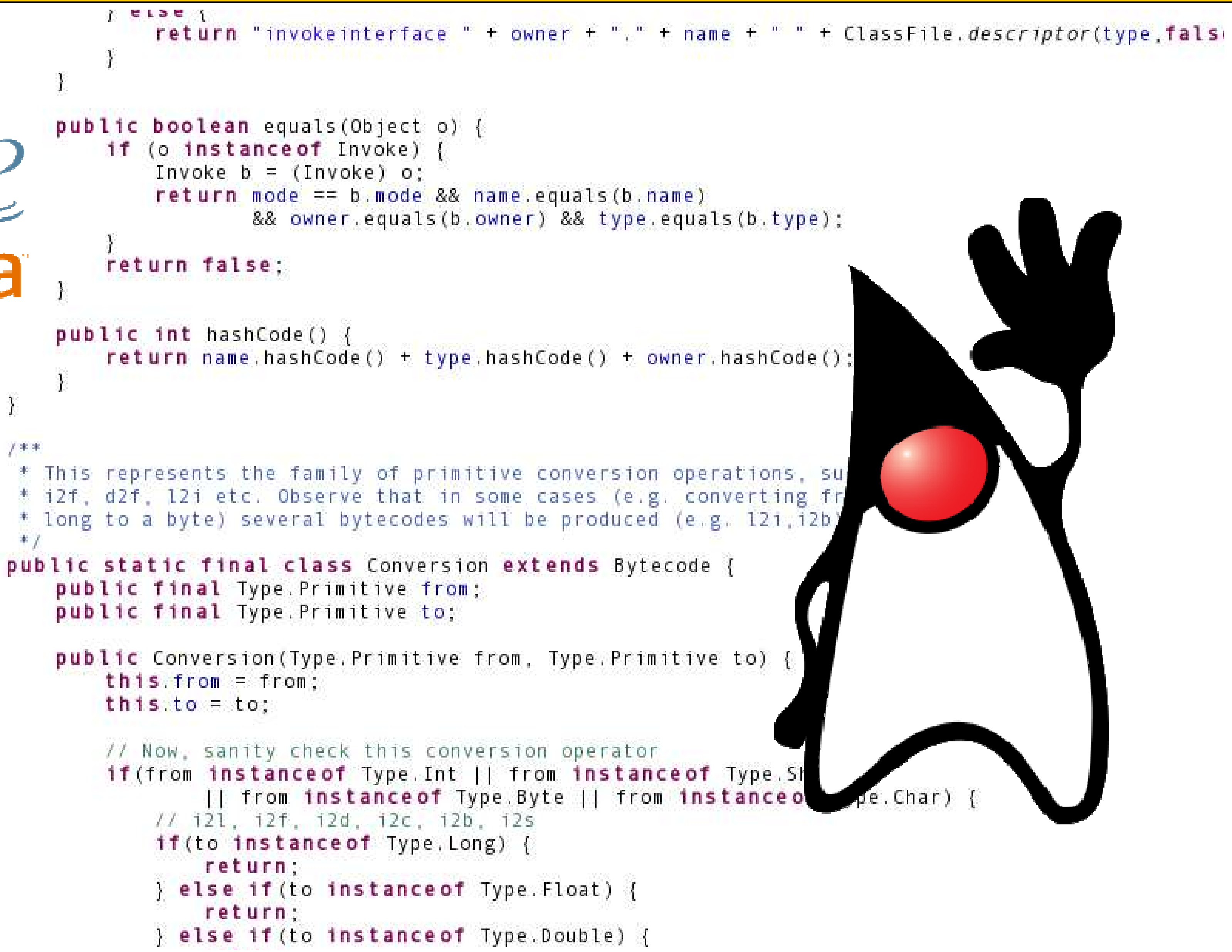
SWEN 221 Software Development

David J. Pearce & Marco Servetto

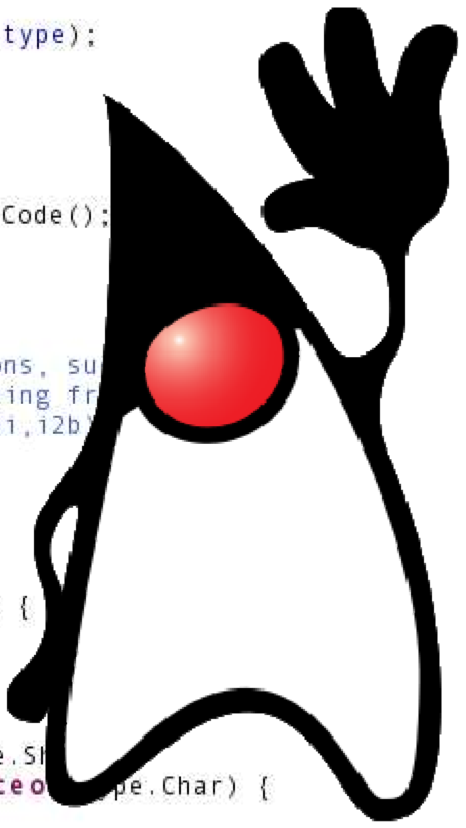
Computer Science, Victoria University

http://www.ecs.vuw.ac.nz/

Welcome !!!



```
        } else {
            return "invokeinterface " + owner + "." + name + " " + ClassFile.descriptor(type,fals
        }
    }

    public boolean equals(Object o) {
        if (o instanceof Invoke) {
            Invoke b = (Invoke) o;
            return mode == b.mode && name.equals(b.name)
                && owner.equals(b.owner) && type.equals(b.type);
        }
        return false;
    }

    public int hashCode() {
        return name.hashCode() + type.hashCode() + owner.hashCode();
    }
}

/**
 * This represents the family of primitive conversion operations, su
 * i2f, d2f, l2i etc. Observe that in some cases (e.g. converting fr
 * long to a byte) several bytecodes will be produced (e.g. l2i,i2b)
 */
public static final class Conversion extends Bytecode {
    public final Type.Primitive from;
    public final Type.Primitive to;

    public Conversion(Type.Primitive from, Type.Primitive to) {
        this.from = from;
        this.to = to;

        // Now, sanity check this conversion operator
        if(from instanceof Type.Int || from instanceof Type.Sh
                || from instanceof Type.Byte || from instanceo    pe.Char) {
            // i2l, i2f, i2d, i2c, i2b, i2s
            if(to instanceof Type.Long) {
