

**1. Course Code**

2291

**2. Course Title**

Data Structures and Object Oriented Programming (Java)

**3. Teacher**

WANNOUS, Muhammad

**4. Term**

Fall 2

**5. Course Requirements (Courses / Knowledge prerequisite for this course)**

Courses:

[1] Fundamentals of Computer Programming (2209)

Skills:

[1] Use of the command-line and IDE

[2] Software installation and setup

**6. Course Overview and Objectives**

This course introduces the students to the Object-Oriented (OO) paradigm and the data structures used in it. The course utilizes the popular programming language Java to introduce the concepts of programming in OO and the implementations of different data structures in it, such as the arrays, lists, queues, and trees. The course serves as a base for any developer who intends to use Java in developing web, mobile, and desktop applications.

**7. Course Outline**

- 1 Course orientation and development environment tools and setup.
- 2 Introduction to the command-line environment, command-line arguments, and the standard I/O.
- 3 How applications run. My first Java application: coding, compiling, and running in the command-line.
- 4 The object-oriented paradigm. Variable types and methods. (exercise-1)
- 5 Objects in Java (classes) and packages. Instantiating and using objects.
- 6 Static variables and methods. Packaging of Java applications (.class and JAR files)
- 7 Expressions, operators, and statements. Control flow statements.
- 8 I/O operations. The class constructor, getters, and setters. (exercise-2)
- 9 Method overloading (polymorphism). Java documentation. String and Math classes.
- 10 Inheritance in Java. Exception handling. (exercise-3)
- 11 Data structures: arrays (1-dimension and n-dimension)
- 12 Data structures: lists (ArrayList and LinkedList)
- 13 Data structures: Maps (HashMap)
- 14 Sample application development-1
- 15 Final project
- 16 -

## 8. Textbooks (Required Books for this course)

For this course, a set of lecture slides/pages, handouts, and other resources will be distributed in a timely manner through Moodle.

## 9. Reference Books (optional books for further study)

[1] Core Java Volume I--Fundamentals, ISBN-13: 978-0134177304

[2] Data Structures and Algorithms in Java, ISBN-13: 978-0672324536

## 10. Course Goals (Attainment Targets)

- (1) Describe the Object-Oriented programming paradigm.
- (2) Explain the differences between the various data structures.
- (3) Use the development environment tools to write, compile, run, and test simple object-oriented Java programs.
- (4) Implement a project in the java language and test its functions
- (5)
- (6)
- (7)
- (8)

## 11. Correspondence relationship between Educational goals and Course goals

Educational goals of the school			Course Goals
High level ICT skills	Basic academic skills		(1) (2)
	Specialized knowledge and literacy		(3) (4)
Human skill (Tankyu skill)	Ability to continually improve own strengths		
	Ability to discover and resolve the problem in society	Problem setting	
		Hypothesis planning	
		Hypothesis testing	
		Practice	
	Fundamental Competencies for Working Persons	Ability to step forward	
		Ability to think through	
		Ability to work in a team	
Professional ethics			

## 12. Evaluation

Goals	Evaluation method & point allocation					
	examination	Quiz	Reports	Presentation	Deliverables	Other
(1)		○				
(2)		○				
(3)					○	
(4)					○	
(5)						
(6)						
Allocation		40			60	

## 13. Evaluation Criteria

Examination	
-------------	--

Quiz	This course will include up to six quizzes. Each quiz consists of several questions of different types, simple answer, multiple choices, space-filling, ...etc. The questions are to verify the knowledge a student has acquired regarding specific lectures. The quiz is limited in time, but a student can try it twice.
Reports	
Presentation	
Deliverables	The course has three exercises and two projects. Each exercise involves writing an application in the Java language to perform a specific function or a small set of features. The projects are relatively more significant than the exercise and involve writing applications that perform several functions. The instructor will demonstrate the development of the first project while students work, individually, on the final one. Each student should deliver the working code for the exercises and the final project, and the instructor will verify it on his computer.
Other	

#### 14. Active Learning

Hourly percentage of active learning within the whole class time		70%
1	Active learning such as problem solving assignment using the knowledge and skills acquired in class.	All the time
2	Active learning such as group works and discussions.	Sometimes
3	Outcome presentations and feedbacks.	Not at all
4	Students actively make decisions on how the class should be conducted.	Not at all

#### 15. Notes

This course contains both theoretical and practical parts. Mastering the command-line and the use of NetBeans IDE is necessary to complete the exercises. Quizzes and exercises have deadlines and they won't be postponed unless a serious issue occurs.

