

```
'use client';

import { QueryClient, QueryClientProvider } from '@tanstack/react-query';

import { useState } from 'react';

import { trpc, trpcConfig } from './trpc';

import { useToast } from '@shared/components/ui/Toasts/use-toast';

import { getURL } from '../helpers';
```

```
export const TrpcProvider: React.FC<{ children: React.ReactNode }> = ({
  children,
}) => {
  const toaster = useToast();

  const [queryClient] = useState(
    () =>
      new QueryClient({
        defaultOptions: {
          queries: { staleTime: 5000 },
          mutations: {
            onError(error, variables, context) {
              toaster.toast({
                title: error.message,
                variant: 'destructive',
              });
            },
          },
        },
      })
  ),
}
```

```
);
```

```
const [trpcClient] = useState(() => trpc.createClient(trpcConfig));
```

```
return (
```

```
  <trpc.Provider client={trpcClient} queryClient={queryClient}>
```

```
    <QueryClientProvider client={queryClient}>
```

```
      {children}
```

```
      { /* <ReactQueryDevtools /> */ }
```

```
    </QueryClientProvider>
```

```
  </trpc.Provider>
```

```
);
```

```
};
```