
11         fields = ['username', 'email', 'password1',
```

4 - Création du view de l'application

Nous créons ensuite le view de l'application qui comporte:

1. une méthode pour l'affichage de la page d'accueil des utilisateurs
2. une méthode pour la redirection après enregistrement
3. une méthode pour le formulaire d'enregistrement des utilisateurs

```
1 # users/views.py
2 from django.shortcuts import render, redirect
3 from .forms import UserRegisterForm
4
5 # page d'accueil des utilisateurs
6 def homeUsers(request):
7     return render(request, 'registration/home.html')
8
9 # page de redirection apres enregistrement des util
```



```

10 def successful(request):
11     return render(request, 'registration/registrati
12
13 # page d'enregistrement des utilisateurs
14 def signup(request):
15     # on teste s'il y a des requetes envoyee par la
16     if request.method == 'POST':
17         form = UserRegisterForm(request.POST)
18         if form.is_valid():
19             form.save()
20             username = form.cleaned_data.get('username')
21             return redirect('/successful')
22     else:
23         form = UserRegisterForm()
24     return render(request, 'registration/signup.htm

```

5 - Configuration des urls

Nous devons maintenant configurer les urls et leurs chemins d'accès au niveau du fichier mysite/urls:

```

1 # mysite/urls
2 from django.contrib import admin
3 from django.urls import path
4 from users import views

```

Hébergement web 0.99€ / mois

Hebergeur Discount

Utilisateurs En Ligne !

94 Users En ligne

Utilisateurs/utilisatrices: [Addi](#),
[genesis_119](#), [PythonTech](#), [amornes](#),
[habakuk03](#), 87 Guests, 2 Bots

Se connecter

Identifiant ou adresse e-mail

```
5
6 urlpatterns = [
7     path('admin/', admin.site.urls),
8     path('homeUsers/' , views.homeUsers),
9     path('signup/' , views.signup , name = 'signup')
10    path('successful' , views.successful , name = '
11    ]
```

6 - Les fichiers du template

6.1 Le fichier home.html

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head><meta charset="UTF-8">
4     <title>Home Users</title>
5 </head>
6 <body>
7 <h2>Welcome to users home page!</h2>
8
9 <a href="{% url 'signup' %}">Signup</a>
10 </body>
11 </html>
```

6.2 Le fichier signup.html

Mot de passe

☐ Se souvenir de moi

SE CONNECTER

[Inscription](#)

[Mot de passe oublié ?](#)



```
1 <!-- signup.html -->
2 <!DOCTYPE html>
3 <html lang="en">
4 <head>
5     <meta charset="UTF-8">
6     <title>Title</title>
7 </head>
8 <body>
9 <h2>Registration form</h2>
10 <form method="POST">
11     {% csrf_token %}
12     <fieldset class="form-group">
13         {{ form.as_p }}
14     </fieldset>
15     <div class="form-group">
16         <button class="btn btn-outline-info" type="submit">Submit</button>
17     </div>
18 </form>
19 </body>
20 </html>
```

6.3 Le fichier de redirection registrationSuccessful.html

```
1 <!-- registrationSuccessful.html -->
```




```
2 <!DOCTYPE html>
3 <html lang="en">
4 <head>
5     <meta charset="UTF-8">
6     <title>Title</title>
7 </head>
8 <body>
9 <h2>Registration form</h2>
10 <form method="POST">
11     {% csrf_token %}
12     <fieldset class="form-group">
13         {{ form.as_p }}
14     </fieldset>
15     <div class="form-group">
16         <button class="btn btn-outline-info" type="submit">Submit</button>
17     </div>
18 </form>
19 </body>
20 </html>
```

Et finalement en démarrant le serveur et en tapant l'adresse url:

<http://127.0.0.1:8000/signup/> , on obtient la vue suivante:



Registration form

Username:

Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

Email:

Password:

- Your password can't be too similar to your other personal information.
- Your password must contain at least 8 characters.
- Your password can't be a commonly used password.
- Your password can't be entirely numeric.

