ITI 1620 Automne 2005

Introduction à l'informatique I

Manuel de laboratoire

Alan Williams
(Traduction : Daniel Amyot)

Table des matières

Laboratoires

Les laboratoires pour ITI 1620 auront lieu dans les salles STE 0130, STE 0131, or STE 2052. La salle STE 0110 contient le laboratoire général, accessible hors des périodes de laboratoire désignées. Les ordinateurs de ces quatre salles sont configurés de la même façon et permettront l'accès à vos fichiers.

Les ordinateurs eux-mêmes sont pour la plupart de la marque IBM/Lenovo, et le système d'exploitation utilisé est Windows XP. Veuillez visiter http://www.site.uottawa.ca/local/infolabos/indexf.shtml pour de plus amples renseignements.

It is also possible to work on your own computer, if you install the Java Development Kit (JDK). The JDK is available at the web site http://java.sun.com. It is also recommended to obtain the "Dr. Java" development environment available from http://www.drjava.org. For information about how to install the software needed to work on your own computer, see the web site http://www.site.uottawa.ca/~awilliam/iti1620/WorkAtHome.html.

Course accounts

Using any of the SITE labs requires a SITE computer account. This is different from your general university account. Your university account user identification (user name or "userid") will be similar to:

`awill039` (a contraction of first initial, family name, and three digits)

The SITE account will have the following form of userid:

`u0618234` ('u' followed by the student number)

The general university account will be used for official e-mail communications – even for the ITI 1620 course – and e-mail should be checked there regularly. The SITE account is only for access to the SITE workstations and network.

If your major is computer science or software engineering, your SITE account will be active for the entire length of your studies. If your major is not one of the above, your SITE account will be activated only during terms in which you take a course with course codes starting with ITI, CSI, or SEG.

Generating a course account

Generating an account is an operation that needs to be done only once at the start of the term. When you generate an account, space on a central disk server will be reserved for you. You will be able to access files stored on this server from any SITE lab computer.

To start to generate an account, you need to access the login screen. You may need to move the mouse on a computer if the screen is blank or a screen saver is running. Otherwise, you should see the “Welcome to Windows XP” screen.

Press all three of the CTRL, ALT, and DEL keys simultaneously to access the login screen. [This is referred to as “control-alt-delete”.] The SITE login window should appear.

Normally, you would login with your SITE userid and password. To generate your account, use “`autopass`” as the userid, and “`autopass`” as the password. As you type the characters of your password, they will appear as • characters so that your password cannot be read from the screen. Click the “Login” button to proceed.

The Account Generation system will ask for your student number, and then prompt you twice for a lab password of your choice. It will then ask you to authenticate yourself using a PIN (personal identification number). This is your InfoWeb student services password, which you should have received as part of your registration package. No other password will work. Your Infoweb Student Services password (or PIN) will remain unchanged; it has nothing to do with your SITE account password. The lab password that you have supplied will be the one used for your SITE lab account.

If you do not know what your InfoWeb password is, click on the 'PIN?' button. This will take you to a page that will allow you to obtain your InfoWeb password.

If all went well, the account generation program will indicate which labs your account will enable you to use.

Problems?

- If your PIN (Infoweb password) was not correct, you will need to find out what is your PIN (it was sent to you in your registration package). You should contact InfoService, at http://www.uottawa.ca/student/infoservice/ or at 562-5700.

- If you are not registered to a course that grants access to an account, the system will indicate this. Note that the registration information that is used to generate accounts may be few days old. If you have registered just recently, come back in a day or two.

- If you forget your password, simply regenerate your account using the same procedure.

- If the Account Generation system fails because of system errors (this is unfortunately unavoidable), report this to the consultant on duty, then try again later.

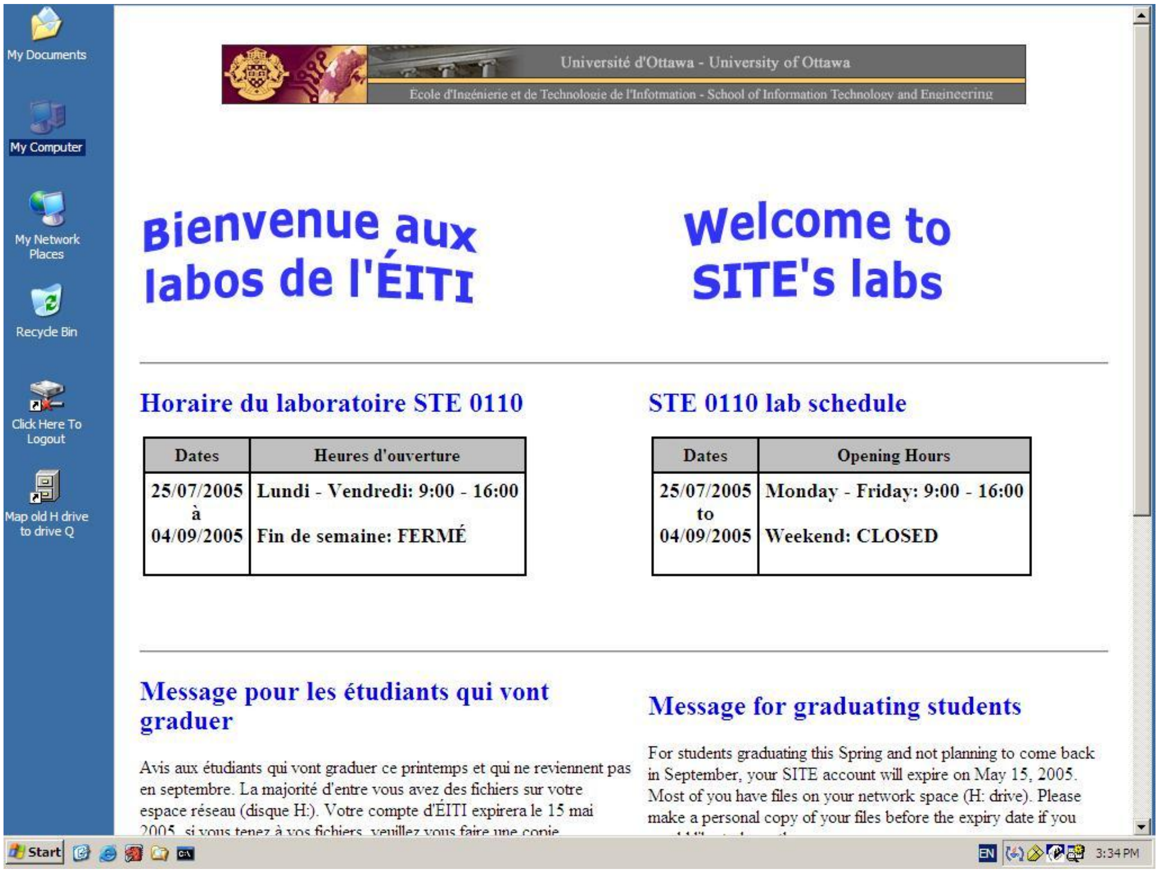
Logging In

An unoccupied computer should be showing the "Welcome to Windows XP" screen. If the computer has been idle for a while, the screen may be blank and you should wiggle the mouse or press any key to turn on the screen.

Press all three of the CTRL, ALT, and DEL keys simultaneously to access the login screen. [This is referred to as "control-alt-delete".] The SITE login window should appear.

Enter your userid in the "user name" area. Type your password in the password box. As you type the characters of your password, they will appear as • characters so that your password cannot be read from the screen. Click on the "Login" button or hit enter to log in.

The login process may take a few seconds. When the login process has finished, the screen should be similar to what is shown in Figure 1, although the information shown is updated regularly with the latest news.



Université d'Ottawa - University of Ottawa

Ecole d'Ingénierie et de Technologie de l'Infotmation - School of Information Technology and Engineering

Bienvenue aux labos de l'ÉITI

Welcome to SITE's labs

Horaire du laboratoire STE 0110

Dates	Heures d'ouverture
25/07/2005 à 04/09/2005	Lundi - Vendredi: 9:00 - 16:00 Fin de semaine: FERMÉ

STE 0110 lab schedule

Dates	Opening Hours
25/07/2005 to 04/09/2005	Monday - Friday: 9:00 - 16:00 Weekend: CLOSED

Message pour les étudiants qui vont graduer

Avis aux étudiants qui vont graduer ce printemps et qui ne reviennent pas en septembre. La majorité d'entre vous avez des fichiers sur votre espace réseau (disque H:). Votre compte d'ÉITI expirera le 15 mai 2005, si vous tenez à vos fichiers, veuillez vous faire une copie

Message for graduating students

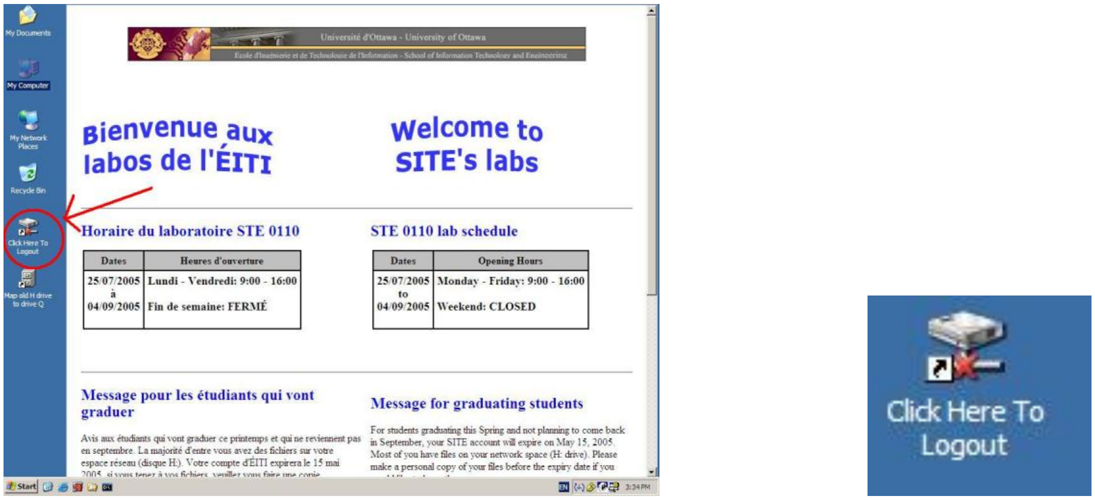
For students graduating this Spring and not planning to come back in September, your SITE account will expire on May 15, 2005. Most of you have files on your network space (H: drive). Please make a personal copy of your files before the expiry date if you

Figure 1: Screen after a successful login

You can now use the computer, and you will have access to your files stored on the central file server.

