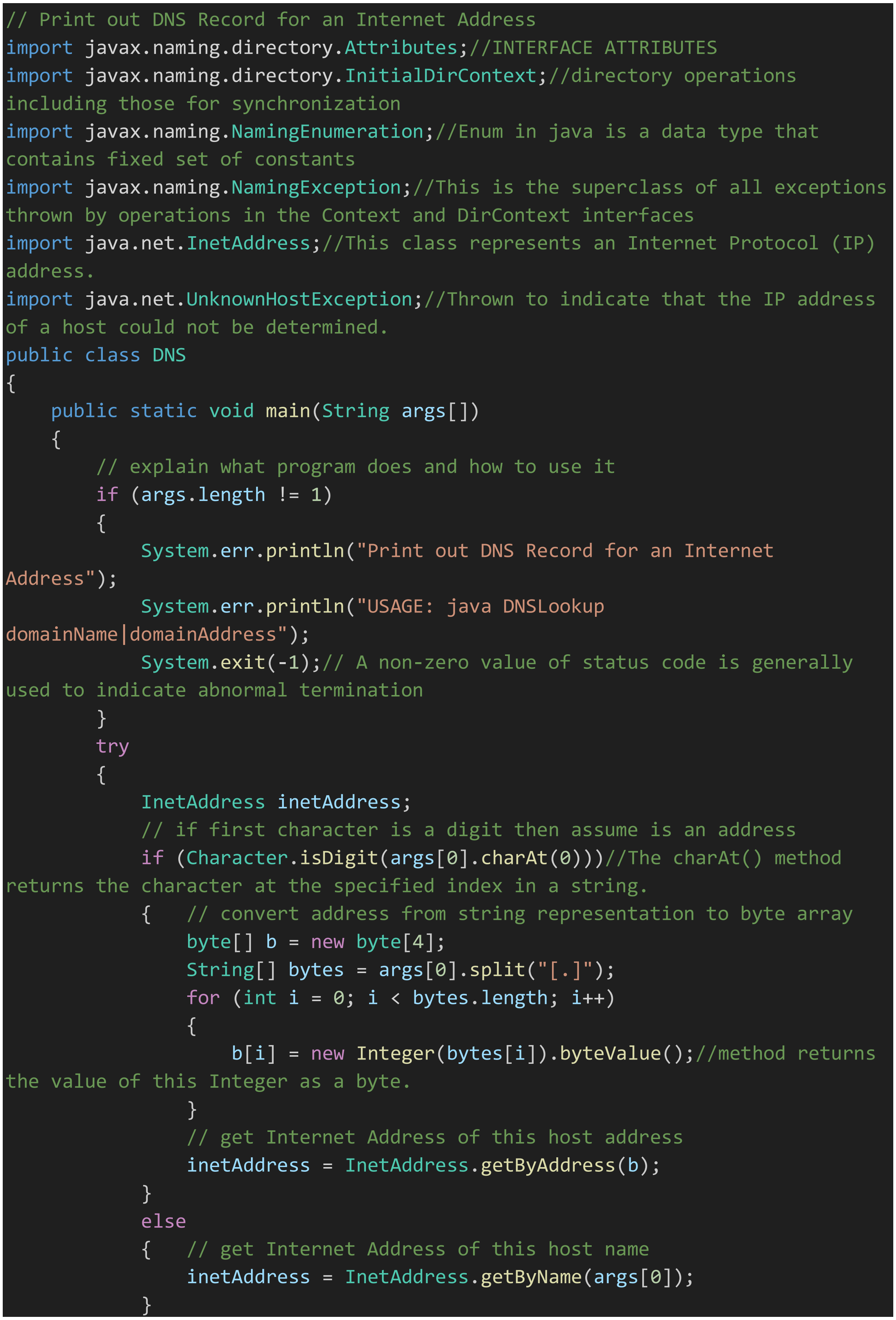
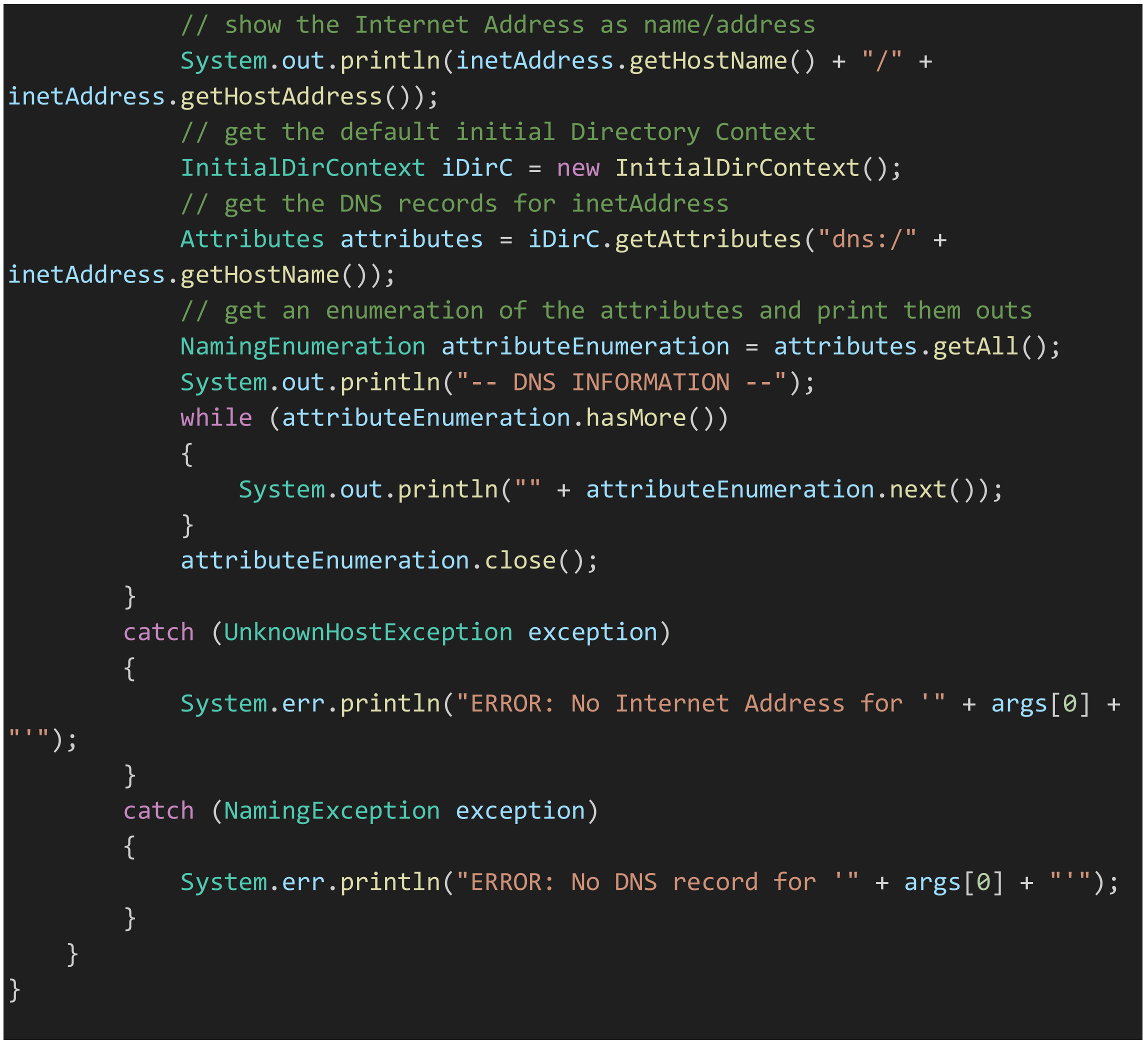
Experiment no.11



```java
// Print out DNS Record for an Internet Address
import javax.naming.directory.Attributes;//INTERFACE ATTRIBUTES
import javax.naming.directory.InitialDirContext;//directory operations
including those for synchronization
import javax.naming.NamingEnumeration;//Enum in java is a data type that
contains fixed set of constants
import javax.naming.NamingException;//This is the superclass of all exceptions
thrown by operations in the Context and DirContext interfaces
import java.net.InetAddress;//This class represents an Internet Protocol (IP)
address.
import java.net.UnknownHostException;//Thrown to indicate that the IP address
of a host could not be determined.
public class DNS
{
    public static void main(String args[])
    {
        // explain what program does and how to use it
        if (args.length != 1)
        {
            System.err.println("Print out DNS Record for an Internet
Address");
            System.err.println("USAGE: java DNSLookup
domainName|domainAddress");
            System.exit(-1);// A non-zero value of status code is generally
used to indicate abnormal termination
        }
        try
        {
            InetAddress inetAddress;
            // if first character is a digit then assume is an address
            if (Character.isDigit(args[0].charAt(0)))//The charAt() method
returns the character at the specified index in a string.
            {   // convert address from string representation to byte array
                byte[] b = new byte[4];
                String[] bytes = args[0].split("[.]");
                for (int i = 0; i < bytes.length; i++)
                {
