

Table 2: Defining Attributes for each Class/Table

Class/ Table	Attributes	Datatype Specifications	Code in apps/models.py files	Comments
Book	name	String (max 120 characters)	name= models.CharField (max_length=120)	Book name
	genre	String based on choice	genre_choices= (('classic', 'Classic'), ('romantic', 'Romantic'), ('comic', 'Comic'), ('fantasy', 'Fantasy'), ('horror', 'Horror'), ('educational', 'Educational')) genre = models.CharField (max_length=12, choices=genre_ch oices, default='classic')	Romance, classic, comic, fantasy, horror, other. The list is shown as a dropdown to the user. The default value is specified as “classic”
	book_type	String based on choice	book_type_choices=(('hardcover', 'Ha rd cover'), ('ebook', 'E-Book'), ('audiob', 'Audiobook')) book_type = models.CharField (max_length=12, choices=book_typ e_choices, default='hardcop y')	Hardcover, eBook, Audiobook The list is shown as a dropdown to the user. The default value is specified as “Hardcover”.

	price		price=models.FloatField(help_text='in US dollars \$')	`help_text` allows to add a tooltip, which you will see below the form field in the admin panel
	author_name	String (max 120 characters)	author_name=models.CharField(max_length=120)	Author's name
	pic	Will be added later when you learn to create views and define templates.		
Salesperson	username	Unique username for the salesperson	username = models.OneToOneField(User, on_delete=models.CASCADE)	<p>Each salesperson will have a username.</p> <p>OneToOne specifies that each username can only be connected to one salesperson</p> <p>Need to import:</p> <pre>`from django.contrib.auth.models import User`</pre> <p>You need to specify what to do when the salesperson is deleted: `CASCADE` specifies that whenever the user with username is deleted, the complete profile of the salesperson will be also deleted.</p>
	name	String (max 120 characters)	name=models.CharField(max_length=120)	Salesperson's name
	bio	Long string of text	bio=models.TextField(default="no bio...")	Text box with default statement: "no bio"
	pic	Will be added later when you learn to create views and define templates.		
Customer	name	String (max 120 characters)	name=models.CharField(max_length=120)	Customer's name

	pic	Will be added later when you learn to create views and define templates.		
	notes	Long text string	<code>notes = models.TextField(default="no notes...")</code>	Any notes the customer wants to make
Sale	book	A book identifier	<code>book=models.ForeignKey(Book, on_delete=models.CASCADE)</code>	<p>book is the foreign key referring to the Book class—you'll need to import it too.</p> <p>`CASCADE` implies that when a book is deleted all the positions related to the book will be deleted</p>
	quantity	A number (amount)	<code>quantity=models.PositiveIntegerField()</code>	Number of books sold (of the same type within a transaction)
	price	a real number	<code>price = models.FloatField()</code>	<p>Total sale price. For example, if a book's price is \$10, and three are sold, the price will be $10 \times 3 = \\$30$.</p> <p>You can also write your own function to automatically calculate the price, but for the sake of simplicity, we'll enter it manually in the Bookstore application.</p>
	date_create	Date Attribute	<code>models.DateTimeField(auto_now_add=True)</code>	<p>Date of sale - will be automatically set to the current date.</p> <p>You won't be able to see this attribute in the panel, as it will be filled in automatically.</p>