LaSalle College
Montréal

COURSE OUTLINE

Course Identification

Name of programs – Codes:	COMPUTER SCIENCE - ADMINISTRATIVE DATA PROCESSING (420.A0) INFORMATION TECHNOLOGY - PROGRAMMER-ANALYST (LEA.3Q)
Course title:	**DEVELOPING DATABASES APPLICATIONS**
Course number:	420-B35-AS
Total number of course hours:	75 Hours
Weighting:	2-3-2
Statement of the competencies – Codes:	Ensure the quality of an application. (0177) Design and develop an application in a database environment. (017B)

Contribution of the Course in the Program

Course position

Developing Database Applications (420-B35-AS) is offered in the fifth semester of the program *Computer science – administrative data processing* (420.A0) and the third semester of the program *Information technology – programmer analyst* (LEA.3Q). It is a 75 hour course divided into 30 hours of theory and 45 hours of computer lab. In addition, students enrolled in this course are encouraged to complete at least 30 hours of self-study outside of the classroom. The course *Advanced Databases* (420-B24-AS) is a prerequisite to this course; which in itself is a prerequisite for *Internship I* (420-ST1-AS) and *Internship II* (420-ST2-AS). The competency (0177) is shared with the following courses: *Internet Programming II* (420-P65-AS), *Multimedia* (420-M14-AS) *and Information System II* (420-S15-AS). This latter finalizes the competency. The competency (017B) initiated in the course *Advanced Databases* (420-B24-AS) is fully achieved in this course. Learning activities acquired in this course will be reinvested during the internship II (420-ST2-AS)

Scope of the course

This course will focus on advanced object oriented programming using java

As a first step, the student enriches his knowledge and skills in object-oriented programming. First, he examines the concepts of object-oriented programming such as encapsulation, inheritance and polymorphism. Then he creates and uses different kinds of classes such as inner classes, sub classes, abstract classes and interfaces. As for the second step he manipulates different types of collections,

java's utilities classes, and generic classes and methods. Concerning the last step of this course, he reinforce his learning in object oriented by examining other specialized classes such as file management and databases classes

Upon completion of this course, the student will be able to build and use classes according to java object-oriented approach, creates and uses several types of collections as well as he manipulate data stored in text, binary files or oracle databases. He will also be able to refine the code of the class and uses appropriate methods. In addition he will be able to create and handle the program's errors using the Exception class.

Course Components (Objective and Standard of the Competency)

Expected outcomes (achievement context of the competencies)

The achievement context of these competencies will reflect the conditions as they occur in the following settings: academic, professional, work, or life environment.

Achievement context specific to competency Ensure the quality of an application – 0177:

- Using a workstation and the appropriate testing software.
- Using applications representative of those found in the workplace.
- Using applications that employ object-oriented and structured programming techniques.
- Based on unit test results.
- Based on company requirements and data processing standards.
- Based on documentation on each application.
- Using appropriate technical reference manuals.

Achievement context specific to competency Ensure the quality of an application – 017B:

- Using a workstation and the development tools of a database management system.
- Based on a request for the development of an application limited to a few functions.
- Based on company requirements and data processing standards.
- Using appropriate technical reference manuals.
- In co-operation with people involved in the development of the application.

Throughout the course, you will engage in various learning situations/activities so that by the end of the course you will have met the expected outcomes.

Elements and performance criteria

The elements of an objective formulated in terms of the competency specify its essential components. They include only what is necessary in order to understand and master the competency. If the competency is described as a process, the elements are the steps for execution.

The performance criteria are the specific pre-established requirements upon which you and your teacher can objectively judge your development of the targeted competency. They are part of the description of this competency. They are prescriptive.

Sometimes an element appears in more than one course. If this is the case, a number indicates its complexity level: level one (1) being the simplest, level two (2), average, and level three (3), advanced, at the ministerial level.

Below are the elements of the competencies and performance criteria for this course that are to be respected:

Competency: Ensure the quality of an application – 0177 ***General ministerial and institutional performance criteria:*** • Autonomy • Work quality	
Elements of the competency	***Performance criteria specific to each element***
1. Plan tests.	1.1 Complete analysis of the application's features. 1.2 Complete analysis of the unit test results. 1.3 Establishment of the relationship between the expected quality of the application and the potential demonstrated in the various tests. 1.4 Proposal of an appropriate test sequence. 1.5 Exploration of the possibility of using a test library. 1.6 Appropriate test planning.
2. Run the various tests.	2.1 Creation of appropriate tests for the application. 2.2 Effective use of the test software. 2.3 Correct programming of tests. 2.4 Strict application of integration, integrity and performance testing techniques. 2.5 Proper use of test libraries. 2.6 Perseverance in conducting the tests. 2.7 Adherence to the testing schedule.
3. Verify the quality of the application.	3.1 Interpretation of results in accordance with established quality requirements. 3.2 Recording of all data on the tests and test results. 3.3 Evaluation of the need to redo certain tests and to adapt the testing procedures.

