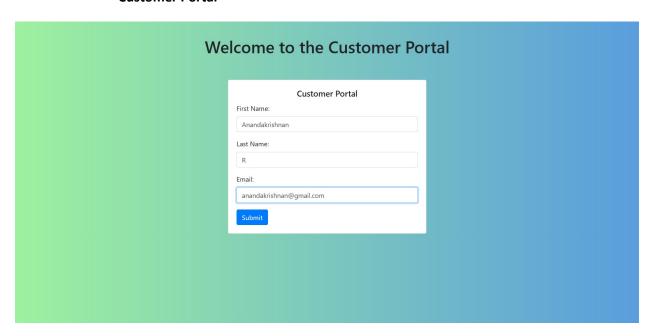
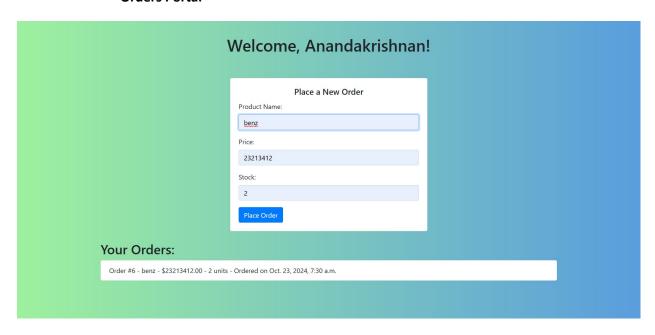
Model migartions and signals task

*Customer Portal



*Orders Portal



*models.py

```
from django.db import models
# Create your models here.
class Customer(models.Model):
                                                        # Customer model
    first name = models.CharField(max length=100)
    last_name = models.CharField(max_length=100)
    email = models.EmailField(unique=True)
class Product(models.Model):
    name = models.CharField(max_length=100)
    price = models.DecimalField(max_digits=10, decimal_places=2)
    stock = models.IntegerField()
                                                        # Order model
class Order(models.Model):
    customer = models.ForeignKey(Customer, on_delete=models.CASCADE)
    product = models.ForeignKey(Product, on_delete=models.CASCADE)
    order_date = models.DateTimeField(auto_now_add=True)
class CustomerProfile(models.Model):
    customer = models.OneToOneField(Customer, on_delete=models.CASCADE)
```

*signals.py

```
from django.db.models.signals import post_save
from django.dispatch import receiver
from .models import Customer, CustomerProfile

# Signal to automatically create a CustomerProfile when a new Customer is created
@receiver(post_save, sender=Customer)
def create_customer_profile(sender, instance, created, **kwargs):
    if created:
        CustomerProfile.objects.create(customer=instance)
```

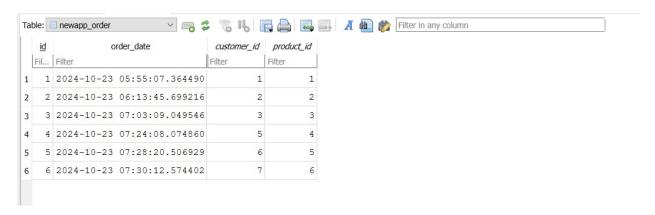
*views.py

```
from django.shortcuts import render
from django.shortcuts import render, redirect
from .models import Customer, Product, Order
def customer_portal(request):
    if request.method == 'POST':
        first_name = request.POST.get('first_name')
        last_name = request.POST.get('last_name')
        email = request.POST.get('email')
        if first_name and last_name and email:
            customer = Customer.objects.create(first_name=first_name, last_name=last name,
email=email)
           return redirect('order_page', customer_id=customer.id)
    return render(request, 'customer_portal.html')
def order_page(request, customer_id):
place orders
    customer = Customer.objects.get(id=customer_id)
    if request.method == 'POST':
        product_name = request.POST.get('product_name')
        product_price = request.POST.get('price')
        product stock = request.POST.get('stock')
        product = Product.objects.create(
           name=product_name,
            price=product_price,
            stock=product stock
        Order.objects.create(
            customer=customer,
            product=product
        return redirect('order_page', customer_id=customer.id) # Refresh the page to show updated
orders
    orders = Order.objects.filter(customer=customer)
                                                              # Retrieve all orders for this
    return render(request, 'order_page.html', {
        'customer': customer,
        'orders': orders
```

*Customer Table



*Orders Table



*Product Table



*Customer Profile Table

