

Lezione 3

Soluzione Esercizio

```
public int readShortRequest(InputStream in, byte[] buffer) {
    try
    {
        int offset = 0;
        int res;
        int len = in.read();
        if (len < 0) {
            return len;
        }

        while ((res = in.read(buffer, offset, Math.min(len,buffer.length) -
offset)) > 0) {
            offset += res;
        }
        if (len == offset) {
            return len;
        }
        else {
            return -1;
        }
    }
    catch (IOException ioe) {
        return -1;
    }
}

public int readLongRequest(InputStream in, byte[] buffer) {
    try
    {
        int offset = 0;
        int res =in.read();
        if (res<0)
            return -1;
        int len = res * 256;
        res = in.read();
        if(res<0)
            return -1;
        len+=res;
        while ((res = in.read(buffer, offset, Math.min(len ,buffer.length)-
offset)) > 0) {
            offset += res;
        }
        if (len == offset) {
            return len;
        }
        else {
            return -1;
        }
    }
    catch (IOException ioe) {
        return -1;
    }
}
```

```
}  
}
```

1) Il protocollo Http

HyperText Transfer Protocol è il protocollo usato per trasferire ipertesti.

Si possono trovare tutti i dettagli su RFC: 1945, 2617

numero porta standard: 80.

nome porta: www.

CLIENT (BROWSER)	SERVER
apre connessione	
GET risorsa HTTP/1.0	
eventuali altre linee dello header	
(linea vuota)	
	HTTP/1.1 200 OK
	eventuali altre linee dello header
	(linea vuota)
	contenuto risorsa
	chiude connessione

1.1) Tipi di richieste del Client

Le richieste del client possono essere: GET, POST, PUT, DELETE e HEAD di cui solo GET, POST e HEAD sono attualmente usate.

1.1.1) GET

Il metodo GET serve per recuperare una qualsiasi risorsa. Se nello header viene specificato il campo If-Modified-Since la richiesta diventa condizionata al fatto che la risorsa viene trasferita solo se modificata dopo la data specificate nel campo If-Modified-Since questo per ridurre il numero di trasferimenti.

1.1.2) HEAD

Il significato del metodo HEAD è simile a quello del metodo GET e si differenzia per il fatto che il metodo HEAD ritorna solo lo header che sarebbe stato ritornato come risposta al metodo GET, ma senza il contenuto della risorsa. Questo metodo può essere utilizzato per ottenere informazioni sulla risorsa senza bisogno di trasferire la risorsa.

1.1.3) POST

Il metodo POST è usato per trasferire risorse dal Client al Server.

1.2) Risposte del server

Le risposte del server possono essere:

1.2.1) 200 OK

La richiesta ha avuto successo. L'informazione restituita con la risposta dipende dal metodo usato per la richiesta nel seguente modo:

- GET: una entità che corrisponde alla risorsa richiesta viene inviata nella risposta;
- HEAD: la risposta contiene solo informazione nello header e nessun corpo;
- POST: una entità che descrive e/o contiene il risultato dell'azione.

1.2.2) 201 Created

La richiesta è stata soddisfatta e ha comportato la creazione di una nuova risorsa che può essere recuperata tramite il link specificato nella risposta. Solo il metodo POST può causare una tale risposta.

1.2.3) 202 Accepted

La richiesta è stata accettata, ma non è stata ancora completata.

1.2.4) 204 No Content

Il server ha soddisfatto la richiesta e non c'è nessuna informazione di risposta.

3.2.5) 300 Multiple Choices

La risorsa richiesta è disponibile in una o più locazioni.

1.2.6) 301 Moved Permanently

Alla risorsa richiesta è stato assegnato un nuovo indirizzo.

1.2.7) 302 Moved Temporarily

La risorsa richiesta è stata spostata temporaneamente ad un nuovo indirizzo.

1.2.8) 304 Not Modified

La risorsa richiesta non è stata modificata. Risposta ottenuta da una richiesta GET con campo If-Modified-Since field attivo.

1.2.9) 400 Bad Request

La richiesta non è sintatticamente corretta.

1.2.10) 401 Unauthorized

La risorsa richiesta richiede l'autenticazione dell'utente. La risposta include il campo WWW-Authenticate. Il cliente può rifare la richiesta con un campo Authorization adatto.

1.2.11) 403 Forbidden

La risorsa non è disponibile per essere pubblicata.

1.2.12) 404 Not Found

Il server non ha trovato la risorsa.

1.2.13) 500 Internal Server Error

Il server ha trovato un errore inaspettato che non gli ha permesso di portare a termine la richiesta.

1.2.14) 501 Not Implemented

Il server non supporta la funzionalità richiesta.

1.2.15) 502 Bad Gateway

Il server mentre si comportava da gateway o proxy ha ricevuto una risposta non valida.

1.2.16) 503 Service Unavailable

Il server è temporaneamente non in grado di gestire la richiesta.