```
import { createServerClient, type CookieOptions } from '@supabase/ssr';
import { type NextRequest, NextResponse } from 'next/server';

export const createClient = (request: NextRequest) => {
  // Create an unmodified response
  let response = NextResponse.next({
    request: {
      headers: request.headers,
    },
  });

  const supabase = createServerClient(
    process.env.NEXT_PUBLIC_SUPABASE_URL!,
    process.env.NEXT_PUBLIC_SUPABASE_ANON_KEY!,
    {
      cookies: {
        get(name: string) {
          return request.cookies.get(name)?.value;
        },
        set(name: string, value: string, options: CookieOptions) {
          // If the cookie is updated, update the cookies for the request and response
          request.cookies.set({
            name,
            value,
            ...options,
          });
  response = NextResponse.next({
    request: {
      headers: request.headers,
    },
  });
  response.cookies.set({
    name,
    value,
    ...options,
  });
},
remove(name: string, options: CookieOptions) {
  // If the cookie is removed, update the cookies for the request and response
  request.cookies.set({
    name,
    value: '',
    ...options,
  });
  response = NextResponse.next({
    request: {
      headers: request.headers,
    },
  });
  response.cookies.set({
    name,
    value: '',
            ...options,
          });
        },
      },
    },
  );

  return { supabase, response };
};

export const updateSession = async (request: NextRequest) => {
  try {
    // This `try/catch` block is only here for the interactive tutorial.
    // Feel free to remove once you have Supabase connected.
    const { supabase, response } = createClient(request);

    // This will refresh session if expired - required for Server Components
    // https://supabase.com/docs/guides/auth/server-side/nextjs
    await supabase.auth.getUser();

    return response;
  } catch (e) {
    // If you are here, a Supabase client could not be created!
    // This is likely because you have not set up environment variables.
    // Check out http://localhost:3000 for Next Steps.
    return NextResponse.next({
      request: {
        headers: request.headers,
      },
    });
  }
};
```