**Django Admin - TabularInline and StackedInline**

1 - models.py

from django.db import models

class Author(models.Model):

id = models.BigAutoField(primary\_key=True)

name = models.CharField(max\_length=100)

def \_\_str\_\_(self):

return self.name

class Meta:

db\_table = 'author'

class Book(models.Model):

id = models.BigAutoField(primary\_key=True)

author = models.ForeignKey(Author, on\_delete=models.CASCADE)

title = models.CharField(max\_length=100)

def \_\_str\_\_(self):

return self.title

class Meta:

db\_table = 'book'

class Student(models.Model):

id = models.BigAutoField(primary\_key=True)

name = models.CharField(max\_length=255)

def \_\_str\_\_(self):

return self.name

class Meta:

db\_table = 'student'

class MarkList(models.Model):

id = models.BigAutoField(primary\_key=True)

student = models.ForeignKey(Student, on\_delete=models.SET\_NULL, null=True)

subject\_name = models.CharField(max\_length=255)

mark = models.IntegerField()

def \_\_str\_\_(self):

return self.student.name

class Meta:

db\_table = 'mark\_list'

2 - admin.py

from django.contrib import admin

from backend.models import Author, Book, Student, MarkList

# Register your models here.

class BookInline(admin.TabularInline):

model = Book

class AuthorAdmin(admin.ModelAdmin):

inlines = [

BookInline,

]

admin.site.register(Book)

admin.site.register(Author, AuthorAdmin)

class MarkListInline(admin.StackedInline):

model = MarkList

class StudentAdmin(admin.ModelAdmin):

list\_display = ['id', 'name']

inlines = [

MarkListInline,

]

admin.site.register(Student, StudentAdmin)

admin.site.register(MarkList)