Logging Out

When you are finished your session, it is critical that you log out, to protect your files from the next user. Double-click on the following icon to log out:



My Documents
My Computer
My Network Places
Recycle Bin
Click Here To Logout
Map old H drive to drive Q
Université d'Ottawa - University of Ottawa
École d'Ingénierie et de Technologie de l'Information - School of Information Technology and Engineering
Start
EN
3:34 PM
Click Here To Logout

Bienvenue aux labos de l'ÉITI

Welcome to SITE's labs

Horaire du laboratoire STE 0110

Dates	Heures d'ouverture
25/07/2005 à 04/09/2005	Lundi - Vendredi: 9:00 - 16:00 Fin de semaine: FERMÉ

STE 0110 lab schedule

Dates	Opening Hours
25/07/2005 to 04/09/2005	Monday - Friday: 9:00 - 16:00 Weekend: CLOSED

Message pour les étudiants qui vont graduer

Avis aux étudiants qui vont graduer ce printemps et qui ne reviennent pas en septembre. La majorité d'entre vous avez des fichiers sur votre espace réseau (disque H:). Votre compte d'ÉITI expirera le 15 mai 2005, si vous tenez à vos fichiers, veuillez vous faire une copie

Message for graduating students

For students graduating this Spring and not planning to come back in September, your SITE account will expire on May 15, 2005. Most of you have files on your network space (H: drive). Please make a personal copy of your files before the expiry date if you

Figure 2: The logout icon

Do NOT turn the power off or shut down the machines.

The SITE Environment

Here are some important features of the SITE lab environment that may be different from your own computer.

On a Windows system, there are several different methods of storing files on your computer. These are represented by "disk drive" letters. While these can vary among different computers, you may be familiar with "C" being the internal disk drive, "A" being a diskette drive, and "D" for a CD or DVD drive.

If you store your files in the "My Documents" folder, these files are normally stored on the C drive. The C drive usually also includes all the software installed on a computer.

There are some differences in the SITE lab configuration for several reasons:

- You may be using a different computer during your next visit to the lab, and you will need access to your files.
- We want to be sure that you have access to only your own files during your session, and not those of students who used the computer previously.
- In case an important system file is deleted accidentally, we want to be able to restore the computer to its original state.

- We want to protect the SITE lab computers against malicious software such as viruses, worms, spyware, and ad-ware.

Therefore, keep the following in mind as it is very important:

When you log out from the computer, the C drive is restored to a standard configuration. This means that any files that you stored on the C drive will be erased at the end of a session. This includes files stored in the “My Documents” folder. These files cannot be recovered once erased.

To provide permanent secure file storage, and to provide access to your files from any SITE lab computer, your files on the central file server will be made available on drive H when you log in. Here are the notable drive letters you will see on a SITE lab computer.

A: The diskette drive. This can be used to transfer small files to other systems with similar drives. Although SITE lab computers have diskette drives, most newer computers do not include a diskette drive.

C: The local internal disk drive of the computer. The “My Documents” folder is on this drive. The contents of the C drive are not available from other computers.

E: The compact disc (CD) drive. The CD drives in the SITE labs are read-only; you will not be able to burn a CD from these computers, nor can you read from a DVD.

F: If you plug in a flash memory stick to the universal serial bus (USB) port, it will appear as drive F.

H: File space reserved for your account on the central file server. It will be available from any SITE lab computer, and can also be accessed from the Internet. This is where you should store files that you want to keep permanently.

You can store files temporarily for a session on the C drive, but any files that you want to keep permanently should be copied to the H drive (or put on a diskette or flash memory stick) before you log out.

There are other drives that store various software applications that are available on the computers.

Printing from SITE labs

SITE provides line printers and laser printers for SITE students. Line printers are free for both graduate and undergraduate students. All Students have access to Relmon printers and must purchase a copy card to print documents using the Relmon printers.

Relmon printers are maintained by Computing & Communications Services (CCS).

Information about the names, locations and usage of SITE's printers are summarized in the following table:

Printer name	Location	Description
uglp1	STE 0110	High speed line printer
uglp2	STE 0110	High speed line printer
relmon	CBY Hall, 1st floor. STE 0110 and STE level 2.	Pay-per-use laser printers. Requires a photocopy card.

Note that the high speed line printers are **not** capable of handling graphics or advanced fonts. If accented letters are sent, they will be printed without the accent. Use the line printers to print programs and rough copies of documentation.

When you print a file using the Relmon system, your file is sent to a central print server. The print server holds your files until you are ready to print (and pay) for them. You can use any of the Self-Service printers located around campus to print. The printers are always available, the only limit being the hours in which the building is unlocked. At each printer there is a terminal to display and select the file(s) you want to print, and a photocopy card reader to accept payment.

You pay for printing in the same manner as when you use photocopiers on campus. There are copy card dispensers at various locations on campus from which you can purchase a copy card and load it with a monetary value. The card is then debited as you use the self-serve photocopiers or printers

Additional information is available from the following web sites:

- SITE printing documentation:

  http://www.site.uottawa.ca/local/labinfo/printing.shtml

- Guide to self-service printers on campus, and their locations.

  http://www.uottawa.ca/services/ccs/docs/cyberleap/relmon/relmone.html

The Virtual Campus

The Virtual Campus is the university's on-line course management system. The Virtual Campus provides the following facilities for ITI 1620:

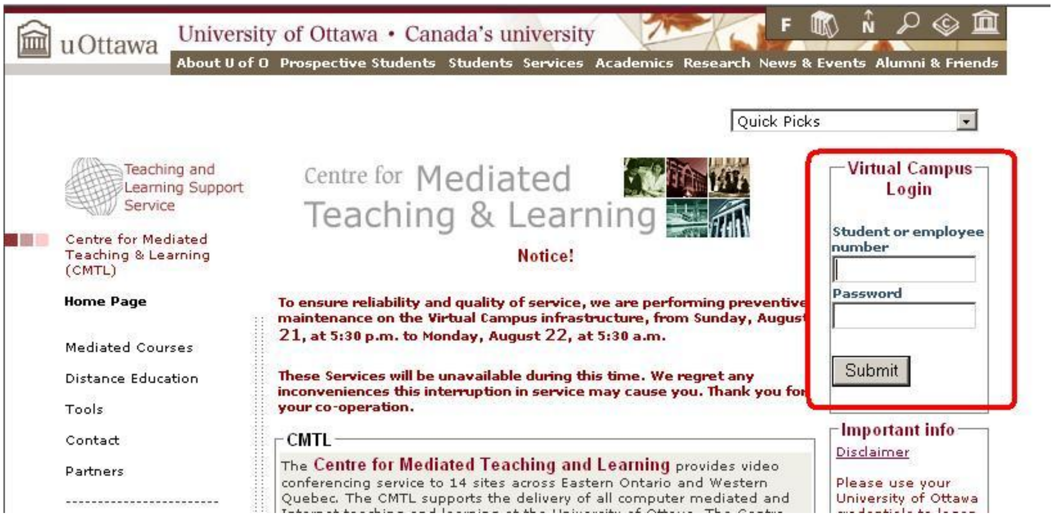
- Access to material from the lectures, labs, and tutorials
- Electronic assignment submission
- Discussion forum
- Current grades
- List of important dates

Students registered in ITI 1620 should have access to the ITI 1620 on-line course.

The virtual campus is available at the following web site:

http://maestro.uottawa.ca

After selecting your preferred language, you should see a screen similar to Figure 3.



Notice!

To ensure reliability and quality of service, we are performing preventive maintenance on the Virtual Campus infrastructure, from Sunday, August 21, at 5:30 p.m. to Monday, August 22, at 5:30 a.m.

These Services will be unavailable during this time. We regret any inconveniences this interruption in service may cause you. Thank you for your co-operation.