## 16. Course plan

(Notice) This plan is tentative and might be changed at the time of delivery

---

### Lesson 1: (Course orientation and development environment)

(Discussion, Lecture 45 minutes, Practice 45 minutes)

---

- [1] Course syllabus
- [2] Grading
- [3] Development environment
  - + Java Development Kit (JDK)
  - + NetBeans IDE
  - + GitHub

---

### Lesson 2: (The command-line environment)

(Practice, 90 minutes)

---

- [1] Shell interface
  - + Standard I/O
- [2] Managing files and folders
- [3] Running applications
  - + Arguments

---

### Lesson 3: (How applications run? My first Java application)

(Lecture 30 minutes, Practice 60 minutes)

---

- [1] How applications run on computers
- [2] My first Java application
  - + Coding (text editor)
  - + Compiling (command-line)
  - + Running (command-line)

---

### Lesson 4: (The object-oriented paradigm, Variable types and methods)

(Lecture 30 minutes, Practice 60 minutes)

---

- [1] The object-oriented paradigm
  - + Object attributes/fields
  - + Object functions/methods
- [2] Variables
  - + Variable types
  - + Declaring variables and assigning/reading values
  - + Type casting
  - + Scopes
- [3] Methods
  - + Invocation
  - + Parameters
  - + Return types
- [4] Exercise-1

---

## Lesson 5: (Objects in Java (classes), Instantiating and using objects)

(Lecture 30 minutes, Practice 60 minutes)

---

### [1] Complex types in Java (classes)

- + Fields
- + Methods

### [2] Packages

- + Access modifiers

### [3] Using objects.

- + Creating objects from classes
  - + Accessing fields
  - + Invoking methods
- 

## Lesson 6: (Static variables and methods, Java application packaging)

(Lecture 30 minutes, Practice 60 minutes)

---

### [1] Static variables and methods

### [2] Java application packaging

- + .class files
- + JAR files

### [3] Using 3rd-party libraries.

---

## Lesson 7: (Expressions, operators and statements, Control Flow)

(Lecture 30 minutes, Practice 60 minutes)

---

### [1] Expressions, operators , and statements

### [2] Control flow statements

- + Loops
  - + Conditional
  - + Break/Continue
- 

## Lesson 8: (I/O operations, constructors, and getters/setters)

(Lecture 30 minutes, Practice 60 minutes)

---

### [1] I/O operations

- + Reading from the standard-in
- + Writing to the standard-out/standard-err
- + File operations.

### [2] Object construction

- + The class constructor
- + Getters
- + Setters

### [3] Exercise-2

---

Lesson 9 : (Polymorphism in Java, documentation, the String and Math classes)

(Lecture 30 minutes, Practice 60 minutes)

---

[1] Polymorphism in Java

- + Method overloading

[2] Java documentation (online)

[3] The String and Math classes

---

Lesson 10 : (Inheritance in Java, Exception handling)

(Lecture 30 minutes, Practice 60 minutes)

---

[1] Inheritance in Java

- + Casting between a super and a sub class

- + Method overriding (Polymorphism)

[2] Exceptions and errors

- + Checked exception

- + Runtime exception

[3] Exercise-3

---

Lesson 11 : (Data structures: arrays)

(Lecture 30 minutes, Practice 60 minutes)

---

[1] 1-dimension arrays

- + Declaration

- + Initialization

- + Copying

[2] n-dimension arrays

---

Lesson 12 : (Data structures: lists)

(Lecture 30 minutes, Practice 60 minutes)

---

[1] Lists

- + ArrayList

- + LinkedList

---

Lesson 13 : (Data structures: HashMaps)

(Lecture 30 minutes, Practice 60 minutes)

---

[1] HashMaps

- + Keys

- + Values

---

Lesson 14 : (Sample project: BookStore Manager)

(Lecture 30 minutes, Practice 60 minutes)

---

Writing a sample application to manage books of different types at a book store.

The BookStore has two types of books, ordinary and e-books. The application introduces an interface to the user to add/remove, list, and search for books.

We will use the time of this lesson to develop the Java classes for this project and introduce a working application.

---

Lesson 15: (Final project: ATM)

(Discussion 90 minutes)

---

This session is dedicated to discussing the details of the final project and answering any questions on it.

---