                return;
            } else if(to instanceof Type.Float) {
                return;
            } else if(to instanceof Type.Double) {
```



What is this course about?

Generics Polymorphism

Exceptions Inheritance

Programming Exceptions

Java Programming Polymorphism

Programming

Inheritance Java Inheritance

Generics Polymorphism Inheritance Generics

Programming Programming Programming Inheritance

Java Java

Polymorphism Inheritance Exceptions

Exceptions Exceptions Java

Java Exceptions

Inheritance Polymorphism

Generics

Polymorphism Generics

People

- **Dr. David Pearce** (course coordinator)
  - **Office**: CO 231
  - **Tel**: 463 5833
  - **Office Hours**: Tue, Wed 3-5pm
  - SWEN Programme Director

- **Dr. Marco Servetto** (lecturer)**:**
  - **Office**: CO 258
  - **Tel**: 463 5820
  - **Office Hours**: Tue, Wed 3-5pm

Elect a Class Rep

- Take the first step as a Student Representative at Victoria and become a Class Rep.

- Class Reps are expected to work with the lecturer and the class to support and improve students’ learning experiences in your course and at Victoria.

- You will be trained, prepared and supported in your role.

- Representing your class goes towards the VicPlus Award and can lead to other representation opportunities.

TE WHARE WĀNANGA O TE ŪPOKO O TE IKA A MĀUI
VICTORIA
UNIVERSITY OF WELLINGTON

VUWSA
YOUR STUDENTS' ASSOCIATION

Course Announcements

- Lectures – come along!