CMTL

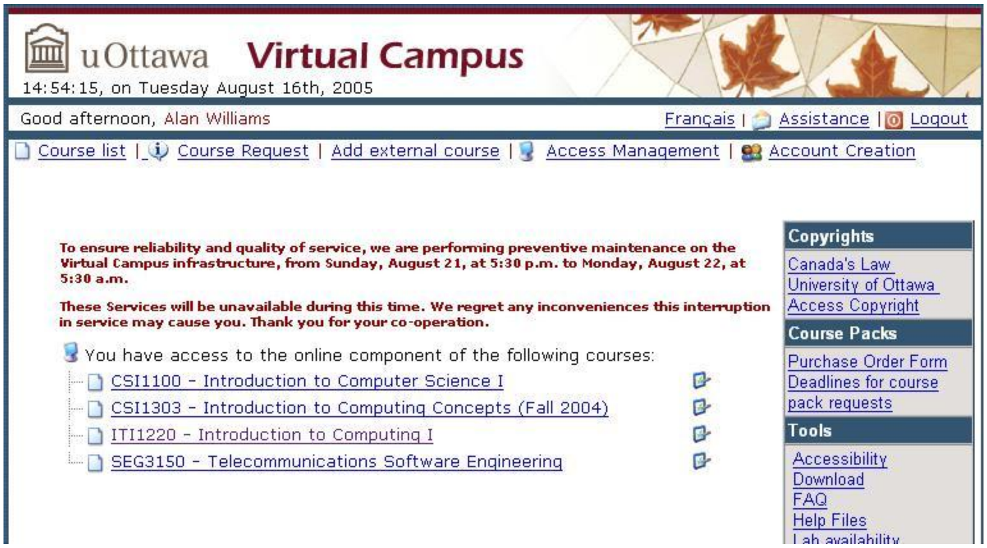
The Centre for Mediated Teaching and Learning provides video conferencing service to 14 sites across Eastern Ontario and Western Quebec. The CMTL supports the delivery of all computer mediated and Internet teaching and learning at the University of Ottawa. The Centre

Please use your University of Ottawa credentials to logon

Figure 3: The Virtual Campus login screen

The login area is in the upper right corner of the web page. To log in, enter your student number and your InfoWeb password (do not use the password from your SITE account).

After logging in, you will see a screen similar to Figure 4, where the course list shows the courses to which you have access.


uOttawa Virtual Campus
14:54:15, on Tuesday August 16th, 2005
Good afternoon, Alan Williams
Français | Assistance | Logout
Course list | Course Request | Add external course | Access Management | Account Creation
Copyrights
Canada's Law
University of Ottawa
Access Copyright
Course Packs
Purchase Order Form
Deadlines for course pack requests
Tools
Accessibility
Download
FAQ
Help Files
Lab availability

To ensure reliability and quality of service, we are performing preventive maintenance on the Virtual Campus infrastructure, from Sunday, August 21, at 5:30 p.m. to Monday, August 22, at 5:30 a.m.

These Services will be unavailable during this time. We regret any inconveniences this interruption in service may cause you. Thank you for your co-operation.

You have access to the online component of the following courses:

- CSI1100 - Introduction to Computer Science I
- CSI1303 - Introduction to Computing Concepts (Fall 2004)
- ITI1220 - Introduction to Computing I
- SEG3150 - Telecommunications Software Engineering

Figure 4: Virtual Campus Course List

Click on the ITI 1620 course to access the course content. The course home screen appears as in Figure 5 (it may vary as the term progresses).

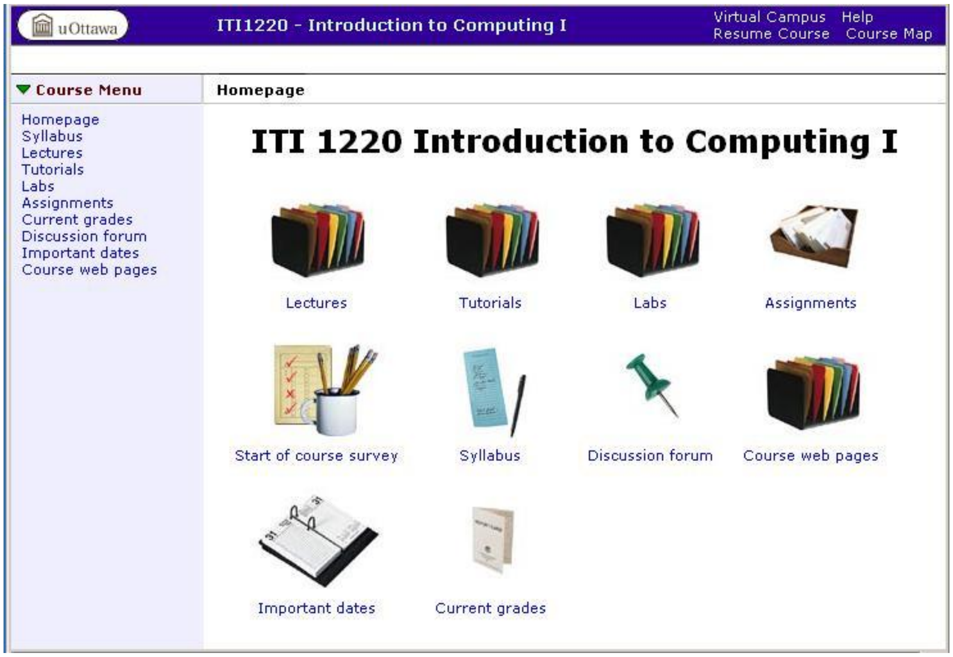


Figure 5: ITI 1620 Virtual Campus Home Page

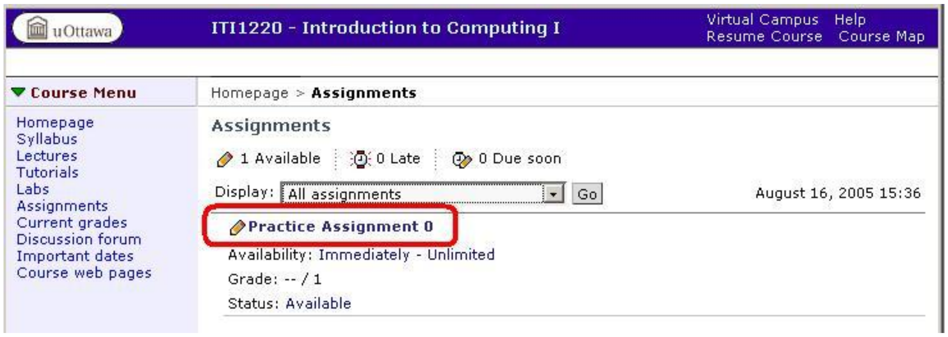
There are folders containing material from the lectures, tutorials, and labs.

The discussion forum is essentially a bulletin board where you can post messages for everyone in the course to read. All students, all teaching assistants, and the professor will be able to read any message that is posted. The professor reserves the right to delete any inappropriate messages posted to the forum; keep in mind that your name will be posted along with the message.

The “current grades” tool lets you see how you are doing in the course. Assignment, midterm, and final exam grades will be available here.

The discussion forum is the preferred way to ask questions about the course. If you have a question, it is likely that someone else may be wondering about the same question. If you post the question, everyone can benefit from the answer.

An important component of the virtual campus is the on-line assignment submission and return system. Assignment descriptions will be made available on-line only; they will not be printed on paper. If you click on the “assignments” icon, you will see the list of assignments for the course. Assignments will become available as the course progresses. Figure 6 shows a sample view of the virtual campus assignments page.



Assignments

1 Available | 0 Late | 0 Due soon

Display: All assignments

August 16, 2005 15:36

Practice Assignment 0

Availability: Immediately - Unlimited

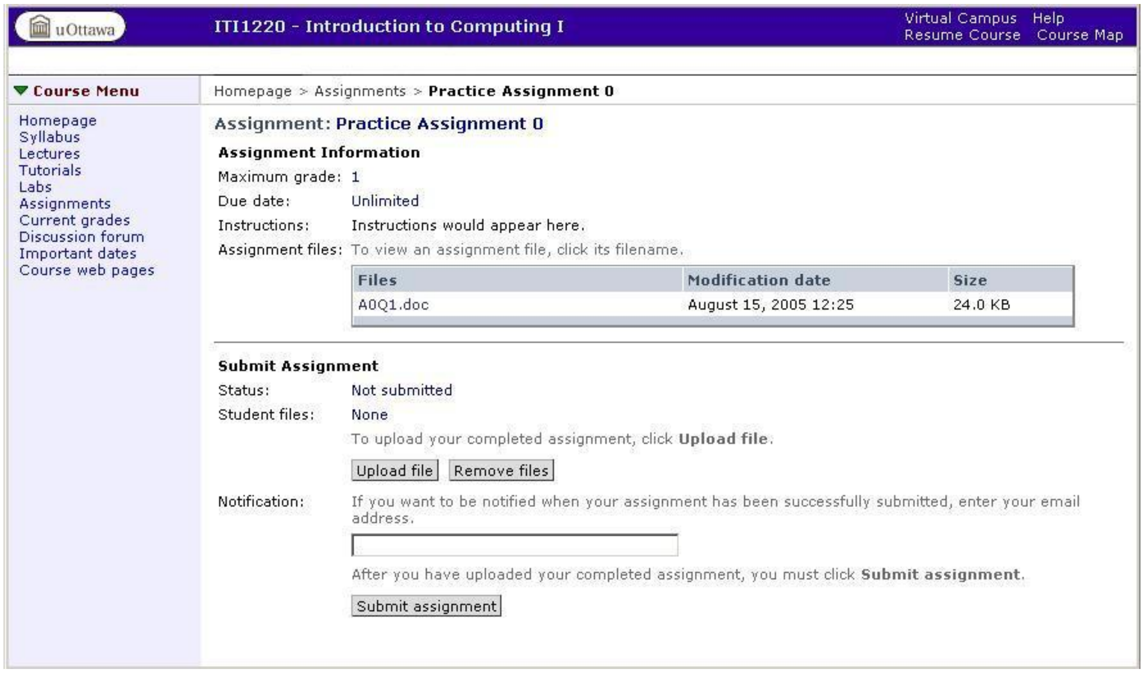
Grade: -- / 1

Status: Available

Figure 6: Virtual Campus Assignment Page

To view a particular assignment, click on its title (as highlighted in Figure 6). You will see a screen that is similar to Figure 7.

