

1.####ذخیره‌سازی سبد خرید در پایگاه داده به جای سشن

ذخیره‌سازی سبد خرید در پایگاه داده به جای سشن می‌تواند به دلایل زیر کارآمدتر باشد:

- ** -پایداری و تداوم داده‌ها:** در این روش، داده‌های سبد خرید با بستن مرورگر یا خروج از سایت حذف نمی‌شوند.
- ** -پشتیبانی از چند دستگاه:** کاربران می‌توانند از طریق دستگاه‌های مختلف به سبد خرید دسترسی داشته باشند.
- ** -تسهیل در تجزیه و تحلیل داده‌ها:** ذخیره‌سازی در پایگاه داده به شما امکان تجزیه و تحلیل بهتر رفتار کاربران را می‌دهد.

2.####مدیریت سبد خرید برای کاربران احراز هویت‌شده و ناشناس

در این رویکرد، دو نوع کاربر در نظر گرفته می‌شود: **کاربران احراز هویت‌شده** و **کاربران ناشناس**.

2.1.#####کاربران احراز هویت‌شده

برای کاربران احراز هویت‌شده که دارای حساب کاربری هستند، سبد خرید به حساب آن‌ها مرتبط می‌شود و داده‌های آن به صورت پایدار در پایگاه داده ذخیره می‌شود. به این ترتیب، با هر بار ورود کاربر به حساب، می‌توان سبد خرید او را از پایگاه داده بارگذاری کرد.

****پیاده‌سازی مدل برای کاربران احراز هویت‌شده در جنگو**:**

در جنگو، شما می‌توانید یک مدل `Cart` ایجاد کنید که به مدل `User` متصل است. هر کاربر تنها یک سبد خرید خواهد داشت و کالاهای سبد خرید به عنوان `CartItem` ذخیره می‌شوند.

```python

```

from django.contrib.auth.models import User

from django.db import models

class Cart(models.Model):

 user = models.OneToOneField(User, on_delete=models.CASCADE, related_name="cart")
 items = models.ManyToManyField('CartItem')
 created_at = models.DateTimeField(auto_now_add=True)
 updated_at = models.DateTimeField(auto_now=True)

class CartItem(models.Model):

 product = models.ForeignKey(Product, on_delete=models.CASCADE)
 quantity = models.PositiveIntegerField(default=1)
 ...

```

در این مدل:

`Cart` -شامل یک رابطه یک-به-یک با مدل `User` است. به این ترتیب هر کاربر احراز هویت شده یک سبد خرید منحصر به فرد دارد.

`CartItem` -برای ذخیره سازی کالاهای موجود در سبد خرید استفاده می شود.

## 2.2.#### کاربران ناشناس

برای کاربران ناشناس که وارد حساب کاربری نشده اند، می توان سبد خرید را به طور موقت در **\*\*سشن\*\*** ذخیره کرد. در این حالت، اطلاعات سبد خرید به شکل یک ساختار JSON (مانند یک دیکشنری) در سشن ذخیره می شوند. زمانی که کاربر وارد حساب کاربری خود شود، این داده ها به پایگاه داده منتقل شده و به سبد خرید مرتبط با حساب کاربر اضافه می شوند.

**\*\*مثال از سبد خرید در سشن برای کاربران ناشناس\*\***:

```

python
views.py

from django.shortcuts import render, get_object_or_404
from .models import Product

def add_to_cart(request, product_id):
 cart = request.session.get('cart', {})
 product = get_object_or_404(Product, id=product_id)
 if str(product_id) in cart:
 cart[str(product_id)]['quantity'] += 1
 else:
 cart[str(product_id)] = {'name': product.name, 'quantity': 1, 'price': product.price}
 request.session['cart'] = cart
 return render(request, 'cart.html', {'cart': cart})

```

3.#### انتقال سبد خرید از سشن به پایگاه داده هنگام ورود کاربر

زمانی که یک کاربر ناشناس وارد حساب خود می‌شود، داده‌های سبد خرید او که در سشن ذخیره شده است به پایگاه داده منتقل می‌شود. برای این کار، داده‌ها از سشن استخراج می‌شوند و به `Cart` مربوط به کاربر اضافه می‌شوند.

**\*\*کد برای انتقال سبد خرید از سشن به پایگاه داده\*\*:**

```

python

from .models import Cart, CartItem

def transfer_cart_session_to_db(user, session_cart):
 cart, created = Cart.objects.get_or_create(user=user)

```

```

for item_id, item_data in session_cart.items():
 product = Product.objects.get(id=item_id)
 CartItem.objects.update_or_create(
 cart=cart,
 product=product,
 defaults={'quantity': item_data['quantity']}
)
'''

```

4.#### مدیریت تعداد کالاهای سبد خرید

هنگام اضافه کردن کالاهای جدید به سبد خرید:

-اگر کالا از قبل در سبد موجود باشد، \*\*تعداد آن به روزرسانی\*\* می شود.

-اگر کالا در سبد موجود نباشد، \*\*به عنوان یک آیتم جدید اضافه\*\* می شود.

این رویه معمولاً با اورراید کردن متدهای اضافه کردن یا به روزرسانی آیتم های سبد خرید پیاده سازی می شود.

5.#### نمونه موردی: دیجی کالا

در فروشگاه آنلاین دیجی کالا، مدیریت سبد خرید به این صورت است:

\*\* -کاربران احراز هویت شده: \*\* اطلاعات سبد خرید به حساب کاربری آنها متصل شده و در پایگاه داده نگهداری می شود. این اطلاعات در دستگاه های مختلف همگام سازی می شود.

\*\* -کاربران ناشناس: \*\* سبد خرید در سشن ذخیره شده و در صورت ورود به حساب، به پایگاه داده منتقل می شود.

6.#### استفاده از داکر برای مقیاس پذیری و نگهداری داده ها

در پروژه‌ای مبتنی بر داکر، می‌توانید از **\*\*کانتینرهای پایگاه داده\*\*** (مثلاً PostgreSQL یا MySQL و **\*\*کش\*\***)  
(مثل Redis) استفاده کنید. برای مدیریت سبد خرید:

**\*\*پایگاه داده\*\***: سبد خرید کاربران احراز هویت شده به طور پایدار در یک کانتینر پایگاه داده ذخیره می‌شود.

**\*\*سشن‌ها\*\***: می‌توانید از Redis به عنوان کش برای مدیریت سشن‌های کاربران ناشناس استفاده کنید. این باعث بهبود سرعت و کارایی می‌شود.

به این ترتیب، سیستم شما برای مدیریت کاربران و سبد خریدها با استفاده از داکر و جنگو مقیاس پذیرتر خواهد بود.