P1：

1. False. Cause it request a web page ,so its using HTTP, and then it send 4 and receive 4.
2. True. The persistent type of http will have connection open.
3. False. Case each TCP have only one request.
4. False. Generation time will be request by date.
5. False. When client not found, 404 error will send and will have no content

P3:

TCP and UDP in transport layer

DNS and HTTP in application layer

P6:

1. Persistent connections are discussed in section 8 of RFC 2616. Sections 8.1.2 and 8.1.2.1 of the RFC indicate that either the client or the server can indicate to the other that it is going to close the persistent connection. It does so by including the connection-token "close" in the Connection-header field of the http request/reply.
2. No encryption services are provided by HTTP.
3. Clients that use persistent connections should limit the number of simultaneous connections that they maintain to a given server. A single-user client SHOULD NOT maintain more than 2 connections with any server or proxy.
4. Yes. A client might have started to send a new request at the same time that the server has decided to close the "idle" connection. From the server's point of view, the connection is being closed while it was idle, but from the client's point of view, a request is in progress.

P10:

Cause non-persistent connection, it’s 150 bit/sec bandwidth for 10 connection which equal 15 bits/sec

Total time = (200/150+t + 200/150 +t+ 200/150+t + 100,000/150+ t) + (200/ (150/10) +t + 200/ (150/10) +t + 200/ (150/10) +t + 100,000/ (150/10) + t) = 7377 + 8\*t (seconds)

Then consider persistent HTTP connection. The total time needed is given by: (200/150+Tp + 200/150 +t + 200/150+Tp + 100,000/150+ t) + 10\*(200/150+Tp + 100,000/150+ t) =7351 + 24\*t (seconds)

Cause so t =10/ (3\*10^8) = 3\*10^-6, and t< delay

So persistent HTTP don’t have significant gain when compared to non-persistent case.