The maximum grade for the assignment and its due date are shown at the top of the page. The instructions area will provide general instructions, such as whether the assignment is to be done individually or as part of a team. The assignment files area will contain the actual files needed for the assignment. One of these files will be the assignment description, and there may be additional files that contain partial programs, or other needed material. In Figure 7, there is a file called `A0Q1.doc`. If you click on the file name, you will have the choice to view the file or download it to your computer.



Homepage > Assignments > **Practice Assignment 0**

Assignment: **Practice Assignment 0**

Assignment Information

Maximum grade:	1
Due date:	Unlimited
Instructions:	Instructions would appear here.
Assignment files:	To view an assignment file, click its filename.

Files	Modification date	Size
A0Q1.doc	August 15, 2005 12:25	24.0 KB

Submit Assignment

Status:	Not submitted
Student files:	None

To upload your completed assignment, click **Upload file**.

Notification: If you want to be notified when your assignment has been successfully submitted, enter your email address.

After you have uploaded your completed assignment, you must click **Submit assignment**.

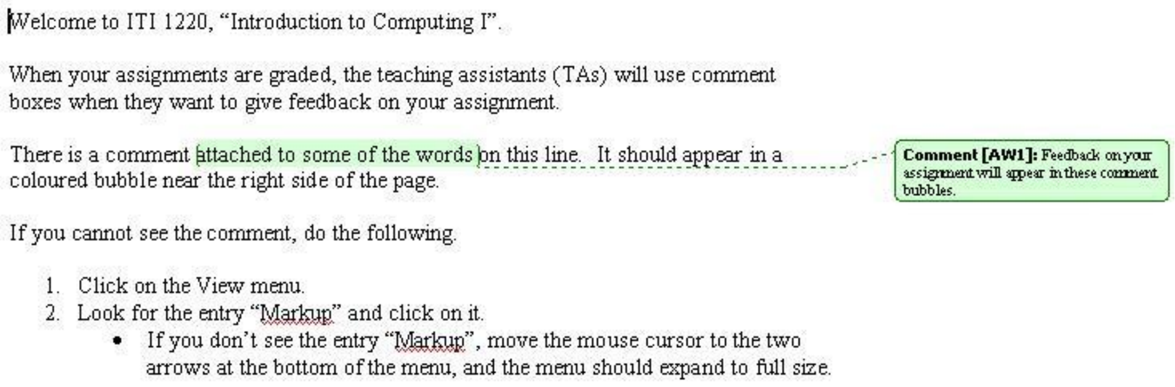
Figure 7: An assignment on the virtual campus

The lower part of the page is used to submit an assignment. When you have prepared the file you want to submit as your assignment, click the “Upload file” button. This will allow you to locate and select your assignment file. Once you have uploaded the file, check to make sure that you have the correct file. If you would like to receive an e-mail confirmation when the assignment is submitted, enter the address to which you would like the message sent.

Click on “Submit assignment” to complete the process. At this point, the status of your assignment will change from “Not submitted” to “Not graded.”

For ITI 1620 assignments, you will be allowed to remove the files and re-submit your assignment at any time up to the assignment due date and time. At that point, all files are locked in, and the teaching assistants can start grading the files.

When your assignment has been graded, the status will change to “Graded”, and there will be an area where you can download the files that include feedback from the teaching assistant who graded the assignment. There is also a comments box and the actual grade. Files provided from the teaching assistant will use the Microsoft Word reviewing features to provide feedback, as shown in Figure 8.



Welcome to ITI 1220, "Introduction to Computing I".

When your assignments are graded, the teaching assistants (TAs) will use comment boxes when they want to give feedback on your assignment.

There is a comment attached to some of the words on this line. It should appear in a coloured bubble near the right side of the page.

Comment [AW1]: Feedback on your assignment will appear in these comment bubbles.

If you cannot see the comment, do the following.

1. Click on the View menu.
2. Look for the entry "Markup" and click on it.
   - If you don't see the entry "Markup", move the mouse cursor to the two arrows at the bottom of the menu, and the menu should expand to full size.

Figure 8: Sample of assignment feedback

When viewing the files, be sure that you have the "Markup" feature turned on in Microsoft Word. The text in Figure 8 describes how to do this.

How to submit assignments

It is very important to read this section, as marks will be deducted if the procedures described here are not followed.

- All assignments must be submitted electronically using your virtual campus account.

- **Don't wait until the very last second to submit your assignment; processing may be slow near deadlines.**

All the files for one assignment must be placed in a folder named, `aX_studentId`, for example, student 0618234 submitting assignment 4 would put all assignment files in a folder named `a4_0618234`. The folder must contain no files except those related to the assignment. If you are working as a team, choose only one member of your team to submit the assignment, and use the student number of that person for the folder name.

Please ensure that you use this naming convention, as the teaching assistants will have files from many students and multiple assignments downloaded to their computers. We want to ensure that each assignment submission has a unique name and avoid problems with duplicate file names.

For each question on the assignment the folder should contain:

- A Microsoft Word file (`.doc`) file or Rich Text Format (`.rtf`) file. If you are working on a machine that does not have Microsoft Word installed, you can use the "Wordpad" program that comes with Windows to create an `.rtf` file.

- For questions that do not involve programming (e.g. software models and diagrams), this file will contain the answer to the question. In this case, this is the only file that is necessary.

- For questions that involve programming, this file will be used for the teaching assistant to give comments about your code. Create the `.doc` or `.rtf` file as follows:

  - Set the font to be “Courier new” and the font size to be 10.

  - Select all text in the `.java` file. Copy it and paste it to the Word or RTF file.

- Java source code files `(.java`)

  - It is only necessary to include code from .java files that you have created, or modified from the original version posted with the assignment. Files with Java code that were posted with the assignment but were not modified by you can be omitted.

- Class files (`.class`) compiled from the `.java` files.

- The names of the file containing the `main` method must indicate the assignment and question numbers: for example, if question 2 on assignment 4 requires submitting a program, you should have 3 files for that question named `A4Q2.doc`, `A4Q2.java`, and `A4Q2.class`.

- If the program requires more than one Java class, additional `.java` and `.class` files should be included. For example, if question 2 on assignment 4 also uses a class that you have created called `Hello`, submit the files `Hello.java`, `Hello.class`, and `Hello.doc`. Classes without `main` methods do not have to follow the `AxQy` convention.

Create a `.zip` file containing the contents of the entire folder. To “zip” a collection of files is to do the two following operations:

- Compress each file so that it uses the minimum amount of space.

- Put all of the compressed files inside of one file, so that everything is in one file in a way that the original folders can be recreated.

In Windows, this operation is called “Send to compressed folder”. There are also many file compression programs available (see warning below). For example, “Ultimate Zip” is installed on SITE lab computers.

Your assignment submission file should be named `aX_StudentId.zip`, where `aX_StudentID` was the name of the original directory. For example, student 0618234 submitting assignment 4 should create a zip file with the name `a4_0618234.zip`

To do this using one of the SITE lab computers, go to the parent folder of `aX_StudentID` and click the right mouse button on the icon of the folder `aX_StudentID`. Select "`Add to aX_StudentID.zip`". This will create the zip file, which is a single file that contains compressed versions of your original files.

- Submit the zip file via the virtual campus assignment submission system (click here to go to the Virtual Campus)

- Assignments that do not conform to the above procedure will not be graded.

WARNING: Some file compression programs produce files in "RAR" format. This is NOT the same format as the ZIP format. The SITE labs do not have a program that can read `.rar` files. If you use a file compression program on your computer instead of the default "Send to compressed folder" from Windows, check very carefully that it produces .zip files. **If you submit a `.rar` file, and the teaching assistant cannot read the file, you will get zero for the assignment.**

Identification

The following information must be included at the start of every `.doc` file and as comments at the start of every `.java` file. Every program should also print this information as the first part of the output.

- Course: IT1 1620

- Assignment number

- Question number

- For each team member:

  - Family name, Given name

  - Student number

  - Section: A or B

Programming Standards

In industry, coding conventions help to reduce errors during program maintenance; in most cases, it will be someone else who has to understand the program well enough to change it without causing new problems. Many companies also have mandatory code inspections for quality control, where you are not allowed to contribute program code to a software product until it has been read out loud by someone else in a meeting attended by several software developers. Using standard coding conventions helps make your programs easier to understand by others (especially teaching assistants!).

For ITI 1620, approximately 10% of the grade for each assignment will be for following the course programming standards, and marks will be deducted for violations.

It is recommended to follow the general form of the Java code conventions at:
http://java.sun.com/docs/codeconv/html/CodeConvTOC.doc.html
The above web site is very detailed, and includes many parts of Java that are beyond the scope of ITI 1620.

In addition, for ITI 1620 the following standards **MUST** be respected:

Input / Output

- Output should be nicely spaced and easy to understand. In particular, it should be easy to understand the output WITHOUT looking at the program listing. This means that for every value in your output you must print a short message explaining the meaning of the value.

- Before your program requests input values from the user, it must print a prompt describing what the user is required to enter.

