

Modelli e Migrazioni

Qualche dettaglio in più

Il DB dell'app “gestione” del progetto “biblio”

E' stato oggetto delle lezioni precedenti.

Ci siamo soffermati sulla creazione del database e le sue modalità di accesso, sia in lettura che in scrittura.

Il DB biblio.gestione è veramente “scarno”.

Proviamo a complicarlo un po...

Creiamo un altro progetto: biblio2

Creiamo un altro progetto “biblio2” e facciamo gli stessi passaggi per la creazione ed inizializzazione del DB della sua app “gestione” che abbiamo visto nella prima versione di biblio.

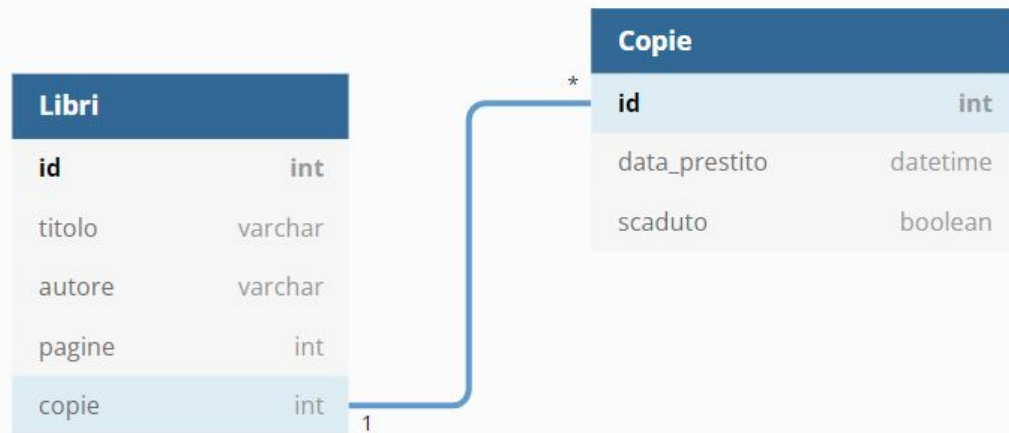
Creo un progetto separato, così da non dover modificare il codice visto per biblio “versione 1”.

Quindi per biblio2, creo l'app gestione. Creo il Modello libro con gli stessi metodi ed attributi e lo stesso comando per “riempirlo” di libri

Ragioniamo sulla tabella Libro

E' troppo semplice per la gestione di una biblioteca.

Rendiamolo un attimino più realistico.



Abbiamo riorganizzato alcune cose.

2 Tabelle. Libro e Copia.

Un libro ha N copie.

E' la copia ad avere una data dell'ultimo prestito. Inoltre, quest'ultimo può essere scaduto.

in gestione/models.py

```
class Libro(models.Model):
    titolo = models.CharField(max_length=200)
    autore = models.CharField(max_length=50)
    pagine = models.IntegerField(default=100)

    def __str__(self):
        out = self.titolo + " di " + self.autore + " ha " + str(self.copie.all().count()) + " copie"
        return out

    class Meta:
        verbose_name_plural = "Libri"

class Copia(models.Model):
    data_prestito = models.DateField(default=None, null=True)
    scaduto = models.BooleanField(default=False)
    libro = models.ForeignKey(Libro, on_delete=models.CASCADE, related_name="copie")

    def __str__(self):
        out = "Copia di " + self.libro.titolo + " di " + self.libro.autore + ": "
        if self.scaduto: out += "copia scaduta"
        else: out += "copia non scaduta"
        return out

    class Meta:
        verbose_name_plural = "Copie"
```

Proviamo a migrare

```
python.exe .\manage.py makemigrations gestione
```

```
Migrations for 'gestione':
```

```
gestione\migrations\0002_remove_libro_data_prestito_copia.py
```

- Remove field data_prestito from libro
- Create model Copia

Proviamo a migrare

```
python.exe .\manage.py migrate
```

```
Operations to perform:
```

```
  Apply all migrations: admin, auth, contenttypes, gestione,  
sessions
```

```
Running migrations:
```

```
  Applying gestione.0002_remove_libro_data_prestito_copia... OK
```


Risultato

Start typing to filter...

AUTHENTICATION AND AUTHORIZATION

Groups + Add

Users + Add

GESTIONE

Copie + Add

Libri + Add

Select libro to change

Action: _____ ▼ Go 0 of 5 selected

☐

 LIBRO

☐

 Iliade di Omero ha 0 copie

☐

 La Fattoria degli Animali di George Orwell ha 0 copie

☐

 Odissea di Omero ha 0 copie

☐

 1984 di George Orwell ha 0 copie

☐

 Promessi Sposi di Alessandro Manzoni ha 0 copie

5 Libri

Pare sia andato tutto bene

Ma esistono migrazioni **potenzialmente distruttive**. Ecco come mai ogni migrazione è salvata in

app/migrations/<migration_description>_<migration_number>.py

In questo modo possiamo “tornare indietro nel tempo”

Migrazioni potenzialmente distruttive

Che succede ai dati già presenti se... ?