- E-mail to your ECS account
  - Read it daily!
  - Forward if you want

- Course website
  - http://www.ecs.vuw.ac.nz/Courses/SWEN221_2016T1

Textbook

There is no official course textbook. However, you should find the following textbook useful

- *Java Foundations: Introduction to Program Design and Data Structures*, by Lewis, DePasquale, and Chase.

Other useful text books include the following:

- *Program Development in Java*, Barbara Liskov
- *Object-Oriented Design & Patterns*, Cay Horstmann, 2nd Ed.
- *Practical Object Oriented Design*, Bhuvan Unhelkar
- *Effective Java*, Josh Bloch

There are a number of other useful books on programming in Java available in the library!

Lectures + Labs

- Lectures
  - **Tuesdays (MCLT101)** and Wednesdays **(HULT323)**, 2-3pm
  - **Fridays** (**MCLT101**) will be used occasionally for **tutorials**
  - Lecture notes – *maybe incomplete*
- Labs
  - **Wed**: 9-11am,11-1pm,3-5pm;
  - **Thur**: 9-11am, 1-3pm,3-5pm;
  - **Fri**: 9-11am,3-5pm
  - **Rooms:** CO242 and CO243
  - **Must attend 8 / 10 weekly labs**
  - Need to sign up for a lab stream this week! (**opens 4pm today**)
- Lab Marking
  - Labs will be marked out of 10 either by the **tutor**, or by the **automated marking system**
  - **Each lab worth:** 1.5%
- **Fair Warning:** First Lab will include a short “diagnostic” test …