Variables

- Identifiers should be as meaningful as possible. Follow the standard upper-case/lower-case conventions for Java programs.

    - Method names and variables usually start with a lower case letter and then any additional words that are part of the name are capitalized. There are no spaces or underscores between words. The first word of a method name is often a verb.

        - Examples: `calculateStudentGrade( ), aNumber`

    - Class names should start with an upper case letter. Class names are generally a noun.

        - Examples: `Student, Course`

- Only declare variables that are actually used.

- Do not use the same variable for different purposes.

- Every variable should have a comment describing its purpose. Be brief. This is called a data dictionary - it should be placed at the beginning of every method, either before or beside the declarations.

- Do not use "global" variables. All values needed by a method must be passed into it as parameters and all values it produces or modifies must be returned as results or passed through parameters.

Program Structure

- Good structure is very important. Decompose your algorithms/methods into meaningful sub-algorithms/methods whenever this improves clarity.

- Each method (including "main") should be preceded by comments describing its purpose and the algorithm it uses (if the algorithm is very simple, it need not be described). These descriptions should be brief.

- Your program must be indented in a systematic way that reflects the nesting of its statements.

- The following are some exceptions to the coding conventions at Sun’s web site:

  - Branch statements must always follow the following form – namely, the braces and else clause must always be present and formatted as shown even when the Java language specification allows them to be omitted. Multiple if {} else if {} else {} statements should not be used.

    ```
    if ( condition )
    {
       // the true part
    }
    else
    {
       // the false part
    }
    ```

  - If either part requires no action, it should be as follows:

    ```
    {
       ; // do nothing
    }
    ```

  - Loops must follow one of the two forms – namely the braces must always be present and formatted as shown even when the Java language specification allows them to be omitted.

    ```
    while ( condition )
    {
       // the loop body
    }
    ```

    (or)

```
for ( initialization; condition; increment )
{
   // the loop body
}
```

- The following should not be used:

```
do

switch

break

+=, -=. *=, /=

condition ? trueClause : falseClause
```

- Increment and decrement operators can only be used in the following forms

```
aVariable++;   // no other code on this line

aVariable--;   // no other code on this line

for ( initialization; condition; loopIndex++ )
```

Restrictions

- Everything you submit (files and programs) must compile and run using the Java Development Kit (JDK) version on PCs in the SITE labs. “It works on my own computer,” is not a valid excuse if there are compatibility problems with your program.

- You may only use features and library methods of Java that have been presented in class or labs at the time an assignment is due. For example, you may not use Java's arrays before they are covered.

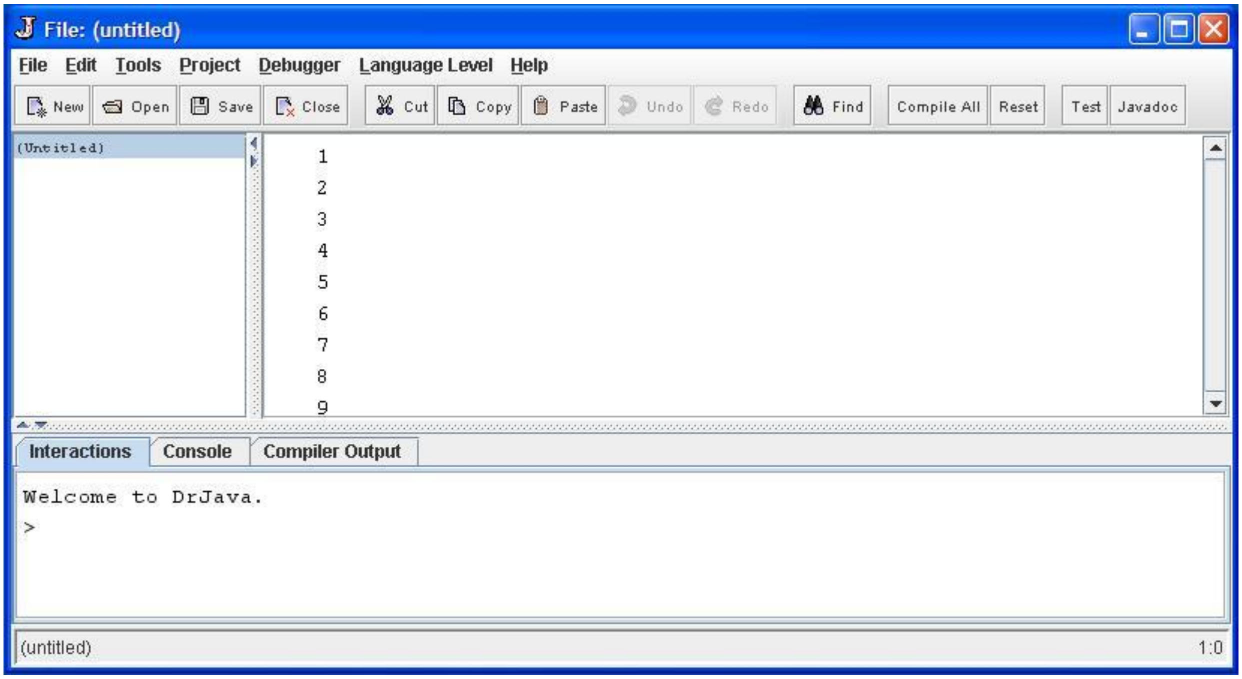
The Dr. Java Environment

The Dr. Java environment is an “integrated development environment” (IDE) that will be used for the ITI 1620 course. Dr. Java is available in the SITE labs, and you can also download your own copy for free from http://www.drjava.org. Dr. Java has been developed at Rice University to help students who are new to programming.

[Note that you can use any method that you wish to enter and compile programs. However, the teaching assistants will be able to give you help on using Dr. Java. If you use another IDE, you are on your own to learn how it works.]

If you start up Dr. Java, you will see a screen similar to Figure 9.

You may not see the line numbers; if you want to turn them on, click on the Edit menu and select “Preferences”. In the categories area, click on “Display Options” and then check the “Show all line numbers” box.



```
Welcome to DrJava.
>
```

Figure 9: Dr. Java window

The main screen area is partitioned into three parts. The upper right area is an editing area where you type in Java code. Dr. Java will help with Java code entry in the following ways:

- Java keywords, comments, and strings will be highlighted in various colours.
- Code will be indented properly.
- If there are problems compiling, the line with the problem will be highlighted.

The area on the left is a list of files that you are currently editing. In Figure 9, `(Untitled)` appears in this area because there is no file associated with the empty editing window. You can edit several files at once, and click on a file name in the list to move between files.

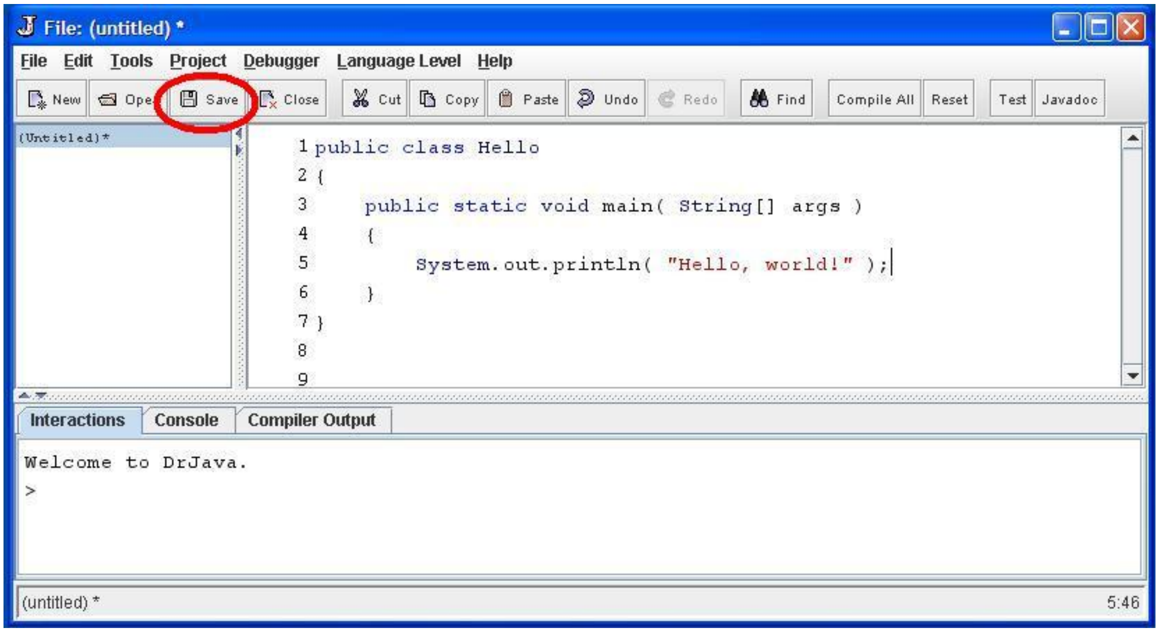
The area at the bottom has three views indicated by the “tabs”: Interactions, Console, and Compiler Output. Currently, the “Interactions” view is selected, and you can move from one view to the other by clicking on a tab. Sometimes, Dr. Java will change the view automatically when it is appropriate. When you compile a program, the output from the compiler will appear in the “Compiler Output” view. The “Console” is the area for text-based input and output for a Java program, and this is where you will see the results of running your program.

