# Macintosh HD:Users:deepa:Downloads:sps_long_color.jpg

# **Intermediate Java**

# **General Course Information**

**Assistant Professor:** Sam Sultan

**Email:** [sam.sultan@nyu.edu](mailto:sam.sultan@nyu.edu)

**Course title/number:**  Introduction to Java – INFO1-CE9238

**CE Units:** 3 units

**Semester/Year**: Fall 2018

**Class meeting/location:** Sunday at 9:00am-5:00pm - November 4 -December 9

**Office Hours** By Appointment

# **Course Description:**

Advance your knowledge with object-oriented programming in Java. Topics covered include abstraction, encapsulation, inheritance, polymorphism, interfaces, packages, garbage collection, exception handling, input/output, multithreaded programming, networking, utility classes, AWT, and Swing classes, Java architecture, security, JDBC, and other features. Develop your aptitude and confidence when working with more advanced topics in Java and object-oriented principles, and gain a heightened comprehension of the more complex Java libraries for advanced programming.

# **Course Structure/Method:**

This course will be delivered on Monday evenings in person. The class will encompass lectures, assignments, examples and demos, midterm and final exams. All class content and assignments will be made available online via the web at <http://oit2.sps.nyu.edu/~sultans/java>1

# **Course Learning Outcomes:**

By the end of this course, students will be able to:

* Create custom Objects and Classes
* Use abstraction, encapsulation, inheritance and polymorphism
* Create method overriding and method overloading
* Build a robust exception handling processes
* Use input/output streams, readers and writers
* Understand various data structures and create collection classes and generics
* Access a database through JDBC drivers
* Build AWT and Swing window components
* Understand Java events, and event handling
* Create multi-threading Java programs
* Perform networking and communication using TCP/IP

# **Communication Policy:**

Credit students must use their NYU email to communicate. Non-degree students do not have NYU email addresses. NYU Classes course-mail supports student privacy and FERPA guidelines. All email inquiries will be responded to within 24 hours during Monday through Friday 5pm. Email sent on Saturday or Sunday will not be responded to until Monday. I will respond to you using NYU email.

# **Course Expectation:**

Students are expected to participate in each class session by offering their understanding of the subject, sharing ideas or discussing/commenting on another students comment. In addition, students must complete and submit all assigned homework on time. Late submission of homework will either not be accepted, or will result in a lower grade. Students are also expected to develop with and present a team project with other students, as well as take and pass a midterm exam and a final exam.

See full detail of expectations under “Assessment Strategy” below. Further information about specific assignments can also be found in the “Course Outline” section.

**Attendance:** Students are expected to attend all classes. Excused absences are granted in cases of documented serious illness, family emergency, religious observance, or civic obligation. In the case of religious observance or civic obligation, this should be reported in advance. Unexcused absences from sessions may have a negative impact on a student’s final grade. Students are responsible for assignments given during any absence. Each unexcused absence may result in a student’s grade being lowered by a letter grade. A student who has three unexcused absences may earn a Fail grade

University Calendar Policy on Religious Holidays <https://www.nyu.edu/about/policies-guidelines-compliance/policies-and-guidelines/university-calendar-policy-on-religious-holidays.html>

**Class Participation:** To receive full credit for the course, you should attend all classes since much of the learning occurs during class lecture, presentation and class discussions. Please contact the instructor if you anticipate missing any part of the class. Please arrive on time so as not to disturb the flow of the lecture. Excessive lateness’s may result in lower overall grade.

**Required & Recommended Material:**

* Beginning Java – 7th Edition
* **Authors** – Ivor Horton
* **Publisher** – Wrox Press Inc.
* Instructor will also provide session by session content available online at class web site
* <http://oit2.scps.nyu.edu/~sultans/java>

**Additional Recommended Material**

* Just Java 2 – 6th Edition
* **Authors** – Peter Van Der Linden
* **Publisher** – Sun Microsystems Press.
* Thinking in Java – 4th Edition
* **Authors** – Bruce Eckel
* **Publisher** – Prentice Hall.

**Assessment Strategy:**

Contributing factors for determining your course grade include:

* Attendance Class Participation - **10%** *(Attendance is prerequisite to participation)*
* Homeworks - **20%**
* Student Object Homework - **10%**
* Address Book Object Homework - **10%**
* Final Exam - **50%**
* Total 100%
* **Class Participation:** To receive full credit for the course, you should attend all classes since much of the learning occurs during class lecture, presentation and class discussions. Please contact the instructor if you anticipate missing any part of the class. Participation grades will be based on:
* Involvement in class discussions and activities
* Participation which demonstrates integration of reading, class work, relevance and application.
* Willingness to learn by accepting feedback, trying new skills and approaches, etc.
* Quality/quantity of providing effective and balanced feedback.
* **Homework:** Homework assignments must be submitted on time within 1 week of date assigned (unless otherwise instructed). Late submission will severely impact your homework grade, or may not be accepted altogether at instructor’s discretion. Late submission will incur 10 points penalty for each session we meet and not submitted. All homework pages must be stapled together (paper clips not accepted).
* **Student Class Project/Homework:** There will be a Java “Student Class” project/homework. The project will be a culmination of all concepts learned in this class, and will test your knowledge and understanding of Object Oriented programming concepts.
* **Final Exam:** There will be a final exam. The exam will be an open book, open notes/internet style exam. The exam will test the student's acquisition of topics, concepts and competencies learned in this class. The final exam will only cover all materials covered throughout the course.