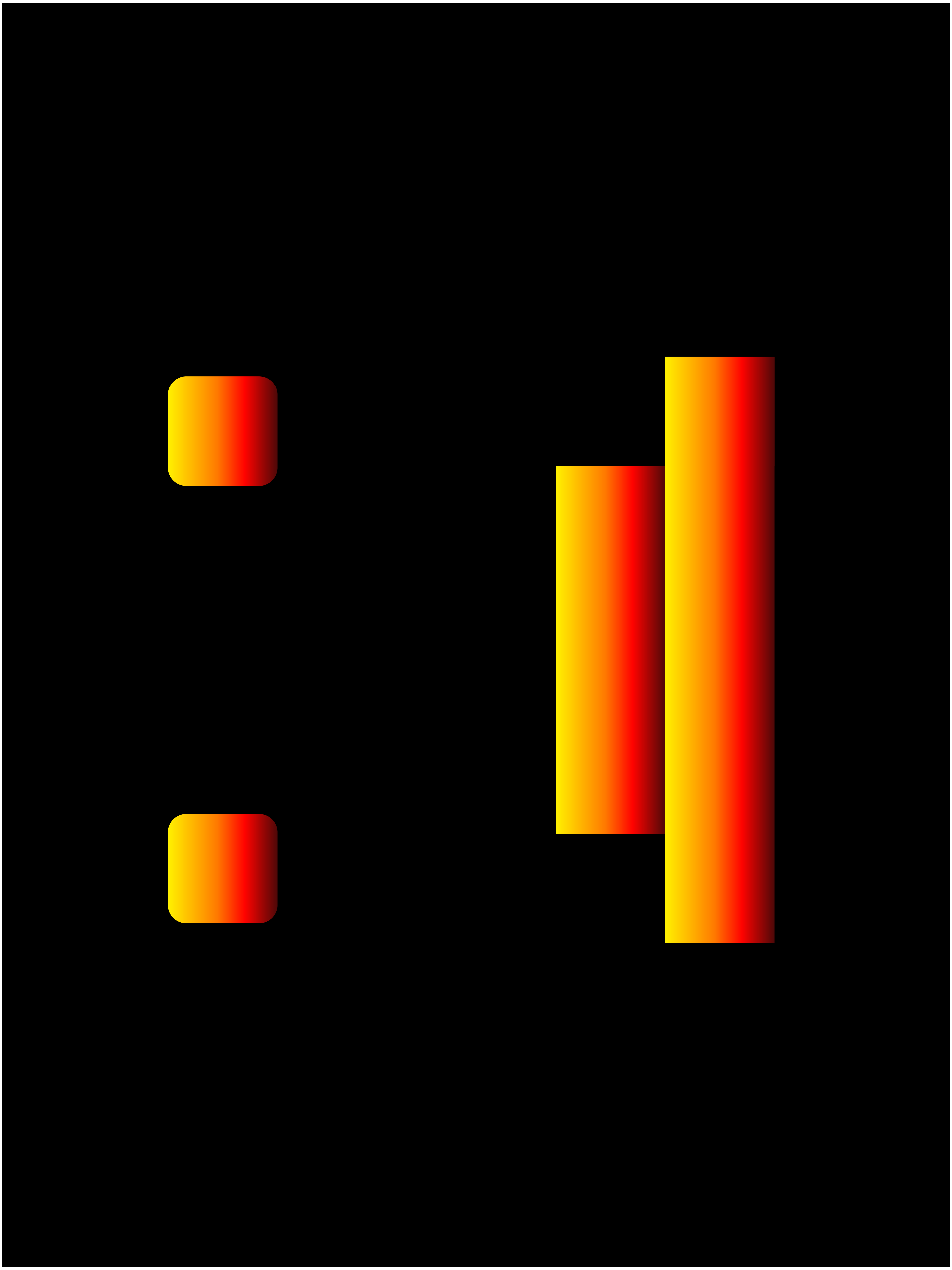
Assignments

- Six Assignments
  - Assignments 1-5 are **two weeks** long, whilst Assignment 6 is **one week**
  - Each assignment worth 3%

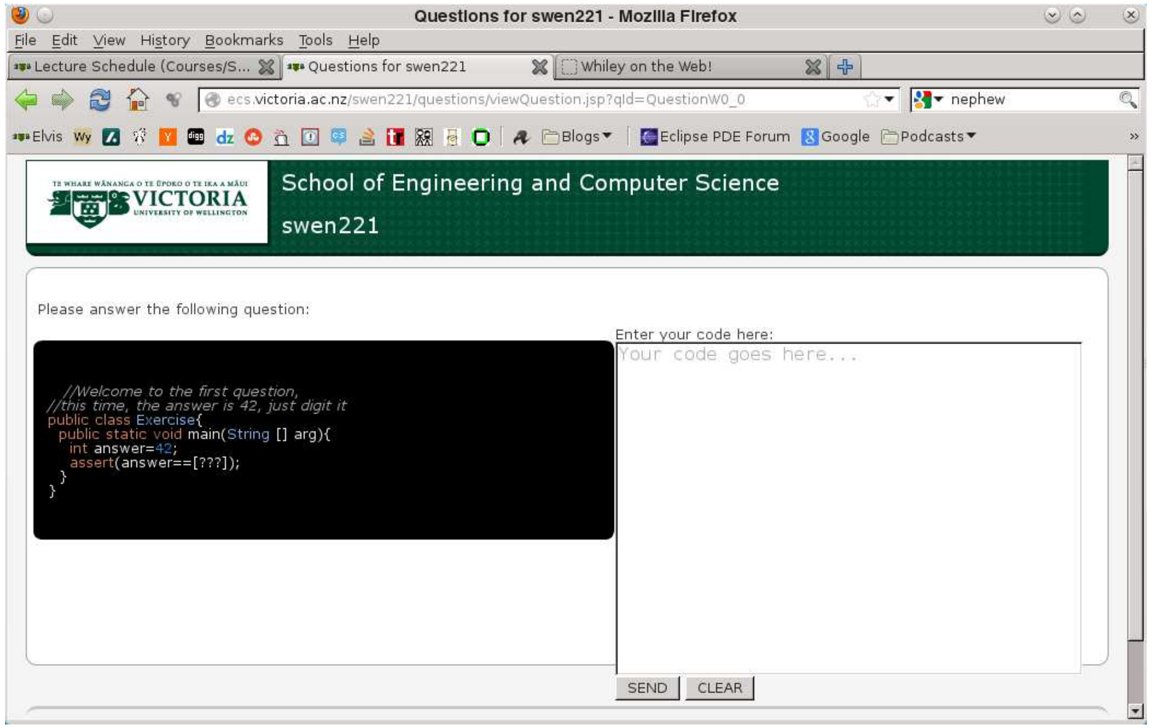
- Automatic Marking Script
  - Runs submission again a batch of test inputs
  - Generates **some or all of your marks**
  - Emails you with submission problems, score and failing inputs
  - **You must follow assignment specification regarding output + submission**