It is a tradition among programmers that when learning a new language that the first program you write will print “Hello, world!” In Java, the following program will print this message to the console:

```
public class Hello
{
    public static void main( String[] args )
    {
        System.out.println( "Hello, world!" );
    }
}
```

Be very careful to type this exactly as shown, including the semicolon on the line with the message to print. By entering the above, you have created a “class” called `Hello`. Java programs always consist of one or more classes.

After typing this into the editing area, the window should appear as follows:

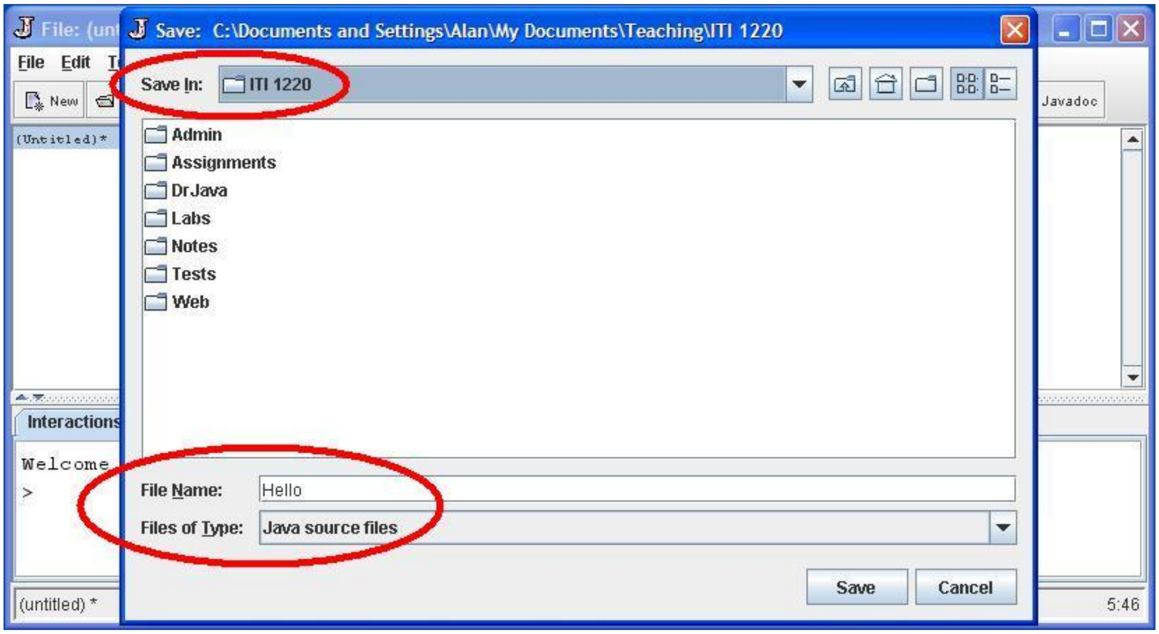

File: (untitled) *
File Edit Tools Project Debugger Language Level Help
New Open Save Close Cut Copy Paste Undo Redo Find Compile All Reset Test Javadoc
(Untitled)*
Interactions Console Compiler Output
(untitled) *
5:46

```
public class Hello
{
    public static void main( String[] args )
    {
        System.out.println( "Hello, world!" );
    }
}
```

```
Welcome to DrJava.
>
```

Figure 10: After entry of the "Hello, world!" program

You are now ready to save the file. Note that a * has appeared after `(untitled)` to indicate that there are changes that have not been saved. Click on the save button to save your file. When you do this, a screen similar to Figure 11 should appear:



Save In: ITI 1220

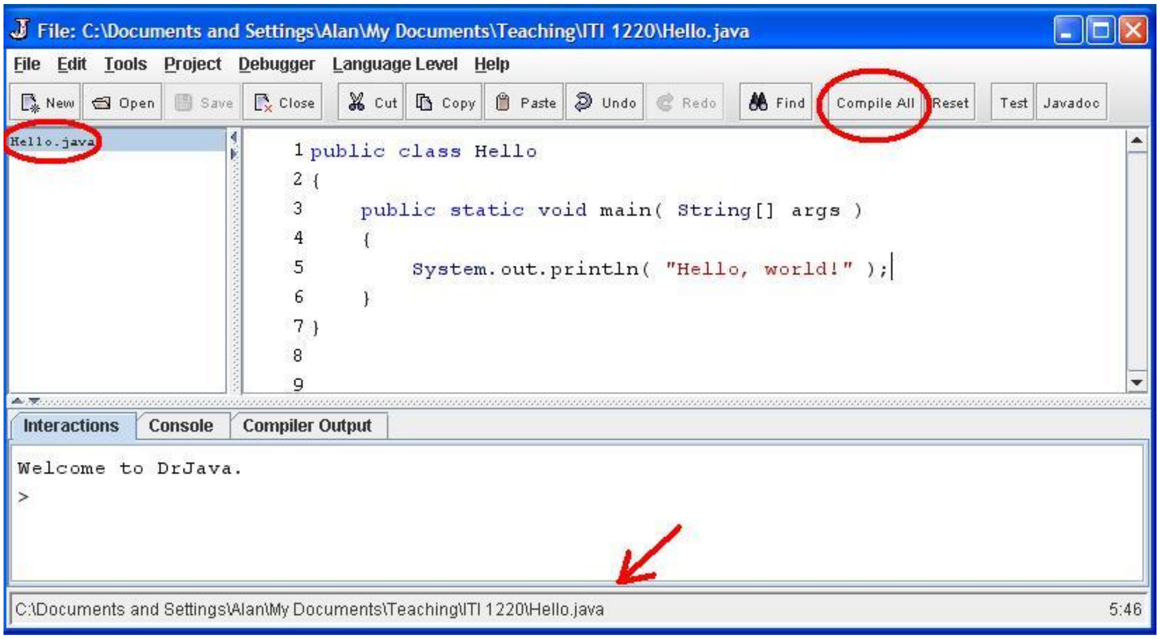
File Name: Hello

Files of Type: Java source files

Figure 11: Saving a file

The top area shows the folder in which the file should be saved. You can choose any folder you wish; in the above case, the file will be saved in the `ITI 1620` folder. Notice that in the “File name:” and “Files of type:” areas, Dr. Java has filled in “Hello” – the name of the class – and Java source files. This means that the proposed name of your file is `Hello.java`, within the `ITI 1620` folder **It is important for Java that the name of the file ends in `.java` to indicate Java source code, and it is also very important that the name of the file matches the name of the class inside the file.** Therefore, you should accept the proposed names by clicking Save.

After saving the file, the Dr. Java window should be similar to Figure 12. Note that `(untitled)*` in the left area has changed to `Hello.java`, and the * has been removed to indicate that there are no unsaved changes. The complete folder path of the file also appears at the bottom of the screen (see arrow).



File: C:\Documents and Settings\Alan\My Documents\Teaching\ITI 1220\Hello.java

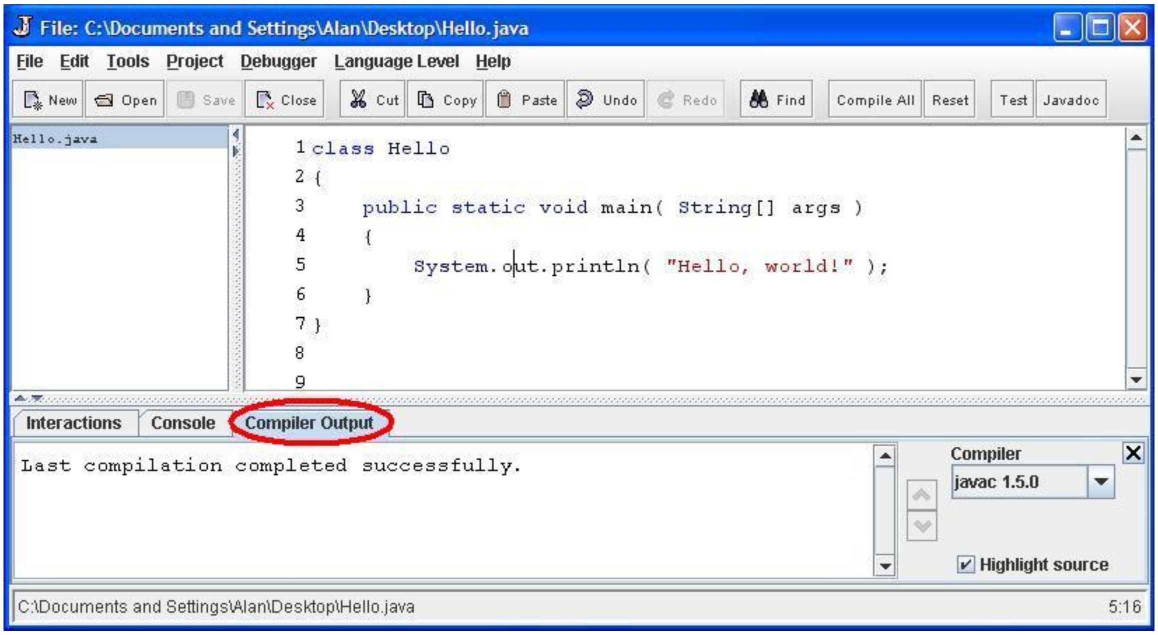
```
public class Hello
{
    public static void main( String[] args )
    {
        System.out.println( "Hello, world!" );
    }
}
```

```
Welcome to DrJava.
>
```

C:\Documents and Settings\Alan\My Documents\Teaching\ITI 1220\Hello.java

Figure 12: File saved, ready to compile

The buttons in the Dr. Java window related to operations with code are the "Compile All", "Test", and "Javadoc". A Java program must be compiled before it can be run. The compiler checks that your program obeys the rules of the Java language, and then translates the code into a format that the machine can understand. Clicking the "Compile All" button will compile **all** files currently listed in the left area of the screen. Output from the compiler will appear in the lower area. If everything was okay, you will see the message "Last compilation completed successfully." in the compiler output view. Try this now, and you should hopefully see a result similar to Figure 13.

