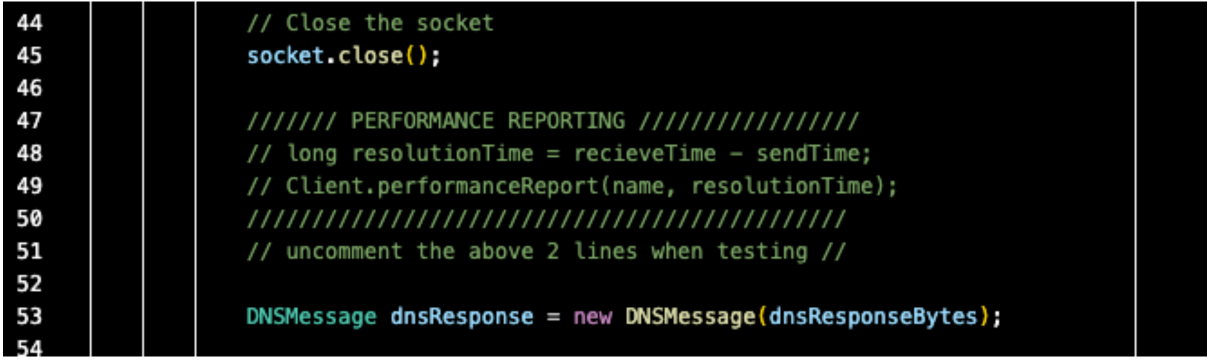
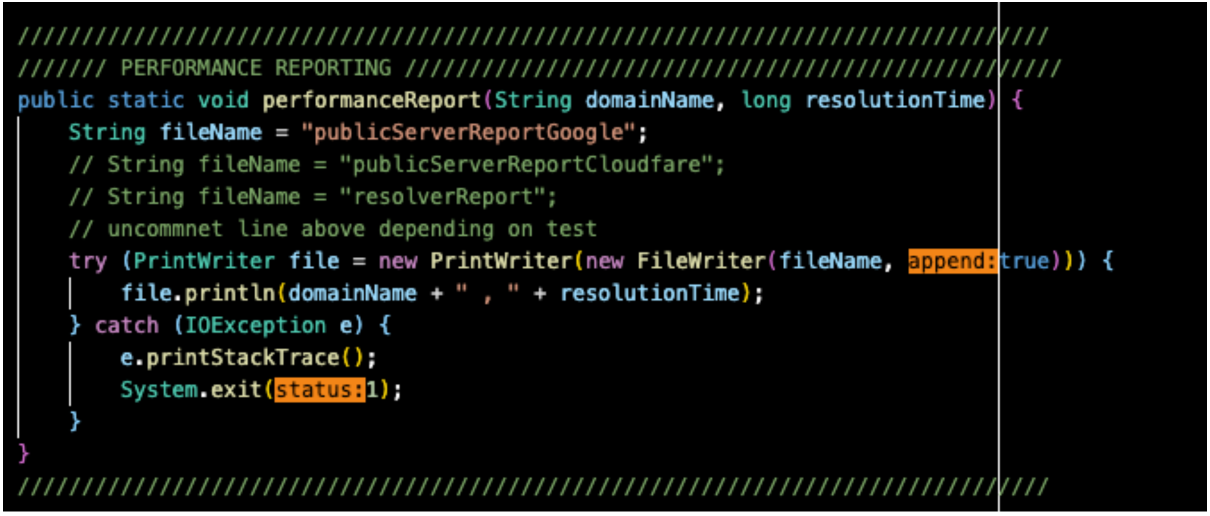
Performance Reporting

1. Testing against public server:
2. First go to Client and then function constructDNSQuery and make sure the recursionDesired variable is set as true.
3. Comment out all the System.exit(1) in the Client. (limitation)
4. Uncomment the lines as specified below in Client.java