                    b[i] = new Integer(bytes[i]).byteValue();//method returns
the value of this Integer as a byte.
                }
                // get Internet Address of this host address
                inetAddress = InetAddress.getByAddress(b);
            }
            else
            {   // get Internet Address of this host name
                inetAddress = InetAddress.getByName(args[0]);
            }
```



```
            // show the Internet Address as name/address
            System.out.println(inetAddress.getHostName() + "/" +
inetAddress.getHostAddress());
            // get the default initial Directory Context
            InitialDirContext iDirC = new InitialDirContext();
            // get the DNS records for inetAddress
            Attributes attributes = iDirC.getAttributes("dns:/" +
inetAddress.getHostName());
            // get an enumeration of the attributes and print them outs
            NamingEnumeration attributeEnumeration = attributes.getAll();
            System.out.println("-- DNS INFORMATION --");
            while (attributeEnumeration.hasMore())
            {
                System.out.println("" + attributeEnumeration.next());
            }
            attributeEnumeration.close();
        }
        catch (UnknownHostException exception)
        {
            System.err.println("ERROR: No Internet Address for '" + args[0] +
"'");
        }
        catch (NamingException exception)
        {
            System.err.println("ERROR: No DNS record for '" + args[0] + "'");
        }
    }
}
```

Output :-

```
google.com/142.250.72.238

-- DNS INFORMATION --

A: 142.250.72.238

AAAA: 2607:f8b0:4007:80e::200e

MX: 10 smtp.google.com.

NS: ns1.google.com.

NS: ns2.google.com.

TXT: "v=spf1 include:_spf.google.com ~all"
```