- Aggiungere un campo senza valori di default o non-nullable
 - Forzerà l'amministratore a risolvere manualmente questi "conflitti"
- Cambio di primary key
 - Rischio potenziale di perdere i dati salvati!

Occorrerebbe fare esportare il proprio DB **prima** di eventuali migrazioni potenzialmente distruttive. Si esegue quindi in seguito la migrazione e si fanno script o comandi ad-hoc per "tradurre" i dati vecchi nella nuova struttura relazionale che abbiamo descritto con l'ultima migrazione

Nel nostro caso

E' **davvero** andato tutto bene?

Si e No.

Non abbiamo perso dati

Il database ha informazioni coerenti, ma il nostro comando iniziale di riempimento della tabella Libro **adesso fallirebbe**:

In initcmds.py possiamo vedere come si cerca di accedere al campo **data_prestito** di Libro, **il quale non esiste più**.

initcmds.py

Deve essere modificato in maniera appropriata.

```
#diamo 8 copie a tutti, di cui una non in prestito.
date = [ func_time(y,m,d) for y in range(2) for m in range(2) for d in range(2) ]

#for k in libridict:
#    print(str(libridict[k]))

for i in range(5):
    l = Libro()
    for k in libridict:
        if k == "autori":
            l.autore = libridict[k][i]
        if k == "titoli":
            l.titolo = libridict[k][i]
        if k == "pagine":
            l.pagine = libridict[k][i]
    l.save()
    for d in date:
        c = Copia()
        c.scaduto = False
        c.libro = l
        c.data_prestito = d
        c.save()
```

Select libro to change

Action: 0 of 5 selected

<input type="checkbox"/>	LIBRO
<input type="checkbox"/>	Iliade di Omero ha 8 copie
<input type="checkbox"/>	La Fattoria degli Animali di George Orwell ha 8 copie
<input type="checkbox"/>	Odissea di Omero ha 8 copie
<input type="checkbox"/>	1984 di George Orwell ha 8 copie
<input type="checkbox"/>	Promessi Sposi di Alessandro Manzoni ha 8 copie

5 Libri

Action: Go 0 of 5 selected

LIBRO

- **Iliade di Omero ha 8 copie**

■ **La Fattoria degli Animali di George Orwell ha 8 copie**

■ **Odissea di Omero ha 8 copie**

■ **1984 di George Orwell ha 8 copie**

■ **Promessi Sposi di Alessandro Manzoni ha 8 copie**

5 Libri

COPIA

■ **Copia di Iliade di Omero: copia non scaduta**

Copia di Iliade di Omero: copia non scaduta

■ **Copia di Iliade di Omero: copia non scaduta**

- **Copia di Iliade di Omero: copia non scaduta**

Copia di Iliade di Omero: copia non scaduta

■ **Copia di Iliade di Omero: copia non scaduta**

■ **Copia di Iliade di Omero: copia non scaduta**

Copia di Iliade di Omero: copia non scaduta

Copia di La Fattoria degli Animali di George Orwell: copia non scaduta

Copia di La Fattoria degli Animali di George Orwell: copia non scaduta

Copia di La Fattoria degli Animali di George Orwell: copia non scaduta

Copia di La Fattoria degli Animali di George Orwell: copia non scaduta

Copia di La Fattoria degli Animali di George Orwell: copia non scaduta

Copia di La Fattoria degli Animali di George Orwell: copia non scaduta

Copia di La Fattoria degli Animali di George Orwell: copia non scaduta

Altre operazioni

Ora abbiamo un database **coerente**. O quasi.

Come faccio a sapere se la copia di un libro è scaduto?

Ho bisogno di un comando che periodicamente mi controlli se la data dell'ultimo prestito è maggiore della durata massimo del prestito di un libro.