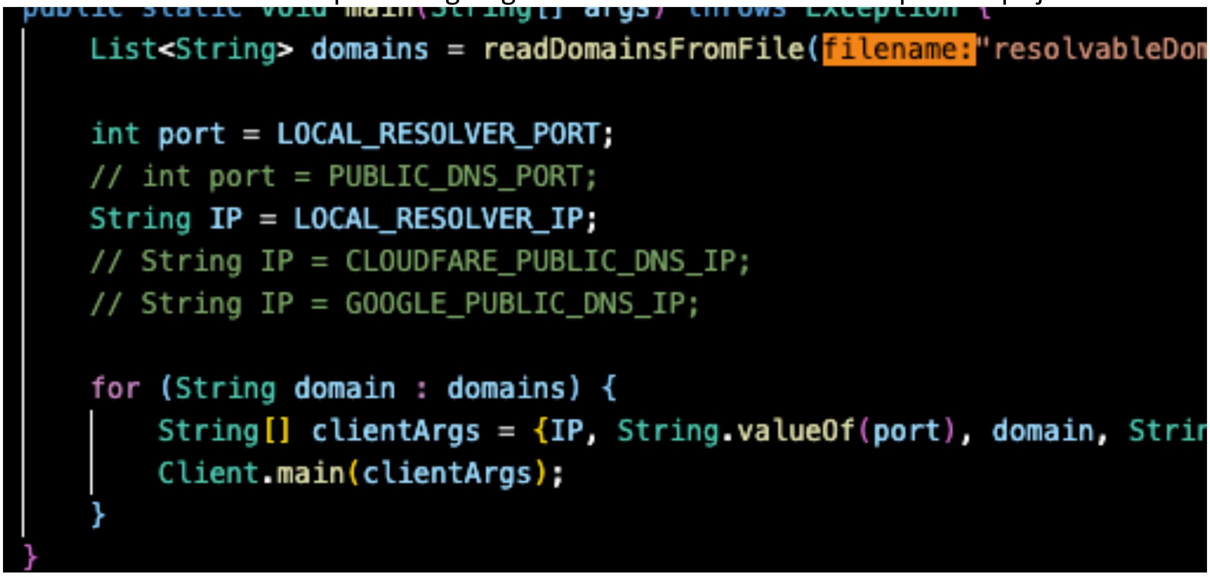
```java
        // Close the socket
        socket.close();

        /////// PERFORMANCE REPORTING ////////////////
        // long resolutionTime = recieveTime - sendTime;
        // Client.performanceReport(name, resolutionTime);
        ////////////////////////////////////////////
        // uncomment the above 2 lines when testing //

        DNSMessage dnsResponse = new DNSMessage(dnsResponseBytes);
```

5. Make sure correct filename is used in Client.java



```java
//////////////////////////////////////////////////////////////////////////////
/////// PERFORMANCE REPORTING /////////////////////////////////////////////////
public static void performanceReport(String domainName, long resolutionTime) {
    String fileName = "publicServerReportGoogle";
    // String fileName = "publicServerReportCloudfare";
    // String fileName = "resolverReport";
    // uncommnet line above depending on test
    try (PrintWriter file = new PrintWriter(new FileWriter(fileName, append:true))) {
        file.println(domainName + " , " + resolutionTime);
    } catch (IOException e) {
        e.printStackTrace();
        System.exit(status:1);
    }
}
//////////////////////////////////////////////////////////////////////////////
```

