**CRM MODELS**

1. **User Profile:**

from django.contrib.auth.models import User

from django.db import models

class UserProfile(models.Model):

user = models.OneToOneField(User, on\_delete=models.CASCADE)

# Add additional fields like phone number, role, etc.

phone\_number = models.CharField(max\_length=15, blank=True, null=True)

role = models.CharField(max\_length=50, choices=[("Admin", "Admin"), ("Teacher", "Teacher"), ("Student", "Student")])

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

return self.user.username

1. **Contact Model:**

class Contact(models.Model):

first\_name = models.CharField(max\_length=100)

last\_name = models.CharField(max\_length=100)

email = models.EmailField(unique=True)

phone\_number = models.CharField(max\_length=15, blank=True, null=True)

# Other contact information fields

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

return f"{self.first\_name} {self.last\_name}"

1. **Lead/Opportunity Model:**

class Lead(models.Model):

contact = models.ForeignKey(Contact, on\_delete=models.CASCADE)

status = models.CharField(max\_length=50, choices=[("New", "New"), ("Contacted", "Contacted"), ("Converted", "Converted")])

source = models.CharField(max\_length=50)

# Other fields like date\_created, notes, etc.

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

return f"{self.contact} - {self.status}"

1. **Deal/Transaction Model:**

class Deal(models.Model):

contact = models.ForeignKey(Contact, on\_delete=models.CASCADE)

amount = models.DecimalField(max\_digits=10, decimal\_places=2)

close\_date = models.DateField()

# Other fields like product sold, payment status, etc.

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

return f"{self.contact} - ${self.amount}"

1. **Task Model:**

class Task(models.Model):

contact = models.ForeignKey(Contact, on\_delete=models.CASCADE)

title = models.CharField(max\_length=200)

due\_date = models.DateField()

status = models.CharField(max\_length=50, choices=[("Not Started", "Not Started"), ("In Progress", "In Progress"), ("Completed", "Completed")])

# Other fields like description, assigned\_to, etc.

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

return self.title