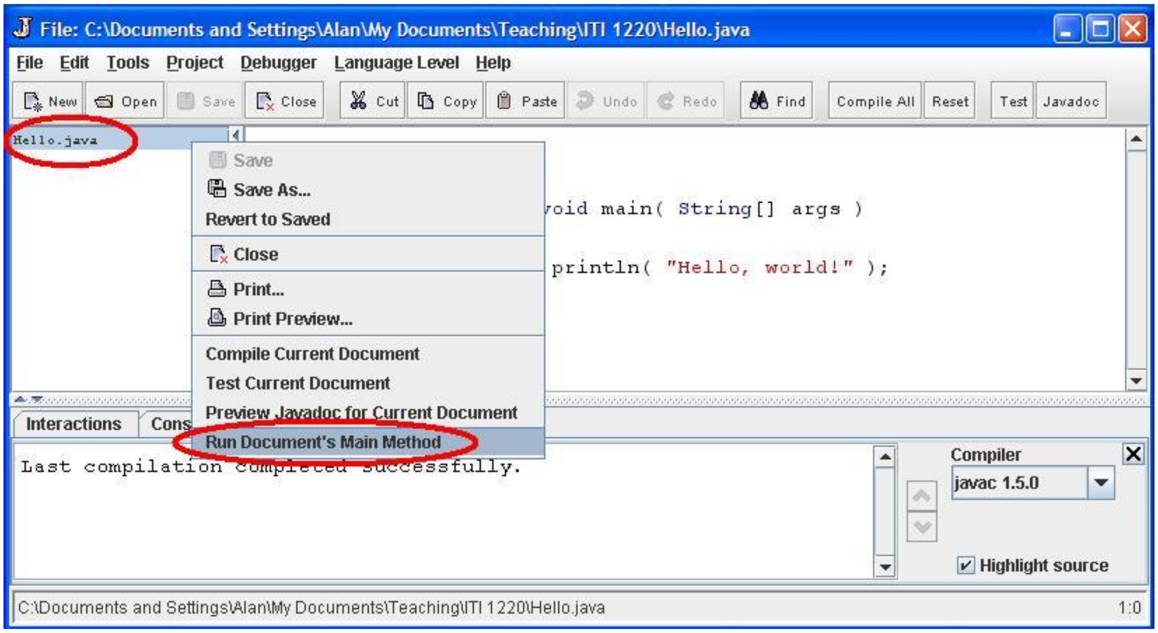

File: C:\Documents and Settings\Alan\Desktop\Hello.java
File Edit Tools Project Debugger Language Level Help
New Open Save Close Cut Copy Paste Undo Redo Find Compile All Reset Test Javadoc
Hello.java
Interactions Console Compiler Output
Compiler
javac 1.5.0
Highlight source
C:\Documents and Settings\Alan\Desktop\Hello.java
5:16

```
class Hello
{
    public static void main( String[] args )
    {
        System.out.println( "Hello, world!" );
    }
}
```

```
Last compilation completed successfully.
```

Figure 13: View after successful compilation

To run a program, you should select the file with the main method you wish to run in the left hand area and then click and hold the right mouse button to see the menu. The result should be similar to Figure 14.

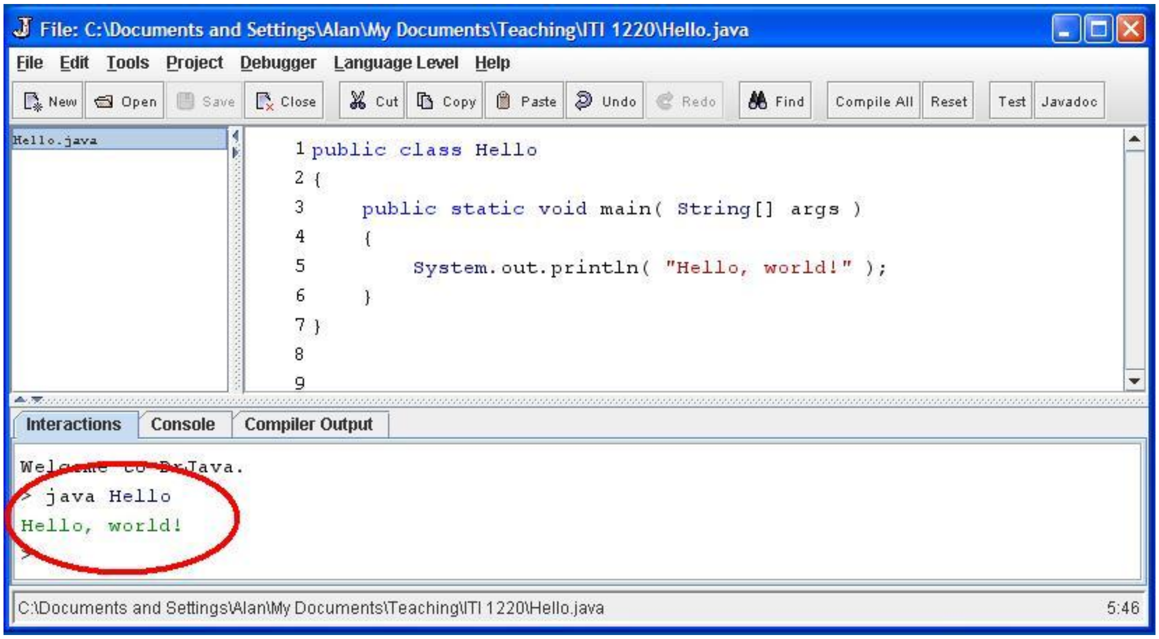

File: C:\Documents and Settings\Alan\My Documents\Teaching\ITI 1220\Hello.java
File Edit Tools Project Debugger Language Level Help
New Open Save Close Cut Copy Paste Undo Redo Find Compile All Reset Test Javadoc
Hello.java
Save
Save As...
Revert to Saved
Close
Print...
Print Preview...
Compile Current Document
Test Current Document
Preview Javadoc for Current Document
Run Document's Main Method
Interactions Cons
Compiler
javac 1.5.0
Highlight source
C:\Documents and Settings\Alan\My Documents\Teaching\ITI 1220\Hello.java
1:0

```
void main( String[] args )
println( "Hello, world!" );
```

```
Last compilation completed successfully.
```

Figure 14: Running a main method

If everything went well, you should see a view similar to Figure 15.


File: C:\Documents and Settings\Alan\My Documents\Teaching\ITI 1220\Hello.java
File Edit Tools Project Debugger Language Level Help
New Open Save Close Cut Copy Paste Undo Redo Find Compile All Reset Test Javadoc
Hello.java
Interactions Console Compiler Output

```
public class Hello
{
    public static void main( String[] args )
    {
        System.out.println( "Hello, world!" );
    }
}
```

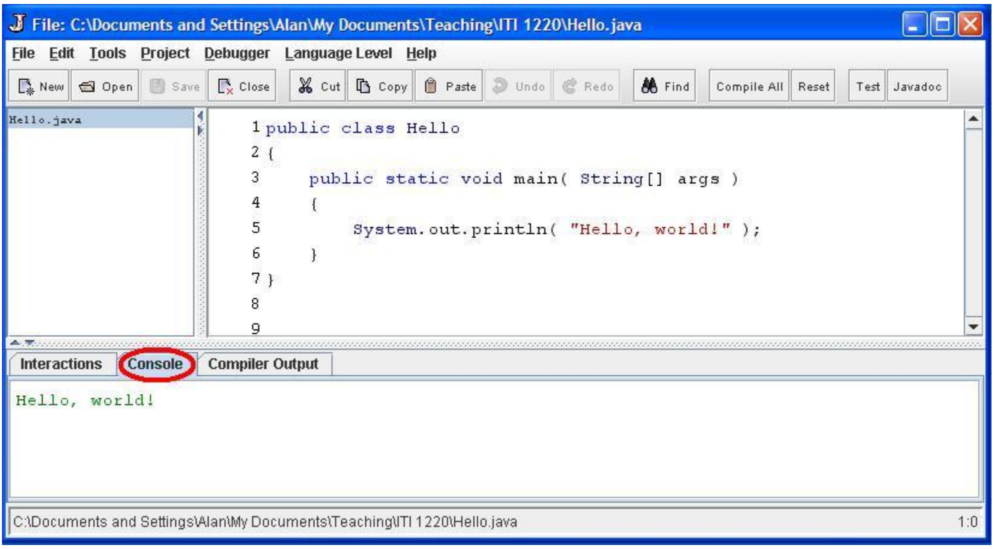
```
Welcome to DrJava.
> java Hello
Hello, world!
>
```

C:\Documents and Settings\Alan\My Documents\Teaching\ITI 1220\Hello.java 5:46

Figure 15: Result from running the program

To run a program from a console window, you could have typed `java Hello`, where java is the command to run a Java .class file. This command has appeared in the Interactions window as a result of selecting to run the method from the menu. Some people prefer to use menus, while others prefer to type commands.