Password confirmation:

Enter the same password as before, for verification.

Register

7 - Amélioration du design du formulaire d'authentification Django à l'aide du framework Bootstrap

Vous avez sans doute remarqué que le formulaire d'enregistrement est n'est pas très agréable! Que faut il faire pour améliorer son look ? Et bien le problème est simple, il suffit pour cela d'utiliser le template starter du framework bootstrap!

Commencez alors par **installer le module** suivant sur votre **virtuel environnement** à l'aide de l'utilitaire pip:

```
1 pip install Django-crispy-forms
```



et **enregistrez** ensuite l'application au niveau du fichier **settings.py**:

```
1 #settigns.py
2 INSTALLED_APPS = [
3     'crispy_forms',
4     'users.apps.UsersConfig',
5     'django.contrib.admin',
6     'django.contrib.auth',
7     'django.contrib.contenttypes',
8     'django.contrib.sessions',
9     'django.contrib.messages',
10    'django.contrib.staticfiles',
11 ]
```

Ajoutez ensuite la ligne de code suivante à la fin du **fichier settings.py**

```
1 # settings.py
2 ....
3 ....
4 CRISPY_TEMPLATE_PACK = 'bootstrap4'
```

Intégrez ensuite le framework bootstrap starter en ajoutant le code suivant à votre fichier signup.html ou votre fichier de base de template si vous utilisez l'héritage de template:

```
1 <link href="https://cdn.jsdelivr.net/npm/bootstrap@5
```



```
2 <script src="https://cdn.jsdelivr.net/npm/bootstrap@
```

Le code peut être télécharger via [cette adresse](#)

Ensuite à la tête de votre code, ajoutez la ligne :

```
1 {% load crispy_forms_tags %}
```

remplacez ensuite:

```
1 {{ form.as_p }}
```

par:

```
1 {{ form|crispy }}
```

et placez le code du formulaire dans une table html pour limiter les dimensions. Voici le code finale du fichier signup.html:

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4     <meta charset="UTF-8">
5     <title>Title</title>
6 <!-- Latest compiled and minified CSS -->
7 <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
8
```



```
9 <!-- Optional theme -->
10 <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css">
11
12 <!-- Latest compiled and minified JavaScript -->
13 <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootstrap.min.js"></script>
14 </head>
15 <body>
16 {% load crispy_forms_tags %}
17 <table border="0" width = "400" height = 400>
18     <tr>
19         <td style = "padding-left:25px;">
20             <h1>Register Form</h1>
21             <form method="POST">
22                 {% csrf_token %}
23                 <fieldset class="form-group">
24                     {{ form|crispy }}
25                 </fieldset>
26                 <div class="form-group">
27                     <button class="btn btn-outline-info" type="submit">Register</button>
28                 </div>
29             </form>
30
31         </td>
32     </tr>
33
```



```
34     </table>
35 </body>
36 </html>
```

Voici l'aspet final du formulaire:

Register Form

Username*

Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

Email*

Password*

- Your password can't be too similar to your other personal information.
- Your password must contain at least 8 characters.
- Your password can't be a commonly used password.
- Your password can't be entirely numeric.

Password confirmation*

Enter the same password as before, for verification.

Register



Younes Derfoufi

CRMEF OUJDA

 Post Views: 3 289

[< Previous Post](#)

[Next Post >](#)

Leave a Reply

Your email address will not be published. Required fields are marked *



Comment *

Name *

Email *

Website

☐ Enregistrer mon nom, mon e-mail et mon site dans le navigateur pour mon prochain commentaire.

CAPTCHA *



Saisissez le texte affiché ci-dessus:

êtes vous un robot ? (Obligatoire)



POST COMMENT

déposé Copyright France.com

© 2023 Très Facile | Wordpress_Theme: by CrestaProject.