- Marking Criteria
  - **Correctness** – does the code adhere to the given specification?
  - **Style** – code follows style guide, has appropriate comments (inc. Javadoc)
  - **Late penalty** - 20% per day, max 5 days!

Plan > 10 hours per week



Self-Assessment

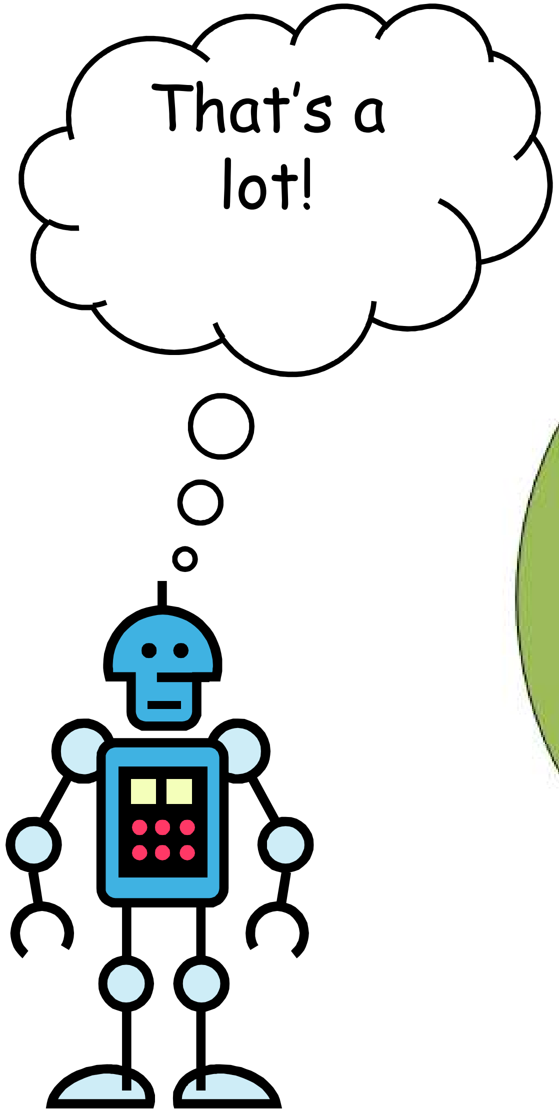

Questions for swen221 - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Lecture Schedule (Courses/S...
Questions for swen221
Whiley on the Web!
ecs.victoria.ac.nz/swen221/questions/viewQuestion.jsp?qId=QuestionW0_0
nephew
Elvis
Blogs
Eclipse PDE Forum
Google
Podcasts
VICTORIA UNIVERSITY OF WELLINGTON
School of Engineering and Computer Science
swen221
Enter your code here:
Your code goes here...
SEND
CLEAR

