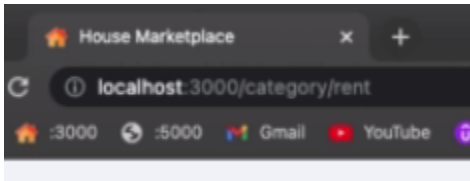


2. Fetch Listings From Firebase

Now that we have our category page , now we want to fetch the listings from firebase and show them in page

We need useEffect bcz here we want to do our try catch async operation .

We need useParams bcz we need to get from the current link that what are we looking for either sell or rent !



From Firestore we need :

Collection , getDocs , query , where , OrderBy , limit , startAfter : from firestore

Single doc ko get krna alag ha or puri collection ko get krna bilkul alag cheez ha !

startAfter : Paining k liye :

Also bring our db

```
22  useEffect(() => {
23    const fetchListings = async () => {
24      try {
25        // Get reference
26        const listingsRef = collection(db, 'listings')
27
28        // Create a query
29        const q = query(
30          listingsRef,
31          where('type', '==', params.categoryName),
32          orderBy('timestamp', 'desc'),
33          limit(10)
34        )
```

Now execute the query :

```
// Execute query
const querySnap = await getDocs(q)
```

Phr hamen loop chalana hoga :

```
let listings = []

querySnap.forEach((doc) => {
  console.log(doc.data())
})
```

```
let listings = []

querySnap.forEach((doc) => {
  return listings.push({
    id: doc.id,
    data: doc.data()
  })
})
```

Agar ham id ko doc k under dkehn to nahi milti ye firebase ka default behavior ha
Ab listing ko apni state me daal do :

```
setListings(listings)
```

SO:

```

useEffect(() => {
  const fetchListings = async () => {
    try {
      //get a reference
      const listingsRef = collection(db, 'listings');

      //create a query
      const q = query(listingsRef, where('type', '=', params.categoryName),
        orderBy('timestamp'), limit(10));
      //execute the query

      const querySnap = await getDocs(q);
      const listings = [];

      querySnap.forEach((doc) => {
        return listings.push({
          id: doc.id,
          data: doc.data(),
        });
      });

      setListings(listings);
      setLoading(false);
    }
  }
});

```

Hamen ek puri collection me se data chaye tha :

Uske liye:

Pehle collection ka ref banaya

Phr ham ne query create ki

Or phr getDocs me pass krke results ko querySnap me save krlia

QuerySnap me or bhi both sa stuff hota ha direct data nahi hota jase API me hota

Us results (querySnap) me se data lene k liye ham ne uspe loop chalay or listings me wo object push krdia with properties that we want , yaad rahy id **doc** me nahoi hoti ye firebase aesy hi deta ha !

Doc.data() k zarye us object me wo data le lia

Ab listings me ek object push ho chuka with 2 things ek id or ek data or phr ham ne usko apni local state me save kr dia !

```

return (
  <div className='category'>
    <header>
      <p className='pageHeader'>{params.categoryName === 'rent' ? 'Places for Rent'
      : 'Places for Sale'}</p>
    </header>

    {loading ? <Spinner /> : listings && listings.length > 0 ? <></> :
    <p>No Houses for {params.categoryName}</p>}
  </div>
);
};

export default Category;

```

Very self-explanatory!

Now we will work on inside the fragments:

```

<main>
  <ul className='categoryListings'>
    {listings.map(() => (
      <h3>{listings.data.name}</h3>
    ))}
  </ul>
</main>
</>

```

Is map se ham ek new compt ban akr return kren gy listingItem but for now we do this!

So the complete code is :

```

Category.jsx U x
src > pages > Category.jsx > Category > useEffect() callback > fetchListings
1  import { useState, useEffect } from 'react';
2  import { useParams } from 'react-router-dom';
3  import { collection, getDocs, query, where, orderBy, limit, startAfter } from 'firebase';
4  import { db } from '../firebase.config';
5  import { toast } from 'react-toastify';
6  import Spinner from '../components/Spinner';

```

```

8  const Category = () => {
9      const [listings, setListings] = useState(null);
10     const [loading, setLoading] = useState(true);
11
12     const params = useParams();
13
14     useEffect(() => {
15         const fetchListings = async () => {
16             try {
17                 //get a reference
18                 const listingsRef = collection(db, 'listings');
19
20                 //create a query
21                 const q = query(listingsRef,
22                     where('type', '=', params.categoryName),
23                     orderBy('timestamp'), limit(10));
24                 //execute the query
25
26                 const querySnap = await getDocs(q);
27                 const listings = [];
28
29                 querySnap.forEach((doc) => {
30                     return listings.push({
31                         id: doc.id,
32                         data: doc.data(),
33                     });
34                 });
35
36                 setListings(listings);
37                 setLoading(false);
38             } catch (error) {
39                 toast.error('Could not fetch listings ');
40             }
41         };
42         fetchListings();
43     }, [params.categoryName]);

```

JSX :

```
<div className='category'>
  <header>
    <p className='pageHeader'>{params.categoryName === 'rent' ? 'Places for Rent'
    : 'Places for Sale'}</p>
  </header>

  {loading ? (
    <Spinner />
  ) : listings && listings.length > 0 ? (
    <
      <main>
        <ul className='categoryListings'>
          {listings.map(() => (
            <h3>{listings.data.name}</h3>
          ))}
        </ul>
      </main>
    </>
  ) : (
    <p>No Houses for {params.categoryName}</p>
  )}
</div>
```
