

Lab 2b

Objectives:

- Familiarizing yourself with the:
 - > [java.net package](#)
 - which contains the `InetAddress` class
 - Which uses, for example the `getByName` and the `getLocalHost` methods

References: "An Intro to Network Programming with Java", Jan Graba; "Network Programming & Distr. Computing", Reilly & Reilly Dr. Nazi Mollah

Introduction

- Some relevant and exciting core packages of Java are [java.net](#) and [java.io](#) which contain a number of useful classes that allow network programmers to do cool stuff
- Using many of these classes will allow us to communicate with any server on the Internet or implement our own Internet server
- Over the semester we will use sockets to program TCP and UDP as well as Mail clients, and we will also look at network programming with GUIs – as we cover the many protocols
- The [java.io package](#) contains classes that support input and output
- The [java.net package](#) provides classes for implementing networking applications

References: "An Intro to Network Programming with Java", Jan Graba; "Network Programming & Distr. Computing", Reilly & Reilly Dr. Nazi Mollah

The `InetAddress` class and the `getByName` method

- The `InetAddress` class represents an Internet Protocol (IP) address
- This class handles Internet addresses both as host names and as IP addresses
- The `static method getByName` of this class uses DNS (Domain Name System) to return the Internet address of a specified host name as an `InetAddress` object
- `getByName` method:
 - > **Parameters:**
 - host or null
 - > **Returns:**
 - an IP address for the given host name.
 - > **Throws:**
 - [UnknownHostException](#) - if no IP address for the host could be found

References: "An Intro to Network Programming with Java", Jan Graba; "Network Programming & Distr. Computing", Reilly & Reilly Dr. Nazi Mollah

Who needs `Whois.com` when you can make your own IP Finder

```
// CSCI 395
import java.net.*;
import java.util.Scanner;
import java.io.*;

public class IPFinder {
    public static void main(String[] args) throws IOException {
        String host;
        Scanner input = new Scanner(System.in);

        System.out.print("Enter host name: ");
        host = input.nextLine();

        try {
            InetAddress address = InetAddress.method(parameter);
            System.out.println("IP address: " + address.toString());
        }
        catch (UnknownHostException e) {
            System.out.println("Could not find " + host);
        }
    }
}

fill in the blanks
```

References: "An Intro to Network Programming with Java", Jan Graba; "Network Programming & Distr. Computing", Reilly & Reilly Dr. Nazi Mollah

The `InetAddress` class and the `getLocalHost` method

- The `static method getLocalHost` of this class returns local host
- `getLocalHost` method:
 - > **Returns:**
 - the IP address of the local host.
 - > **Throws:**
 - [UnknownHostException](#) - if no IP address for the host could be found.

References: "An Intro to Network Programming with Java", Jan Graba; "Network Programming & Distr. Computing", Reilly & Reilly Dr. Nazi Mollah

Who needs `command line` and `Ipconfig` when you can make your own

```
// CC 395
import java.net.*;

public class MyLocalIPAddress {
    public static void main(String[] args) {
        try {
            InetAddress address = InetAddress.method();
            System.out.println(address);
        }
        catch (UnknownHostException e) {
            System.out.println("Could not find local IP");
        }
    }
}

fill in the blanks
```

References: "An Intro to Network Programming with Java", Jan Graba; "Network Programming & Distr. Computing", Reilly & Reilly Dr. Nazi Mollah

Submissions and Resources

Submit as follows

Hardy395

to the following respective folders

IPFinder (10 points) MyLocalIPAddress (10 points)

- For extra credit (5 points) – in case you feel ambitious – add some GUI elements. For those of you who are new to Java, here is a [very simple starter lab](http://cs.millersville.edu/~cswweb/lib/userfiles/Lab_1b.pdf) that may give you some ideas for a basic interactive GUI.
http://cs.millersville.edu/~cswweb/lib/userfiles/Lab_1b.pdf
- The hard submission deadline for these 2 labs are 9/16 before 11:59PM