Creating a Course Class

• An instructor has students in each class course.
  – Each course can have different students.
• For a course we might want to know
  – The instructor name
  – The students in the course
UML Class Diagram

- There is a standard way to diagram object-oriented classes
  - It is a UML class diagram
    - It shows the classes as boxes and the relationships between them

![UML Class Diagram]

Creating a Course class

- We want fields for the
  - Instructor’s name
  - Students in the course
- What type should each of these be?
  - A name can be a string
  - The students in the course can be an array of Student objects.
    - With a max of 35 students in a course.

Create a Class Exercise

- Declare a new class Course in the file Course.java
- Add the fields:
  - private String instructor;
  - private Student[] students = new Student[35];

Create a Class Exercise - Continued

- Add constructors to Course
  - Add the no argument constructor
  - Add a constructor that takes the instructor’s name
- Remember that constructors are declared as
  ```java
  public ClassName(paramList)
  {
  }
  ```
Add Accessors and Modifiers

• Add methods to get and set the fields in the class period

  ```java
  public String getInstructor()
  {
      return this.instructor;
  }

  public void setInstructor(String instructor)
  {
      this.instructor = instructor;
  }
  ```

Override toString

• Add the following method to Course

  ```java
  public String toString()
  {
  }
  ```

• Add code to the method to return
  – a String with the instructor
  – the number of students in the course

• Also for the method:
  – The length of the array isn’t a count of the actual students
  – Find out how many in the array are not null

Summary

• Object-oriented programs
  – Have interacting objects

• Classes define and create objects
  – The contain the fields, constructors, and methods
    • Some methods are called accessor and modifier methods since they give you access to private fields and let you modify private fields

• A class inherits methods from a parent class

Text as Unimedia

• Computers only understand 0 and 1
  – On and off, using magnetism, voltage, etc.

• But we can represent anything with those:
  – Text, Pictures, Sounds, Movies…
Text as Unimedia

- We can do the same with text:
  - Convert a picture to text.
  - Convert a sound to text.
- HTML is a text-based language.
  - It is used to show pictures and play sounds.

Java.lang.String

- Text in Java is stored as a String object
  - In a format called Unicode
    - It uses 2 bytes per character

- A string literal is enclosed in double quotes
  - `String message = "Hi There";`

- How do you add a double quote to a string?
  - For example to express: `She said, "Hi there"`
    - To do this, use `"`
      - For example:
        ```java
        String s = "She said, \"Hi there\"\";
        System.out.println(s);
        She said, "Hi there"
        ```
- Other special characters:
  - `\n` for new line
  - `\t` for tab
Strings are Sequences of Characters

• You can get the character at an index in the string
  – Starting with index 0
    stringObj.charAt(index);
• Try this:
  > String test = "Hello";
  > System.out.println(test.charAt(0));
  > System.out.println(test.charAt(4));
• How would you get the second character?

Working with Delimited Strings

• Sometimes you get information about an object
  – In the form of a delimited string:
    Jane Dorda:88,92,95,87,93,85
    Mike Koziatek:75,92,83,81,91,87
    Sharquita Edwards:91,93,95,92,94,99
• Here the delimiters are
  – a colon after the name, and
  – commas between the grades.

API

• An API is an Application Programming Interface.
• Basically, it lists and describes all of the methods and classes available in that language.
• Open the Java API
  http://docs.oracle.com/javase/7/docs/api/
  – Click on the java.lang package
  – Click on the String class

Parsing a Delimited String

• Add another constructor to the Student class
  – That takes a delimited string of the form –
    Name: grade1, grade2, grade3, grade4, grade5
String Methods

• Look at the methods available for the String class.
  – Which will return part of a string?
  – Which will return the first index of a list of characters in the string?
  – While will return the last index of a list of characters?
  – Which will remove extra space before and after any other characters?
  – Which will return an array of String objects -
    • By chopping the string up into substrings?
    • Based on specified delimiters (like spaces or commas)?

Parsing a Delimited String

• Use the `split()` method to get an array of Strings
  – First based on the colon delimiter
• Use `trim()` to clear off any additional space from the name
  – The name is the first element in the returned array
• Use the `split()` method again to get the array of grades as strings
  – The grades will be the second element of the returned array
  – Use the comma as the delimiter
• Use `Double.parseDouble()` to translate the grade string into a double value
  – For the grade array

Converting to a Number

• Strings are stored in Unicode format
  – Two bytes per character
• Integers are stored in 4 bytes (32 bits)
• You need to convert a number that is represented as a string into the number representation.
• The wrapper classes have methods to do this
  – `Integer.parseInt(numStr)`

The string "1234" is stored in 8 bytes
With each character taking 2 bytes

```
bytes ——— 0|1|0|2|0|3|0|4
```

The integer 1234 is stored in 4 bytes

```
bytes ——— 0|0|4|214
```

Where the 4 is really
4 * 256 = 1024
So,
1024 + 214 = 1234