DJANGO MODELS

GitHub Link: https://github.com/asimthaha/tax-calculator-django.git

Submitted To:

Satheesh Kumar Assistant Professor

Submitted By:

Asim Thaha Azeez 22PMC118

from django.db import models # Create your models here.

class UserDetails(models.Model):

    user\_id = models.AutoField(db\_column="User\_id", primary\_key=True)

    name = models.CharField(db\_column="Name", max\_length=40)

    email = models.CharField(db\_column="Email", max\_length=254)

    password = models.CharField(db\_column="Password", max\_length=20)

    zipcode = models.IntegerField(null=True, db\_column="Zipcode")

    def \_\_str\_\_(self):

        return self.name

class TaxDetails(models.Model):

    REGIME\_STATUS = (

        ("New Regime", "New Regime"),

        ("Old Regime", "Old Regime"),

    )

    AGE\_GROUP = (

        ("For All Age Groups", "For All Age Groups"),

        ("Below 60", "Below 60"),

        ("60 to 80", "60 to 80"),

        ("Above 80", "Above 80"),

    )

    CATEGORY = (

        ("Employee/Pensioner", "Employee/Pensioner"),

        ("Other","Other")

    )

    FINANCIAL\_YEAR =(

        ("2023-2022", "2023-2022"),

        ("2022-2021", "2022-2021"),

        ("2021-2020", "2021-2020"),

    )

    name = models.ForeignKey(UserDetails, on\_delete=models.CASCADE, default=True)

    financial\_year = models.CharField(max\_length=20 ,blank=True,null=True,

        choices=FINANCIAL\_YEAR,

        default="s",

        help\_text="Age Group of People")

    age\_group = models.CharField(max\_length=5,blank=True,null=True,

        choices=AGE\_GROUP,

        default="s",

        help\_text="Age Group of People")

    category\_emp\_or\_pen = models.CharField(max\_length=30,blank=True,null=True,

        choices=CATEGORY,

        default="Employee/Pensioner/",

        help\_text="Category of People",)

    regime = models.CharField(max\_length=20,blank=True,null=True,choices=REGIME\_STATUS,

        default=True,

        help\_text="regime status new or old")

    salary\_income = models.IntegerField(null=True, blank=True)

    other\_income = models.IntegerField(null=True, blank=True)

    standard\_deduction = models.IntegerField(null=True, blank=True)

    professional\_tax = models.IntegerField(null=True, blank=True)

    house\_rent\_exemption = models.IntegerField(null=True, blank=True)

    home\_loan = models.IntegerField(null=True, blank=True)

    deductions\_u\_80c = models.IntegerField(null=True, blank=True)

    nps\_u\_80c = models.IntegerField(null=True, blank=True)

    taxable\_income = models.IntegerField(null=True, blank=True)

    tax\_on\_taxable\_income = models.IntegerField(null=True, blank=True)

    rebate\_u\_87a = models.IntegerField(null=True, blank=True)

    surcharge\_on\_tax = models.IntegerField(null=True, blank=True)

    education\_cess = models.IntegerField(null=True, blank=True)

    total\_tax = models.IntegerField(null=True, blank=True)

    def \_\_str\_\_(self):

        return self.regime

    regime = models.CharField(

        max\_length=20,

        choices=REGIME\_STATUS,

        blank=True,

        default=True,

        help\_text="regime status new or old",

    )

    age\_group = models.CharField(

        max\_length=20,

        choices=AGE\_GROUP,

        blank=True,

        default="s",

        help\_text="Age Group of People",

    )

class UserFeedback(models.Model):

    name = models.ForeignKey(UserDetails, on\_delete=models.CASCADE, default=False)

    email = models.EmailField(max\_length=254)

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

    def \_\_str\_\_(self):

        return self.description

class CarouselImages(models.Model):

    photo = models.ImageField(upload\_to="photo", blank=True, null=True)

    description = models.CharField(max\_length=300)

    def \_\_str\_\_(self):

        return self.description

class TaxSavingsGuide(models.Model):

    photo = models.ImageField(upload\_to="photo", blank=True, null=True)

    card\_title = models.CharField(max\_length=50)

    card\_description = models.CharField(max\_length=200)

    def \_\_str\_\_(self):

        return self.card\_title

class TaxSlabRates(models.Model):

    REGIME\_STATUS = (

        ("New Regime", "New Regime"),

        ("Old Regime", "Old Regime"),

    )

    AGE\_GROUP = (

        ("For All Age Groups", "For All Age Groups"),

        ("Below 60", "Below 60"),

        ("60 to 80", "60 to 80"),

        ("Above 80", "Above 80"),

    )

