Python i Django. Programowanie aplikacji webowych

Jeff Forcier, Paul Bissex, Wesley Chun

Drogi Czytelniku! Poniżej zamieszczona jest errata do książki:
"Python i Django. Programowanie aplikacji webowych"

Jest to lista błędów znalezionych po opublikowaniu książki, zgłoszonych i zaakceptowanych przez naszą redakcję. Pragniemy, aby nasze publikacje były wiarygodne i spełniały Twoje oczekiwania. Zapoznaj się z poniższą listą. Jeśli masz dodatkowe zastrzeżenia, możesz je zgłosić pod adresem https://helion.pl/user/erraty

Strona	Linia	Jest	Powinno
43	28	#11	#12
43	28	#11	#12
44	9	12. Podobnie jak w punkcie 10.	12. Podobnie jak w punkcie 11.
49	30	>>> book = {´title´: Python i Django. Programowanie aplikacji webowych, ´year´: 2009}	>>> book = {´title´:´Python i Django. Programowanie aplikacji webowych´, ´year´: 2009}

120	7	class Book(models.Model):
		 title = models.CharField(max_length=100)
		 author = models.ForeignKey("Author")
		
		class Author(models.Model):
		 name = models.CharField(max_length=100)
		
			class Author(models.Model):
			 name = models.CharField(max_length=100)
			
			class Book(models.Model):
			 title = models.CharField(max_length=100)
			 author = models.ForeignKey("Author")

138	17	SQL prawie bez zmian (lub prawie bez zmian;	SQL bez zmian (lub prawie bez zmian;
138	30	nazwa aplikacji_nazwamodelu	nazwaaplikacji_nazwamodelu
149	5	url(r´^blog/´, include(´myproject.blog.urls´),	url(r´^blog/´, include(´myproject.blog.urls´)),
169	9	<title>Section 2</title>	<title>Sekcja 1</title>
186	26, 38	@permalink	@models.permalink
226	27	story_list = Story.objects.filter(Q(title__contains=term) | (markdown_content__contains=term))	tak jest w przykładach z ftp i działa: story_list = Story.objects.filter(Q(title__contains=term | Q(markdown_content__contains=term))

Poniżej znajduje się lista błędów znalezionych przez czytelników, ale jeszcze nie potwierdzonych przez Redakcję:

Strona	Linia	Jest	Powinno
44	**6 od dołu**	**i dla łańcuchów Inicode**	**i u dla łańcuchów Inicode**