

Python - Kivy : Hello World et un peu plus

Le programme personne

- Crée un dossier **personne** ;
- Dans ce dossier crée un fichier **main.py** avec le contenu suivant : <code python main.py> # -- coding: utf-8 -- from kivy.app import App from kivy.uix.gridlayout import GridLayout from kivy.uix.label import Label from kivy.uix.textinput import TextInput

```
class Saisie(GridLayout):
    def __init__(self, **kwargs):
        super(Saisie, self).__init__(**kwargs)
        self.cols = 2
        self.add_widget(Label(text='Nom'))
        self.nom = TextInput(multiline=False)
        self.add_widget(self.nom)
        self.add_widget(Label(text='prénom'))
        self.prenom = TextInput(multiline=False)
        self.add_widget(self.prenom)

class Personne(App):
    def build(self):
        return Saisie()

if __name__ == 'main':
    Personne().run()

</code>
```

Exécution du programme

- Lance l'exécution du fichier **main.py** depuis Spyder ou en ligne de commande :

```
python main.py
```

- un fenêtre s'ouvre et contient un bouton cliquable sur lequel est inscrit **“Hello World !”** qui couvre toute la fenêtre

Explication

Ta fenêtre va contenir :

- 2 champs de saisie et deux labels pour indiquer ce qu'il faut saisir.
- une grille de 2 colonnes pour permettre de placer correctement les différents widgets

Quelques explications :

- **from kivy.uix.gridlayout import GridLayout**
 - importation d'un Widget GridLayout qui sera la classe de base du widget principal Saisie :
 - class Saisie(GridLayout):
- **super(Saisie, self).init(%%%kwargs)** : * la fonction (méthode) spéciale **init** permet de construire l'objet Saisie en modifiant (surchargeant) la classe de base GridLayout afin d'ajouter les widgets et leur définir leur comportement <code python> def init(self, kwargs): super(Saisie, self).init(kwargs) </code> Les liens Kivy pour connaître les propriétés de ces trois objets : * **Les boutons * Les labels * Les TextInput** ===== Prendre en compte la saisie du prénom ===== * ajout d'un bouton * ajout d'un label pour afficher le nom avec le prénom <code python> personne2.py> # -- coding: utf-8 -- from kivy.app import App from kivy.uix.gridlayout import GridLayout from kivy.uix.label import Label from kivy.uix.textinput import TextInput from kivy.uix.button import Button class Saisie(GridLayout): def init(self, kwargs): super(Saisie, self).init(kwargs) self.cols = 2 self.addwidget(Label(text='Nom')) self.nom = TextInput(multiline=False) self.addwidget(self.nom) self.addwidget(Label(text='prénom')) self.prenom = TextInput(multiline=False) self.addwidget(self.prenom) self.bouton = Button(text='Afficher') self.addwidget(self.bouton) self.bouton.bind(onpress=self.afficher) self.nomprenom = Label() self.add_widget(self.nomprenom) def afficher(self, instance): self.nomprenom.text = self.nom.text + " " + self.prenom.text class Personne(App): def build(self): return Saisie() if __name__ == 'main': Personne().run() </code> ===== Retour au menu de ma première application =====

o Première application

From:

[/ - Les cours du BTS SIO](#)

Permanent link:

[/doku.php/dev/python/kivy/app1/helloworld2](#)

Last update: **2017/05/04 14:21**