**NYU SPS Policies:**

“NYUSPS policies regarding the Family Educational Rights and Privacy Act (FERPA), Academic Integrity and Plagiarism, Students with Disabilities Statement, and Standards of Classroom Behavior among others can be found on the NYU Classes Academic Policies tab for all course sites as well as on the University and NYUSPS websites.   
Every student is responsible for reading, understanding, and complying with all of these policies.”

The full list of policies can be found at the web links below:

* University: <http://www.nyu.edu/about/policies-guidelines-compliance.html>
* NYUSPS: <http://sps.nyu.edu/academics/academic-policies-and-procedures.html>

**School Grading Policies:**

**NYUSPS Career Advancement (non-degree)**

http://sps.nyu.edu/content/scps/academics/noncredit-offerings/academic-noncredit-policies-and-procedures.html

### 

### **Course Outline:**

**Session 1, Week 1, Object Oriented Concepts and Programming**

* Object Oriented Concepts
* Defining Classes
* Defining Fields and Methods in a class
* Instance vs. Static fields and methods
* Instantiating Objects
* Defining Constructors
* Data Encapsulation
* Defining getters and setters
* Method Overloading
* The 'this' variable
* Creating objects from within the same class
* Creating and Importing Packages
* The CLASSPATH
* Access Control to Class Members

**Reading:** Chapter 5

**Session 2, Week 1, Extending Classes and Inheritance**

* Class Inheritance
* Abstract Classes
* Extending Classes
* What is a Superclass, what is a Subclass
* The Universal Superclass
* The toString( ) Method
* Determining the type of an Object
* Inheritance and Polymorphism

**Reading:** Chapter 6

**Session 3, Week 2, Java Exceptions**

* What are Exceptions
* Type of Exceptions
* Handling Exceptions
* The try/catch Block
* Define and Throw your own Exceptions
* Extend a generic Exception
* Creating an Exception Package

**Reading:** Chapter 7

**Session 4, Week 2, File Input and Output**

* Input and Output Streams
* Using Readers, using Writers
* Working with the File Object
* Testing for Files and Directories
* File Input and Output
* Dealing with Buffers
* Reading from Files, writing to Files

**Reading:** Chapters 8-11

**Session 5, Week 3, Data Structures & Collection Classes**

* Data Structures
* Collection Classes
* Differences between Arrays & Collection classes
* Types of Data Structures
* Java <Generics>
* Lists: ArrayList, Vector, LinkedList
* Stacks and Queues
* Sets: HashSet, TreeSet
* Maps (key/value pairs): HashMap, TreeMap
* Iterators

**Reading:** Chapters 13-14

**Session 6, Week 3, Using Databases**

* Using Databases with Java
* The SQL Language
* Reading data from databases
* Inserting, Updating and Deleting Data
* Creating database tables and indexes
* Connecting Java to a database using JDBC
* Working with a ResultSet

**Reading:** Chapters 24-25

**Session 7, Week 4, Creating Windows**

* Creating windows
* AWT and Swing components
* Menus, text, buttons & other components
* Using containers
* The Flow Layout Manager
* Adding menus to windows
* Event Listeners
* Registering event listeners/interfaces
* Event handlers

**Reading:** Chapters 17-19

**Session 8, Week 4, Using Multi-Threading**

* Understanding Java Threads
* Multi-Threading vs. Multi-Processing
* Lifecycle and States of a Java Thread
* The Thread Class & the Runnable Interface
* Creating Threads
* User vs. Daemon Threads
* Interrupting/Terminating Threads
* Making Threads Safe with Synchronization
* Communicating Between Threads
* Preventing Threads Deadlocks

**Reading:** Chapter 16

**Session 9, Week 10, Java Networking**

* What is a Networking?
* The Internet IP Address
* Domain Name Server (DNS)
* Port Numbers - standard, non-standard
* Network Sockets
* The Java socket Classes
* The InetAddress Class
* TCP/IP Communication Protocol
* The ServerSocket and Socket Class
* UDP/IP Communication Protocol
* The DatagramPacket and DatagramSocket Class
* Building Client/Server Applications

**Reading:** Chapter 15 (Thinking in Java)

**Session 10, Week 10, Servlets, JSP and Web Services**

Java Servlets, JSP and Web Services

Dynamic web output

Tomcat and other JSP/Servlet Containers

Java Server Pages

The HTTP protocol

Java Request/Response Objects

The GET and POST methods

Java Web Services

Client Side Request

Server Side Response

**Reading:** None

* **Final Exam**