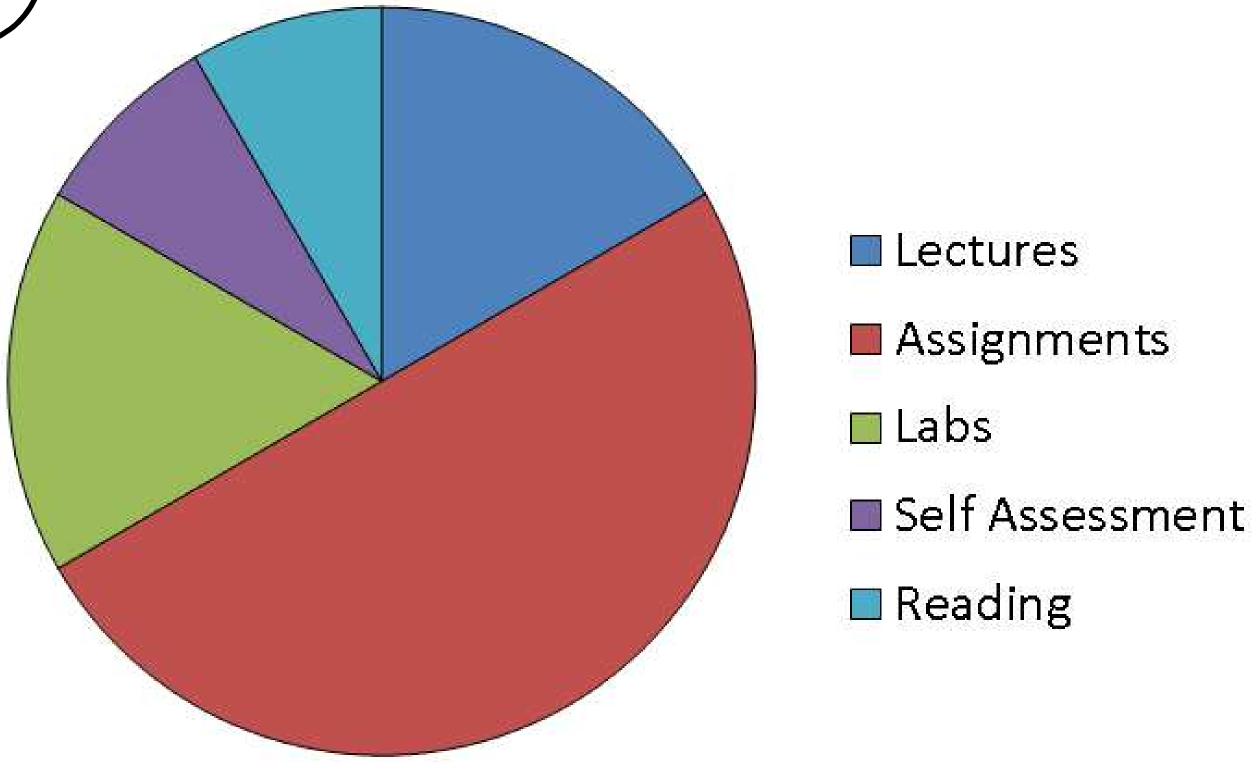
Please answer the following question:

```
  //Welcome to the first question,
//this time, the answer is 42, just digit it
public class Exercise{
  public static void main(String [] arg){
    int answer=42;
    assert(answer==[???]);
  }
}
```

- Online **Self-Assessment** Tool
  - Currently lists **~100** questions which test **knowledge** of Java
  - Questions due at the **end of each week**, and each worth **one mark**
  - To get the mark, need to complete question **before due date**
  - **You are encouraged to search the web for answers!!**

Workload


That's a
lot!


🟦 Lectures
🟥 Assignments
🟩 Labs
🟪 Self Assessment
🟦 Reading

Assessment

- Assignments 1-6 @ 3% each = **18%**
- Self-Assessment @ **7%**
- Labs @ ~1.5% each = **15%**
- Terms Test @ **10%**
- Final Exam @ **50%**

- **Mandatory Requirements**
  - Reasonable attempt on at least 5 / 6 assignments
  - Must attend at least 8 / 10 weekly labs
  - Must complete 75% of self-assessment questions
  - **D** grade on final exam

  - (can catch up if you miss a lab or assignment)

Expectations

- What’s reasonable to think?
  - *“If I attend the lectures, try most of the labs and assignments, then I’ll pass”*
  - *“I might have to do some extra reading on the internet if I get stuck, or ask a tutor/lecturer for help”*
  - *“The tutors/lecturers know everything and they never make mistakes”*
  - *“I can pass the course if I copy the assignments from my friend. He/She’s a Java Expert!”*
  - *“As soon as I get stuck, I ask my friend for help. He/She’s great!”*