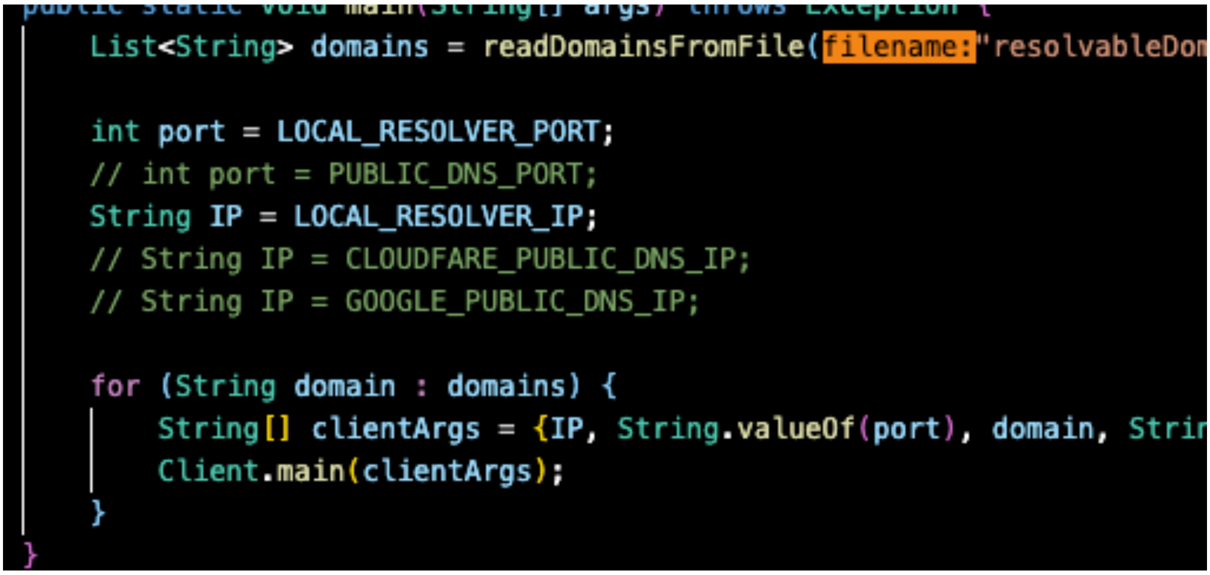
6. Make sure correct IP and port are going to be used in PerformanceReportScript.java



```java
public static void main(String[] args) throws Exception {
    List<String> domains = readDomainsFromFile(filename:"resolvableDom

    int port = LOCAL_RESOLVER_PORT;
    // int port = PUBLIC_DNS_PORT;
    String IP = LOCAL_RESOLVER_IP;
    // String IP = CLOUDFARE_PUBLIC_DNS_IP;
    // String IP = GOOGLE_PUBLIC_DNS_IP;

    for (String domain : domains) {
        String[] clientArgs = {IP, String.valueOf(port), domain, Strin
        Client.main(clientArgs);
    }
}
```

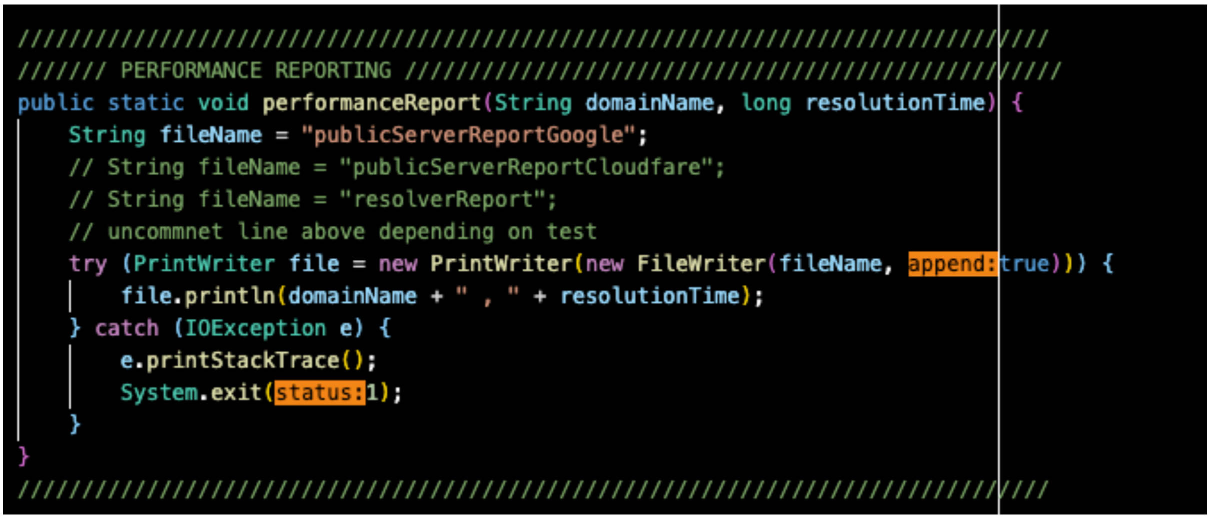
7. Run PerformanceReportScript.java

8. Get the results in tabular form which will be in a publicServerReport.txt file.
9. Repeat steps 5-8 for the other public Server. Making sure the other public server is used.