Potrei fare un thread periodico che faccia questo controllo, lanciato come operazione di inizializzazione...

initcmds.py

```
def controllo_scadenza():

    MAX_PRESTITO_GIORNI = 15

    print("Controllo copie scadute in corso...")
    for l in Libro.objects.all():
        s0 = l.copie.filter(scaduto=False).exclude(data_prestito=None)
        for c in s0:
            dt = datetime(timezone.now().year, timezone.now().month, timezone.now().day).date()
            if (dt - c.data_prestito) > timedelta(days = MAX_PRESTITO_GIORNI):
                c.scaduto = True
                c.save()
                print(c)

def start_controllo_scadenza(check_time_in_seconds):
    Timer(check_time_in_seconds, controllo_scadenza).start()
```



```
System check identified no issues (0 silenced).
February 16, 2022 - 12:41:19
Django version 4.0.2, using settings 'biblio2.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
Controllo copie scadute in corso...
Copia di Promessi Sposi di Alessandro Manzoni: copia scaduta
Copia di Promessi Sposi di Alessandro Manzoni: copia scaduta
Copia di Promessi Sposi di Alessandro Manzoni: copia scaduta
Copia di Promessi Sposi di Alessandro Manzoni: copia scaduta
Copia di Promessi Sposi di Alessandro Manzoni: copia scaduta
Copia di Promessi Sposi di Alessandro Manzoni: copia scaduta
Copia di 1984 di George Orwell: copia scaduta
Copia di 1984 di George Orwell: copia scaduta
Copia di 1984 di George Orwell: copia scaduta
Copia di 1984 di George Orwell: copia scaduta
Copia di 1984 di George Orwell: copia scaduta
Copia di 1984 di George Orwell: copia scaduta
Copia di Odissea di Omero: copia scaduta
Copia di Odissea di Omero: copia scaduta
Copia di Odissea di Omero: copia scaduta
Copia di Odissea di Omero: copia scaduta
Copia di Odissea di Omero: copia scaduta
Copia di Odissea di Omero: copia scaduta
Copia di La Fattoria degli Animali di George Orwell: copia scaduta
Copia di La Fattoria degli Animali di George Orwell: copia scaduta
Copia di La Fattoria degli Animali di George Orwell: copia scaduta
Copia di La Fattoria degli Animali di George Orwell: copia scaduta
Copia di La Fattoria degli Animali di George Orwell: copia scaduta
Copia di La Fattoria degli Animali di George Orwell: copia scaduta
Copia di Iliade di Omero: copia scaduta
Copia di Iliade di Omero: copia scaduta
Copia di Iliade di Omero: copia scaduta
Copia di Iliade di Omero: copia scaduta
Copia di Iliade di Omero: copia scaduta
Copia di Iliade di Omero: copia scaduta
Controllo copie scadute in corso...
Controllo copie scadute in corso...
Controllo copie scadute in corso...
```

Esercizi

- Creare una pagina web in cui è visualizzata la lista dei libri
 - Vicino a ciascun libro, devono essere presenti i link “prestito” e “restituzione”
- Prestito
 - raggiungibile da gestione/prestito/titolo/autore/
 - La copia deve essere disponibile
 - Se il prestito è andato a buon fine il sistema mostra la pk della copia prestata
- Restituzione
 - gestione/restituzione/titolo/autore/ qui si chiede di inserire l'id della copia, tramite blocco html select. Redireziona in gestione/restituzione/pk/
 - gestione/restituzione/pk/ qui si finalizza la restituzione, ammonendo l'utente nel caso abbia restituito il libro in ritardo

Lista di Libri

Promessi Sposi di Alessandro Manzoni

Prestito

Restituzione

1984 di George Orwell

Prestito

Restituzione

Odissea di Omero

Prestito

Restituzione

La Fattoria degli Animali di George Orwell

Prestito

Restituzione

Iliade di Omero

Prestito

Restituzione

Scegli copia da restituire: 1984 George Orwell

129: prestato il Feb. 17, 2022

Seleziona

130: prestato il Feb. 15, 2022

Seleziona

131: prestato il Jan. 16, 2022

Seleziona

132: prestato il Jan. 15, 2022

Seleziona

133: prestato il Feb. 16, 2021

Seleziona

134: prestato il Feb. 15, 2021

Seleziona

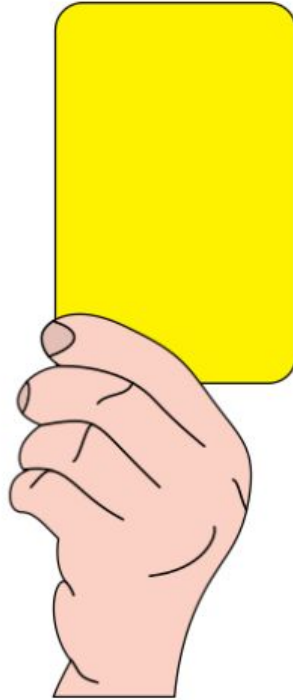
135: prestato il Jan. 16, 2021

Seleziona

136: prestato il Jan. 15, 2021

Seleziona

Conferma Restituzione



La prossima volta magari riportalo prima...
