CT2106
Object Oriented Programming

Dr. Frank Glavin
Room 404, IT Building
Frank.Glavin@University*of*Galway.ie
School of Computer Science

University *of* Galway.ie

Contact Information

Lecturer: Dr Frank Glavin

Frank.Glavin@nuigalway.ie

Office : Room 404, Information Technology Building

Note:

The bulk of this course content was originally developed by

Dr Conor Hayes

Ollscoil na Gaillimhe
University of Galway

Lecture/Lab Times and Location

Lecture - Thursday 9 am – 10 am:
AC003, D'Arcy Thompson Lecture Theatre

Lecture - Friday: 10 am – 11 am:
IT250, Information Technology Building

Lab – Tuesday 11 pm – 1 pm:
BLE2012 Comp Suite
Arts Sci Rm 105
Block E, Ground Flr, E102

Lab – Friday 3pm – 5pm
IT106 [4BSE1 and 4BSE4]

Ollscoil na Gaillimhe
University of Galway

Learning Materials

- Lecture content will be provided in advance
- Lectures themselves will be in tutorial format
- You will need to bring a laptop to each lecture
- Weekly lab sessions

Ollscoil na Gaillimhe
University of Galway

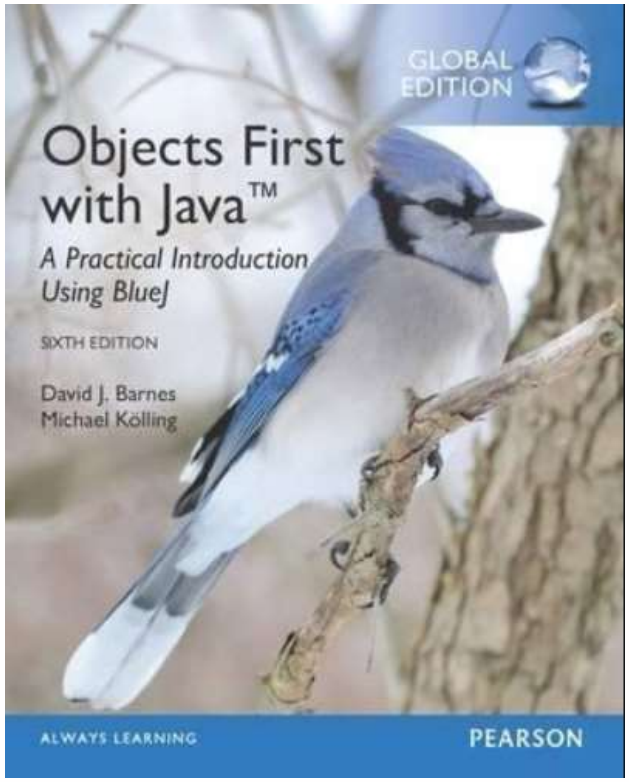
Attendance

- Attendance at each lecture/tutorial will be recorded
- Attendance will be captured using the Qwickly app
- You will have time during the lecture to enter the pin

Ollscoil na Gaillimhe
University of Galway

Recommended Reading

Objects First with Java:
A Practical Introduction using BlueJ
David J. Barnes,
Michael Kölling



Ollscoil na Gaillimhe
University of Galway

Other Reading Texts

- Think Java by Allen B. Downey
  http://www.greenteapress.com/thinkapjava/
- Thinking in Java by Bruce Eckel
  http://www.mindview.net/Books/TIJ/
- The Java Tutorials hosted by Oracle
  http://docs.oracle.com/javase/tutorial/index.html

Java, A Beginner's Guide, 5th Edition by Herbert Schildt
Effective Java (2nd Edition) by Joshua Bloch
Head First Java by Kathy Sierra, Bert Bates

Ollscoil na Gaillimhe
University of Galway

Useful Online Resources

- https://www.geeksforgeeks.org/java/
- https://www.w3schools.com/java/default.asp
- https://www.w3schools.com/java/exercise.asp?filename=exercise_syntax1
- https://www.tutorialspoint.com/java/index.htm
- https://www.tutorialspoint.com/java/java_online_quiz.htm

Ollscoil na Gaillimhe
University of Galway

Extra Support

ComputerDISC is a Computer Programming Drop-In Support Centre for all NUI Galway students who are taking any programming/software development courses. The DISC is a free service that supports all students with their self-directed learning in computing topics at all years and levels in NUI Galway.

Room 205 in the Information Technology Building
https://www.universityofgalway.ie/science-engineering/school-of-computer-science/currentstudents/computerdisc/
https://www.universityofgalway.ie/science-engineering/school-of-computer-science/currentstudents/computerdisc/timetable/

Ollscoil na Gaillimhe
University of Galway

Module Assessment

Assessment:

- There will be between 3 and 5 lab assignments
- Computer-based programming exam at the end of semester
- Attendance/participation at the weekly lecture tutorials
- If you should have to repeat in Autumn, your overall result is **capped** at 40%

Ollscoil na Gaillimhe
University of Galway

Computer Based Exam

- In December, you will have a two-hour computer-based exam
- You will be required to solve two/three problems by programming in Java
- You will not be able to pass this exam without having developed programming competence
- Like riding a bicycle, this is not something you can learn from a book.
- You should be programming for at least two hours every week

