# The http Protocol

## Monitoring http Headers 1

Et billede, der indeholder skærmbillede

Automatisk genereret beskrivelse

Request URL: IP-adressen efterfulgt af porten, projektet og dernæst stien til html filen

Request Method: Her ses GET metoden.

Status code: Dette er kode 304 (Not Modified), hvilket fortæller, at responset ikke er ændret i forhold til tidligere cached version af responset. Der er er derfor ikke ændret noget.

Det er også ved status code hvor der differentieres fra index.html, hvor status kode 200 vises, hvilket er OK (request er succesfuldt).

Remote address: Dette viser blot porten.

Referrer Policy: Dette er standard opførsel hvis der ikke er nogen specifik politik til stede.

## Monitoring http headers 2

Et billede, der indeholder skærmbillede

Automatisk genereret beskrivelse

Request URL: Dette viser stien til selve billedet der er blevet lagt op.   
Request Method: GET

Status Code: 200, det var succesfuldt.

Remote address: viser den IP-adresse og port, som hoster billedet.

Referrer policy: Ingen valgt (default).

Længere nede ses Response Headers

Her kan tidspunktet for responset ses, hvilket content type det er (jpeg) og hvor lang tid cachen bliver gemt og herefter skal fornyes.

Et billede, der indeholder skærmbillede

Automatisk genereret beskrivelse

Dette er infoen for myStyle.css, hvilket ikke er oprettet i projektet, kun i index.html headeren.

Her ses tydeligt at statuskoden er 404, hvilket betyder, at URL’et ikke er genkendt.

## Monitoring http headers 3

Et billede, der indeholder skærmbillede

Automatisk genereret beskrivelse

Her ses statuskoden 304, som viser, at responset ikke er blevet modificeret, så den benytter sig af samme cache fra det originale respons.

Et billede, der indeholder skærmbillede

Automatisk genereret beskrivelse

Dette er ved URL’et set ovenover. Her fås statuskode 302, hvilket fortæller, at ’requestet ressource’ er midlertidigt ændret. Det er en redirect status respons kode, som fortæller at resourcen er blevet rykket til et nyt URL.

## 3a) Redirecting to HTTPs instead of HTTP

Et billede, der indeholder skærmbillede

Automatisk genereret beskrivelse

Dette er første respons der kommer, hvilket i URL viser at http bliver benyttet. Til gengæld når man kigger på statuskoden, bliver 301 Moved Permanently vist. Som fortæller, at resourcen der spørges efter er blevet rykket permanent.   
Remote address viser IP-adressen efterfulgt af port 80 (http), som benytter sig af TCP protokollen.

Et billede, der indeholder skærmbillede

Automatisk genereret beskrivelse

Dette er den næste respons, som tydeligt viser https i URL’et samt en ny statuskode, herunder 304. Den viser at responset ikke er blevet ændret, og er en implicit redirection til en cached ressource.

Noget nyt er også, at port nummeret har ændret sig til 443, hvilket er porten for https frem for den tidligere http(port 80).

## 4a) Status codes(5xx)

Et billede, der indeholder skærmbillede

Automatisk genereret beskrivelse

Her fås statuskode 500 Internal Server Error, hvilket betyder, at der er noget, en condition, som gør, at requestet ikke kunne blive fulført. Ved indlæsning af siden ses tydeligt under beskeden: / by zero. Man kan ikke dividere med nul.

## 4b) Status codes(4xx)

Et billede, der indeholder skærmbillede

Automatisk genereret beskrivelse

Her fås statuskode 404, som viser at URL ikke genkendes. Det går hånd i hånd med, at i\_dont\_exist ikke eksisterer.

## 4c) Status codes – ranges

2xx statuskoder omhandler succesfulde responses, blandt andet ses dette i opgave 2 ved jpg filen.

3xx statuskoder er koder for redirects, hvor specielt statuskode 304 er oplevet i opgaverne.

4xx statuskoder er klient fejl. Kode 404 er blevet observeret flere gange når URL’et ikke blev genkendt.

5xx statuskoder viser server fejl, f.eks. fik jeg kode 500 da der blev forsøgt at dividere med nul.

De specifikke statuskoder er blevet kommenteret i ovenstående opgaver.

## Get HTTP request headers on the server

Et billede, der indeholder skærmbillede

Automatisk genereret beskrivelse

Request method er sat til GET i html filen i projektet. URL’et bliver forlænget med ?fname=x&lname=x (x værende inputtet i felterne). GET metoden benytter sig dermed af inputtene som parametre i URL’en.

Et billede, der indeholder skærmbillede

Automatisk genereret beskrivelse

Her ses Request Method som værende POST, ligesom formen er sat til. URL’et forlænges ikke med attributerne fra input felterne på html siden. POST metoden gør ikke som GET og bruger input som parametre i URL’en men som form data i stedet. Værdierne er skjulte.

## Sessions

Ved dette projekt ses en ny oversigt under Network.

Et billede, der indeholder skærmbillede

Automatisk genereret beskrivelse

Cookies er blevet tilføjet, herunder en request cookie med navnet JSESSIONID, som gemmer navnet der blev submittet fra siden.

Dette er en session cookie, den bliver gemmes midlertidigt og dermed kun mens browseren er åben. Hvis man derimod lukker for browseren, vil denne session cookie blive slettet.

HTTP er stateless, hvilket betyder, at når transaktionen er overstået så slutter forbindelsen mellem browseren og serveren. Dette er muliggjort ved, at http benytter sig af http cookies der tracker user state, antallet af forbindelser, den sidste forbindelse og mere. Den her en persistent forbindelse og flere requests kan sendes gennem samme TCP forbindelse.

## Persistent cookies

Et billede, der indeholder skærmbillede

Automatisk genereret beskrivelse

I modsætning til session cookies bliver persistent cookies ikke gemt i browseren. Det er en tekst fil der gemmes lokalt på computeren som oftest har en udløbsdato på et til to år. De ”lever” dermed længere end session cookies.   
I billedet ovenfor ses en persistent cookie, der efter gentagne forsøg når browseren genstartes stadig holder værdien Karsten.