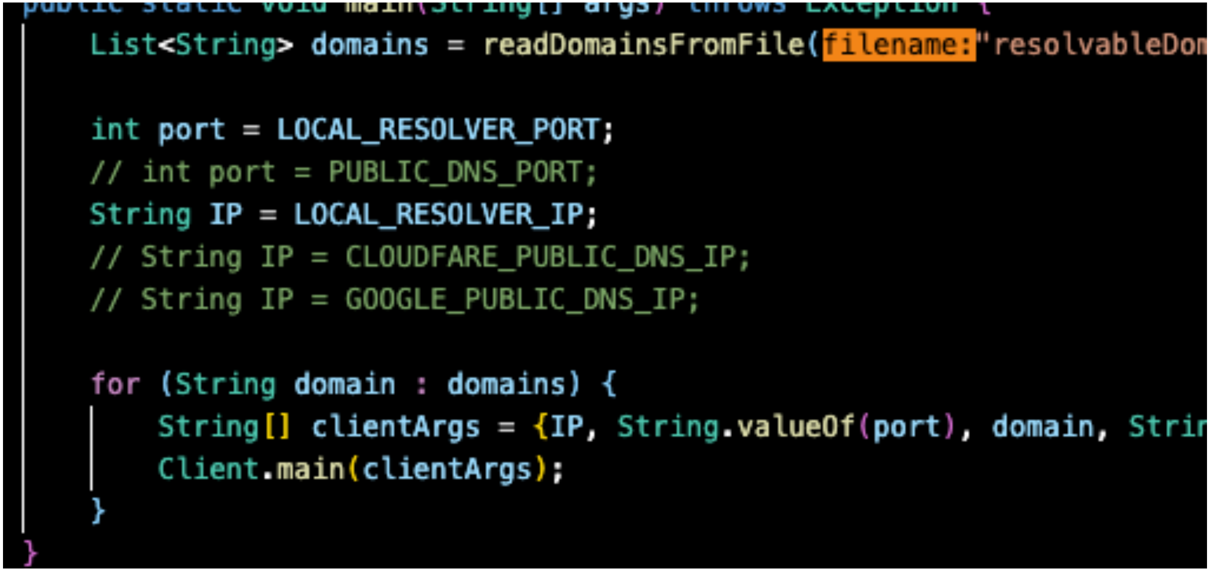
```
public static void main(String[] args) throws Exception {
    List<String> domains = readDomainsFromFile(filename:"resolvableDom

    int port = LOCAL_RESOLVER_PORT;
    // int port = PUBLIC_DNS_PORT;
    String IP = LOCAL_RESOLVER_IP;
    // String IP = CLOUDFARE_PUBLIC_DNS_IP;
    // String IP = GOOGLE_PUBLIC_DNS_IP;

    for (String domain : domains) {
        String[] clientArgs = {IP, String.valueOf(port), domain, Strin
        Client.main(clientArgs);
    }
}
```

10. Comment out all the System.exit(1) in the Resolver. (limitation).
11. Go to Client and then function constructDNSQuery and make sure the recursionDesired variable is set as false.
12. Make sure correct filename is used.



```
////////////////////////////////////////////////////////////////////////////////
/////// PERFORMANCE REPORTING //////////////////////////////////////////////////////
public static void performanceReport(String domainName, long resolutionTime) {
    String fileName = "publicServerReportGoogle";
    // String fileName = "publicServerReportCloudfare";
    // String fileName = "resolverReport";
    // uncommnet line above depending on test
    try (PrintWriter file = new PrintWriter(new FileWriter(fileName, append:true))) {
        file.println(domainName + " , " + resolutionTime);
    } catch (IOException e) {
        e.printStackTrace();
        System.exit(status:1);
    }
}
////////////////////////////////////////////////////////////////////////////////
```

13. Make sure the correct port and ip are being used in PerformanceReportScript.java



```
public static void main(String[] args) throws Exception {
    List<String> domains = readDomainsFromFile(filename:"resolvableDom

    int port = LOCAL_RESOLVER_PORT;
    // int port = PUBLIC_DNS_PORT;
    String IP = LOCAL_RESOLVER_IP;
    // String IP = CLOUDFARE_PUBLIC_DNS_IP;
    // String IP = GOOGLE_PUBLIC_DNS_IP;

    for (String domain : domains) {
        String[] clientArgs = {IP, String.valueOf(port), domain, Strin
        Client.main(clientArgs);
    }
}
```

14. Run the resolver by going “java Resolver 5300 5” in a terminal.
15. Run PerformanceReportScript.java
16. Get the results in tabular form in localResolverReport.txt.
17. Copy results into google sheets and plot for all.
18. Diagrams are in the tar file as pdfs.