    RATES\_STATUS = (

        ("0%", "Nil"),

        ("5%", "5%"),

        ("10%", "10%"),

        ("15%", "15%"),

        ("20%", "20%"),

        ("30%", "30%"),

    )

    regime = models.CharField(

        max\_length=20,

        choices=REGIME\_STATUS,

        blank=True,

        default=True,

        help\_text="regime status new or old",

    )

    age\_group = models.CharField(

        max\_length=20,

        choices=AGE\_GROUP,

        blank=True,

        default="s",

        help\_text="Age Group of People",

    )

    rates = models.CharField(

        max\_length=20,

        choices=RATES\_STATUS,

        blank=True,

        default="0",

        help\_text="Tax Slab Rates",

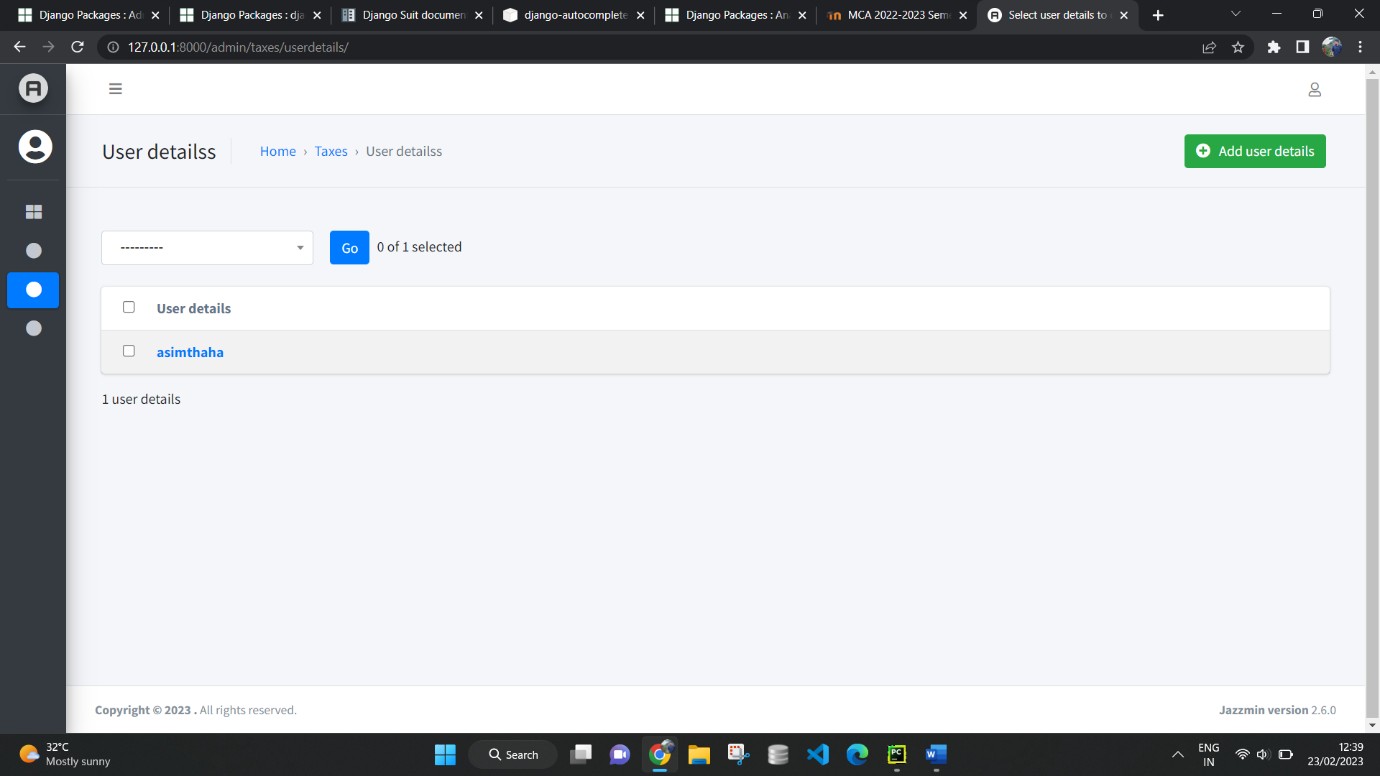
    )

    slabs = models.CharField(max\_length=30, help\_text="Tax Slabs")

    def \_\_str\_\_(self):

        """String for representing the MyModelName object (in Admin site etc.)."""

        return self.slabs

**ADMIN INTERFACE SCREENSHOTS**

