(DJANGO~1) C:\Users\jinal\Desktop folders\Desktop\DojoAssignments\Python\Django\_ORM\user\_orm>python manage.py shell

Python 2.7.13 (v2.7.13:a06454b1afa1, Dec 17 2016, 20:42:59) [MSC v.1500 32 bit (Intel)] on win32

Type "help", "copyright", "credits" or "license" for more information.

(InteractiveConsole)

>>> from apps.book\_authors.models import \*

>>> Book.objects.create(name="C Sharp", desc="C sharp book released in 2016")

<Book: Book object>

>>> Book.objects.create(name="Java", desc="Java book released in 2015")

<Book: Book object>

>>> Book.objects.create(name="Python", desc="Python book released in 2015")

<Book: Book object>

>>> Book.objects.create(name="PHP", desc="Python book released in 2015")

<Book: Book object>

>>> Book.objects.create(name="Ruby", desc="Ruby book released in 2015")

<Book: Book object>

>>> this\_book = Book.objects.get(id=2)

>>> author = Author.objects.create(first\_name="Mike", last\_name="Welsh", email="mikew@gmail.com")

>>> author = Author.objects.create(first\_name="Speros", last\_name="Taylor", email="speros@mail.com")

>>> author = Author.objects.create(first\_name="John", last\_name="Doe", email="john@mail.com")

>>> author = Author.objects.create(first\_name="Jadee", last\_name="diodato", email="jadee@gmail.com")

>>> author = Author.objects.create(first\_name="Jay", last\_name="Patel", email="jayp@gmail.com")

(DJANGO~1) C:\Users\jinal\Desktop folders\Desktop\DojoAssignments\Python\Django\_ORM\user\_orm>python manage.py makemigrations

Migrations for 'book\_authors':

apps\book\_authors\migrations\0002\_author\_notes.py

- Add field notes to author

(DJANGO~1) C:\Users\jinal\Desktop folders\Desktop\DojoAssignments\Python\Django\_ORM\user\_orm>python manage.py migrate

Operations to perform:

Apply all migrations: admin, auth, book\_authors, contenttypes, dojo\_ninjas, sessions, user\_login

Running migrations:

Applying book\_authors.0002\_author\_notes... OK

(DJANGO~1) C:\Users\jinal\Desktop folders\Desktop\DojoAssignments\Python\Django\_ORM\user\_orm>

(DJANGO~1) C:\Users\jinal\Desktop folders\Desktop\DojoAssignments\Python\Django\_ORM\user\_orm>python manage.py shell

Python 2.7.13 (v2.7.13:a06454b1afa1, Dec 17 2016, 20:42:59) [MSC v.1500 32 bit (Intel)] on win32

Type "help", "copyright", "credits" or "license" for more information.

(InteractiveConsole)

>>> from apps.book\_authors.models import \*

## Change the name of the 5th book to C#

>>> book = Book.objects.last()

>>> book.name = "C#"

>>> book.save()

## Change the first\_name of the 5th author to Ketul

>>> author = Author.objects.last()

>>> author.first\_name = "Ketul"

>>> author.save()

## Assign the first author to the first 2 books

>>> author = Author.objects.first()

>>> book1 = Book.objects.get(id=1)

>>> book2 = Book.objects.get(id=2)

>>> author.books.add(book1)

>>> author.books.add(book2)

## Assign the second author to the first 3 books

>>> author2 = Author.objects.get(id=2)

>>> book1 = Book.objects.get(id=1)

>>> book2 = Book.objects.get(id=2)

>>> book3 = Book.objects.get(id=3)

>>> author2.books.add(book1)

>>> author2.books.add(book2)

>>> author2.books.add(book3)

## Assign the third author to the first 4 books

>>> book4 = Book.objects.get(id=4)

>>> author3 = Author.objects.get(id=3)

>>> author3.books.add(book1)

>>> author3.books.add(book2)

>>> author3.books.add(book3)

>>> author3.books.add(book4)

## Assign the fourth author to the first 5 books (or in other words, all the books)

>>> author4 = Author.objects.get(id=4)

>>> book5 = Book.objects.last()

>>> author4.books.add(book1)

>>> author4.books.add(book2)

>>> author4.books.add(book3)

>>> author4.books.add(book4)

>>> author4.books.add(book5)

## For the 3rd book, retrieve all the authors

>>> book3.authors.all()

<QuerySet [<Author: Author object>, <Author: Author object>, <Author: Author object>]>

>>> book3.authors.first()

<Author: Author object>

## For the 3rd book, remove the first author

>>> book3.books.remove(author2)

## For the 2nd book, add the 5th author as one of the authors

>>> book2.authors.add(Author.objects.last())

## Find all the books that the 3rd author is part of

>>> Book.objects.filter(authors\_\_id\_\_contains=3)

<QuerySet [<Book: Book object>, <Book: Book object>, <Book: Book object>, <Book: Book object>]>