



### • وب سرور ساده

- در این تمرین شما می بایستی یک وب سرور ساده ایی طراحی کنید که قابلیت پردازش یک درخواست را در لحظه داشته باشد. جزئیات پیاده سازی به شرح زیر می باشد:
- وب سرور هنگامی که به یک کلاینت متصل شد، می بایست یک سوکت ارتباطی با کلاینت بسازد.
  - از طریق این اتصال TCP، وب سرور درخواست های HTTP دریافت کند.
  - در خواست های HTTP از طریق مرورگر کلاینت به وب سرور داده می شوند.
  - بتواند درخواست فرستاده شده را پردازش کند تا مشخص کند که چه فایل توسط کلاینت درخواست شده است.
  - بتواند فایل درخواست شده را از فایل سیستم وب سرور بگیرد.
  - وب سرور باید بتواند فایل خواسته شده را در قالب یک پاسخ HTTP (HTTP response) برای کلاینت درخواست کننده با header های مناسب بسازد.
  - در نهایت نیز وب سرور باید پاسخ HTTP مورد نظر را از طریق اتصال TCP ایی که با کلاینت (مرورگر کلاینت) برقرار کرده است ارسال کند.

### • ارسال کننده ایمیل

- در این تمرین شما می بایست یک mail client با استفاده از پروتکل SMTP پیاده سازی کنید که بتواند به آدرس ایمیل های دلخواهی ایمیل ارسال کند. جزئیات پیاده سازی به شرح زیر می باشد:
- Mail client شما باید یک اتصال TCP با یک mail server مشخص برقرار کند.
  - با استفاده از پروتکل SMTP با mail server ارتباط برقرار کند. (شکل ۱)

- پیام شما را به شناسه ایمیل دلخواه ارسال کند.
- در نهایت نیز اتصال TCP را با mail server قطع کند.

## Sample SMTP interaction

```

S: 220 hamburger.edu
C: HELO crepes.fr
S: 250 Hello crepes.fr, pleased to meet you
C: MAIL FROM: <alice@crepes.fr>
S: 250 alice@crepes.fr... Sender ok
C: RCPT TO: <bob@hamburger.edu>
S: 250 bob@hamburger.edu ... Recipient ok
C: DATA
S: 354 Enter mail, end with "." on a line by itself
C: Do you like ketchup?
C: How about pickles?
C: .
S: 250 Message accepted for delivery
C: QUIT
S: 221 hamburger.edu closing connection

```

شکل ۱ - نمونه ایی از کامند های SMTP برای ارتباط با mail server و ارسال ایمیل

**توجه:** در حین پیاده سازی نیز به موارد زیر دقت کنید.

## Electronic Mail: SMTP [RFC 2821]

- uses TCP to reliably transfer email message from client to server, port 25
- direct transfer: sending server to receiving server
- three phases of transfer
  - handshaking (greeting)
  - transfer of messages
  - closure
- command/response interaction (like HTTP)
  - **commands:** ASCII text
  - **response:** status code and phrase
- messages must be in 7-bit ASCII

## نکات تمرین:

- ✓ برای پیاده سازی ۲ تمرین می توانید از زبان ++C, java و python استفاده کنید.
  - ✓ برای ۲ تمرین شما فقط حق استفاده از سوکت های پروتکل TCP را دارید و از کتابخانه ها و کلاس های آماده مختلف زبان های برنامه نویسی نمیتوانید استفاده کنید.
  - ✓ به فرمت پیام های پروتکل های SMTP و HTTP دقت کنید که در پیاده سازی دچار مشکل نشوید.
  - ✓ گزارشی برای ۲ تمرین شامل توضیحاتی در رابطه با نحوه پیاده سازی به همراه تصاویر مناسب از خروجی تهیه بفرمایید.
  - ✓ نام اعضای گروه در فایل گزارش قرار گرفته شده باشد.
  - ✓ کد ها و گزارش خود را در قالب فایل zip و با فرمت [CA#][Group#] در درس افزار آپلود کنید.
- موفق و پیروز باشید.