Rules And Policies

- Read **course outline** on the SWEN221 webpage

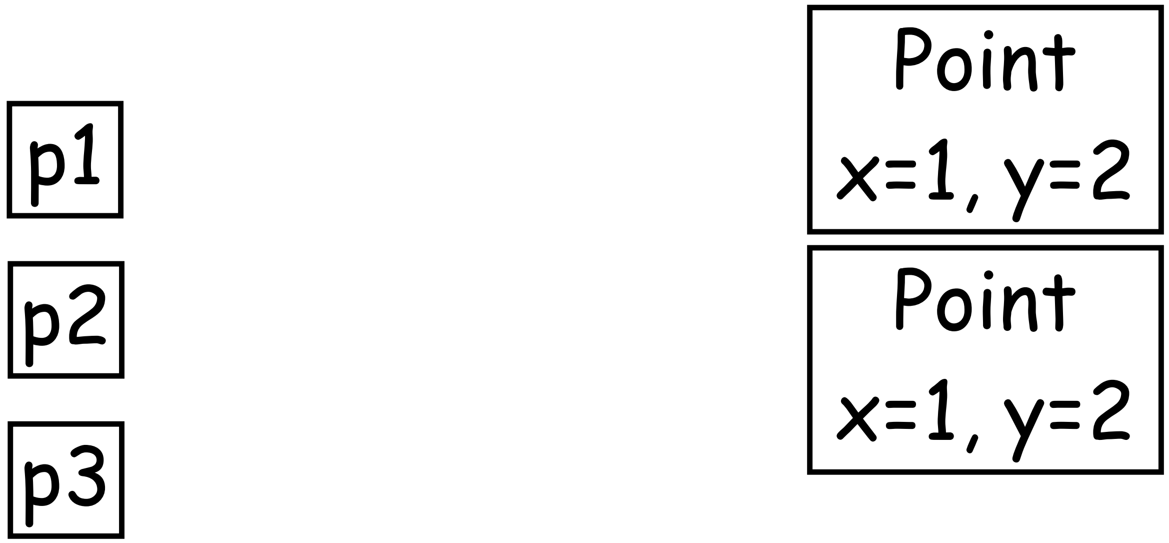
- Standard Policies
  - Academic Integrity — *do your own work*
  - Group Work — *shared responsibility*
  - Student Support
  - Student & Staff Conduct

Equality Quiz – what gets printed?

```
class Point {
 int x; int y;
 public Point(int x, int y) {
  this.x = x; this.y = y;
}}

Point p1 = new Point(1,2);
Point p2 = p1;
Point p3 = new Point(1,2);
p2.x = 2;
if(p1 == p2) { System.out.println(“p1==p2!”); }
if(p1 == p3) { System.out.println(“p1==p3!”); }
```

A) “p1==p2!” B) Nothing C) “p1==p2!” “p1==p3!”

Why?



- The '==' operator
  - Checks whether two objects are *same object*
  - Not just whether they have *same value*
    - Must override Object.equals() for value identity

Quiz: What gets printed?

```
class Point {
 int x = 0;
 int y = 0;
 static int z = 0;
 Point() { z = z + 1; }
}

Point p1 = new Point();
Point p2 = new Point();
System.out.println(“x = ” + p2.x);
System.out.println(“y = ” + p2.y);
System.out.println(“z = ” + p2.z);
```

- A) “x = 0” B) “x = 1” C) “x = 0”
- “y = 0” “y = 0” “y = 0”
- “z = 0” “z = 1” “z = 2”

Engineering

“Any intelligent fool can make things bigger, more complex, and more violent. It takes a touch of genius -- and a lot of courage -- to move in the opposite direction.”

-- Albert Einstein

“A charlatan makes obscure what is clear; a thinker makes clear what is obscure.”

-- Hugh Kingsmill