Ollscoil na Gaillimhe
University of Galway

Learning Objectives 1

Just a pass	• Define the basic principles of OOP • List a subset of best programming practices • List the differences between OOP and procedural programming • Name the basic Java data types and demonstrate how to use these as variables • Write and compile a basic OOP program based on a given set of instructions using an IDE such as Eclipse
Quite Satisfactory	• Create and Implement a subset of stub classes and methods so that an initial overall approach compiles • Recognise when inheritance can be used to reduce code redundancy. • Apply basic inheritance approach to solve redundancy • Implement basic software engineering best practices - such as use of methods to reduce redundancy, appropriate use of access modifiers, encapsulation • Demonstrate appropriate use of instance vs static methods/variables • Demonstrate appropriate use of getter/setter methods

Ollscoil na Gaillimhe
University of Galway

Learning Objectives 2

Highly satisfactory	• Distinguish when inheritance or an interface approach is most appropriate • Demonstrate the appropriate use of polymorphism in a coding solution • Distinguish between data structures (Arrays, ArrayLists, HashMaps, Stacks) • Recognise when to use key utility libraries in the Java language (e.g java.utils. Collections) and demonstrate how to implement them
The very best understanding	• Explain the modelling rationale for using a set of classes and methods to solve a problem description • Formulate, design and implement and test a full OO solution to a problem description • Independently recognise and apply a design pattern to solve a coding problem • Employ creative and original thinking in formulating the solution • Demonstrate a test-driven (unit-testing) approach to solving a coding problem • Assess and Compare one solution approach against another

Ollscoil na Gaillimhe
University of Galway

Topics

- Classes, objects, methods
- Primitive and reference types
- Object interactions
- Arrays and collections and how to iterate
- Modelling decisions - what classes, relationships and methods to design
- Inheritance: using it to improve structure
- Polymorphism: how to use to implement the open-close principle
- Object interactions again: composition
- Java libraries
- Using Interfaces
- Good programming practice: unit testing and exception handling
- Using a design pattern to solve an OOP problem

Ollscoil na Gaillimhe
University of Galway

Learning Objectives: Week 1

You should be able to:

- Describe what an Object Oriented Programming language is
- Differentiate between a class and an object
- Create a simple **class** in BlueJ and create several **objects** of that class
- Create some simple **methods** in Java

Ollscoil na Gaillimhe
University of Galway

What is an Object-Oriented Programming language?

“Hello World”

```c
#include <stdio.h>

int main() {
   /* my first program in C */

   char hello[] = "Hello, World! \n";

   printf(hello);

   return 0;
}
```

```java
public class Greeting
{

    public Greeting()
    {
        System.out.println("Hello World");
    }

    public static void main(String[] args)
    {
        new Greeting();
    }
}
```

C

Java

What are the similarities and differences between the two code snippets?

Ollscoil na Gaillimhe
University of Galway

Information on public static void main...
https://www.journaldev.com/12552/public-static-void-main-string-args-java-main-method

Definitions:

- Class
  - Something from which you create objects.
    - Template

- Object
  - A Java object is a self-contained component which consists of methods and properties
  - E.g. in an ecommerce program, we could have customer object, item object, or book object (it will have name, ID, Price etc.)

Ollscoil na Gaillimhe
University of Galway

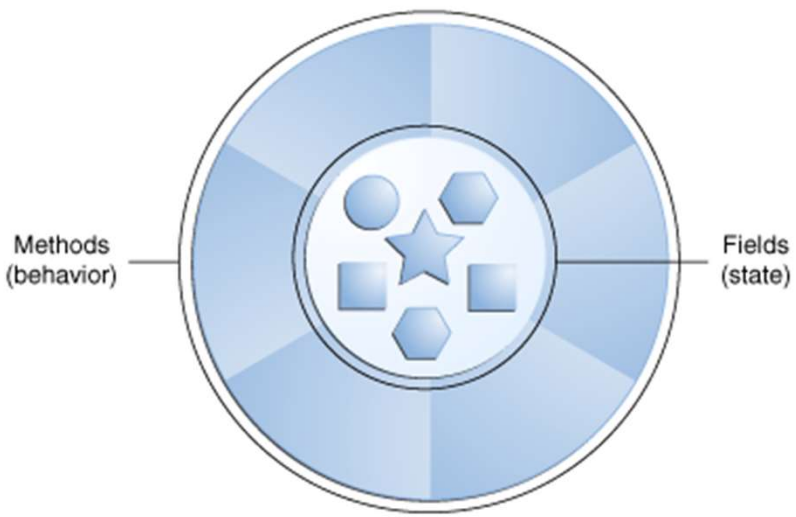
What is a class?

- A class is a type of **blueprin**t or **template** from which you make objects
- The use of classes and objects are the principal differences between programming in C and programming in Java.
- However, it entails a fundamentally different way of designing your code

Ollscoil na Gaillimhe
University of Galway

What is an object?

- A piece of programming code that has a **state** and has **behaviour**
- Often it represent a real thing
- It is created in code by *instantiating* a class



Ollscoil na Gaillimhe
University of Galway

Bytecode

Unlike other high-level programming languages, Java code is **not** compiled into machine specific code that can be executed by a microprocessor.
Instead, Java programs are compiled into something called **bytecode**. The bytecode is input to a Java Virtual Machine (JVM), which interprets and executes the code. The JVM is usually a program itself.
The bytecode is **platform independent**. So, the JVM is specific for each platform, but the bytecode for the program remains the same across different platforms. This is a very nice feature of Java.
Of course there is always a trade off….
The main trade off is the effect it has on the execution speed.

Ollscoil na Gaillimhe
University of Galway

Creating your first class

- Lets write a simple program in BlueJ

- In the lecture, you are going to
  - Create your first class
  - Create several objects of this class

Ollscoil na Gaillimhe
University of Galway