Beneath the command is the output from your program. We have successfully printed “Hello, world!” If you want to see a view of the program output without the interaction commands, click on the console tab to get a view similar to Figure 16.


File: C:\Documents and Settings\Alan\My Documents\Teaching\ITI 1220\Hello.java
File Edit Tools Project Debugger Language Level Help
New Open Save Close Cut Copy Paste Undo Redo Find Compile All Reset Test Javadoc
Hello.java
Interactions Console Compiler Output

```
public class Hello
{
    public static void main( String[] args )
    {
        System.out.println( "Hello, world!" );
    }
}
```

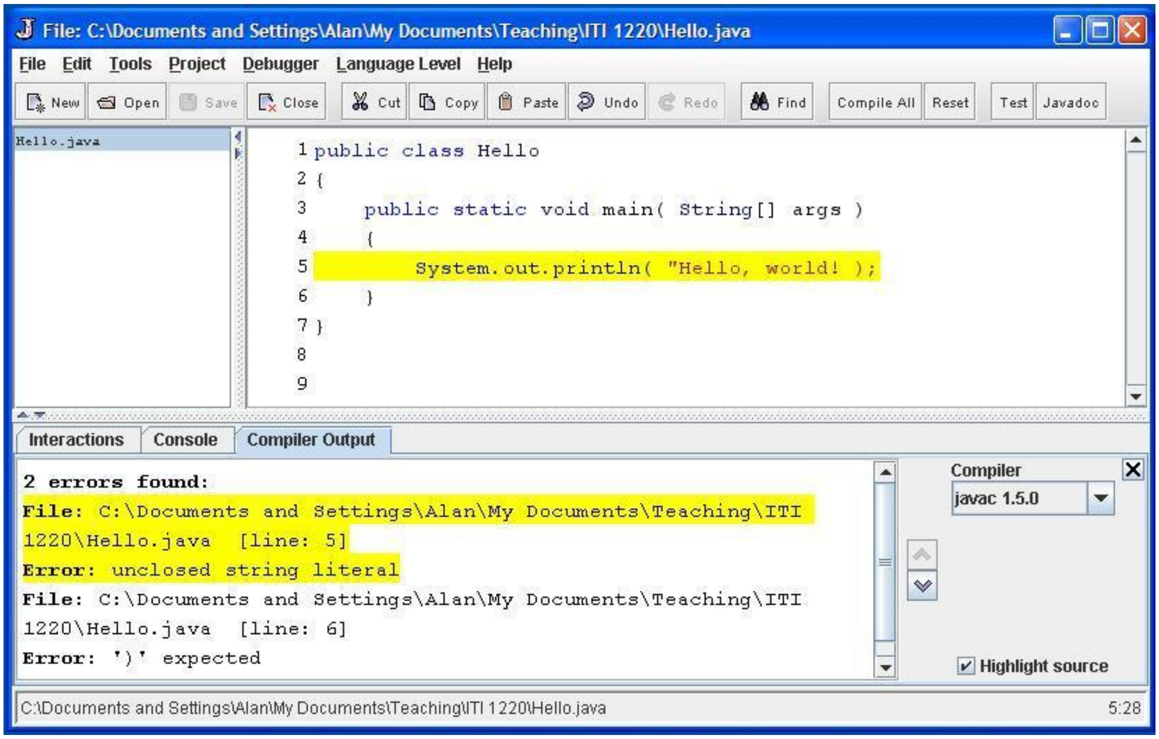
```
Hello, world!
```

C:\Documents and Settings\Alan\My Documents\Teaching\ITI 1220\Hello.java 1:0

.

Figure 16: Console view of program output

When there is a compile error, you may see something like Figure 17



```
public class Hello
{
    public static void main( String[] args )
    {
        System.out.println( "Hello, world! );
    }
}
```

```
2 errors found:
File: C:\Documents and Settings\Alan\My Documents\Teaching\ITI
1220\Hello.java  [line: 5]
Error: unclosed string literal
File: C:\Documents and Settings\Alan\My Documents\Teaching\ITI
1220\Hello.java  [line: 6]
Error: ')' expected
```

Compiler: javac 1.5.0

☑ Highlight source

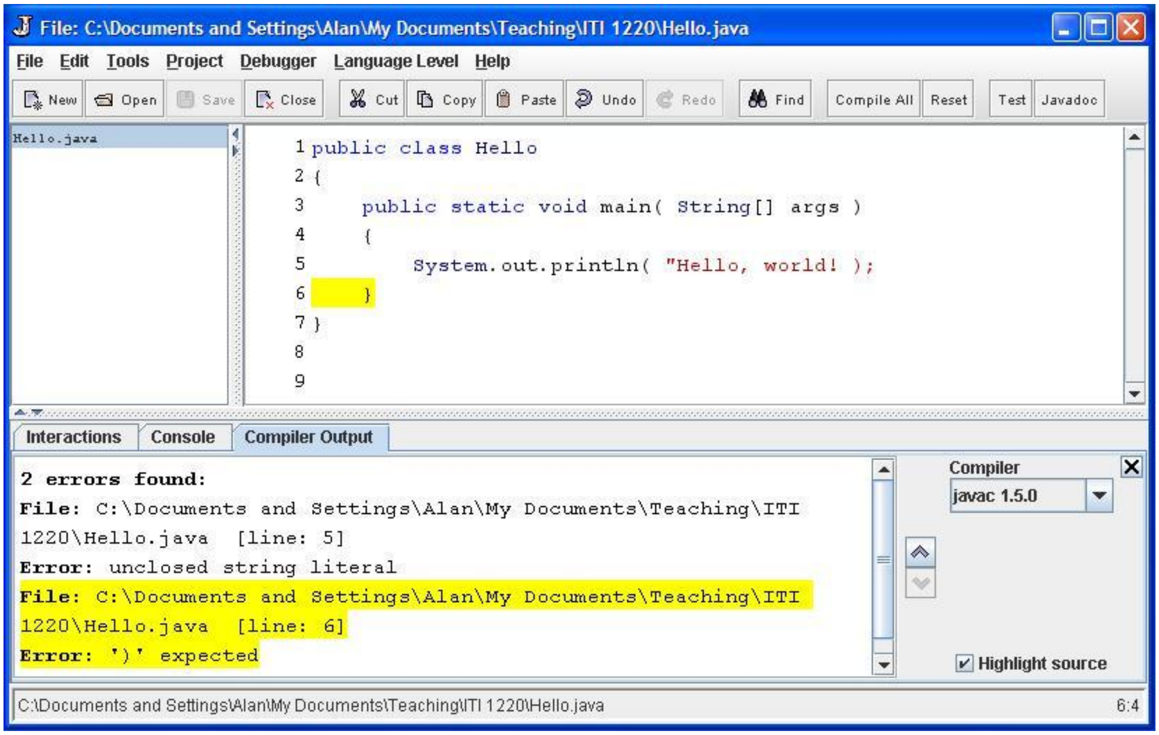
C:\Documents and Settings\Alan\My Documents\Teaching\ITI 1220\Hello.java 5:28

Figure 17: Result when there is a compile error

In this case, the closing `"` has been omitted in the `println( )` statement.

For each problem, the compiler message indicates the file and line number of the problem, followed by a message of what the compiler believes is the problem. For the first problem, both the message and the corresponding line in the file are highlighted in yellow. If you click on the second problem in the compiler output window, the highlighting will move to the location of that problem, as shown in Figure 18.

The message **`unclosed string literal`** means that the compiler could not find the closing double-quote character to indicate the end of what to print on the screen.


File: C:\Documents and Settings\Alan\My Documents\Teaching\ITI 1220\Hello.java
File Edit Tools Project Debugger Language Level Help
New Open Save Close Cut Copy Paste Undo Redo Find Compile All Reset Test Javadoc
Hello.java
Interactions Console Compiler Output
C:\Documents and Settings\Alan\My Documents\Teaching\ITI 1220\Hello.java
6:4

```
1 public class Hello
2 {
3     public static void main( String[] args )
4     {
5         System.out.println( "Hello, world! );
6     }
7 }
8
9
```

```
2 errors found:
File: C:\Documents and Settings\Alan\My Documents\Teaching\ITI
1220\Hello.java  [line: 5]
Error: unclosed string literal
File: C:\Documents and Settings\Alan\My Documents\Teaching\ITI
1220\Hello.java  [line: 6]
Error: ')' expected
```

Compiler: javac 1.5.0

☑ Highlight source

Figure 18: Locating errors

The second error message, `')' expected` is related to the same problem, and can be confusing because the `)` is actually present in the program. If you look carefully at Figure 18, you may notice that the characters `)` and `;` are coloured in red, and this means that the compiler is assuming that these characters are part of what is included in the message to be printed on the screen. It is only after the compiler reaches the end of the line that it realizes that the `"` is missing. Now, the compiler wants to match the `)` character for the `println( )` statement, but the next character is a brace on the following line (6) and the compiler reports an error at that point.

In fact, restoring the `"` will eliminate both error messages.

As you can see, interpreting compiler messages can be tricky and this is one of the things you will have to learn in the course. Sometimes, as in the case we have seen here, several compiler messages can result from just one problem in the code. In other cases, after fixing one compiler problem, more problems that were obscured by the first problem can appear during the next compile. If the messages from the compiler do not appear to make any sense, it helps to remember that the compiler reads one character of your program at a time in the forward direction, and will not back up. The compiler can only print a message when it is sure there is a problem, and that is why it cannot tell that the `)` was not meant to be part of the message to be printed.