Competency: Design and develop an application in a database environment. (017B)

General ministerial and institutional performance criteria:

- Autonomy
- Work quality

Elements of the competency	***Performance criteria specific to each element***
1. Define the functions of the application.	1.1 Determination of the customer's needs. 1.2 Proper use of data collection techniques. 1.3 Complete analysis of the characteristics of activities related to the application to be developed. 1.4 Deduction and justification of the application's functions. 1.5 Global representation of the functions. 1.6 Production of a clear and complete report. 1.7 Application of professional ethics rules. 1.8 Effective communication with all project participants. 1.9 Proposal of creative solutions suited to the customer's needs.
2. Define the technological requirements.	2.1 Thorough analysis of the characteristics of the data processing and customer environments. 2.2 Identification of the hardware and software required for the development and installation of the application. 2.3 Justification of choices according to priorities and requirements. 2.4 Search for ways to optimize the existing computing environment. 2.5 Search for data processing solutions to meet established needs. 2.6 Proposal of a pertinent plan for bringing the system up to standard. 2.7 Production of a clear and comprehensive report.
3. Lay the groundwork for the application.	3.1 Creation of a realistic work schedule. 3.2 Appropriate use of planning methods and tools. 3.3 Acquisition of the proper environment in which to develop the application.
4. Model the application.	4.1 Appropriate modelling of the data. 4.2 Creation of an appropriate data dictionary. 4.3 Proper modelling of processes. 4.4 Proper modelling of events. 4.5 Proper creation of tables. 4.6 Complete adaptation of the planning schedule. 4.7 Production of clear and comprehensive

	technical documents.
5. Produce a user interface using a prototype.	5.1 Effective use of development tools. 5.2 Creation of menus inputs and outputs consistent with data type and customer needs and requirements. 5.3 Complete validation of interface performance. 5.4 Clear presentation of the user interface for approval. 5.5 Modification of the interface consistent with customer requests. 5.6 Adaptation of algorithms to the user interface. 5.7 Appropriate modification of tables. 5.8 Production of complete documentation on the interface. 5.9 Proper reconciliation of functional and aesthetic requirements.
6. Develop programs.	6.1 Appropriate coding of functions and procedures. 6.2 Proper use of function libraries and procedures specific to the database management system. 6.3 Appropriate programming of modules. 6.4 Optimal use of the programming language. 6.5 Thorough verification of the performance of each program and of the application in the development environment. 6.6 Production and archiving of comprehensive documentation for the programs.
7. Produce documentation for the application.	7.1 Appropriate modification of all data on the application. 7.2 Appropriate creation of an on-line help function. 7.3 Production of a clear and comprehensive user manual for the application.

Course Content/Main Themes

*Listed below is the **essential** content to be covered in this course:*

CLASSES AND OBJECTS
Instance and class attributes, instance and class methods, constructors, types of modifiers, encapsulation, this operator, getters and setters, overloading and overriding methods, overriding of the methods toString and equals, relationship between classes (association, aggregation, etc ...), manipulating array of objects, inner classes, generic classes and methods.

SUB-CLASSES

Inheritance of classes, using attributes of the upper class, calling the constructors and methods of the upper class, manipulating arrays containing objects of a class and its subclasses, building and using of abstract classes, building and using interfaces, using the interface Comparable, implementing the method compareTo, Sorting an array of objects.

COLLECTIONS
Types of collection, manipulating collections (TreeSet, ArrayList, TreeMap)

EXCEPTIONS
The exception handling block: try, catch, and finally, throwing exception using the keywords throw and throws, creating and using the class Exception.

BINARY AND TEXT FILES
Declaring, reading, writing in a text and binary file. Manipulating of objects in a text or binary file. using the interface Serializable and related methods.

DATABASES
Classes and methods related to oracle database , creating and using data manipulation classes

Learning Activities

Provided below are examples of learning activities that correspond to the competencies for this course. The learning activities are found in the course calendar that complements this course outline.

- Brain storming
- Application exercises following demonstrations from teacher
- Case studies
- Situation problems
- Project

Terms for Evaluating Learning

The evaluation of your learning is based on two inseparable methods: formative evaluation and summative evaluation. These two evaluation types are formal. Detailed information on the evaluation schedule is found in the course calendar, under the "Formative and summative evaluation schedule" column.

Formative evaluation

Following a learning activity or learning period, time is set aside for introspection. You will determine what has been understood and achieved and seek to identify the nature and origin of weak areas. These designated periods consist of simple means: short tests, association games, logbooks, a portfolio, questions, creating of samples, etc.

Formative evaluation is frequent and covers as many aspects as possible. It takes place in class,

individually or in groups, and leads to immediate decisions. ***You are the one who assumes the bulk of the work during individual or group corrections, adjustments and other self-evaluation tasks. The purpose is not to determine grades.***

If you take the formative evaluations seriously throughout the course, you will ensure preparedness for the summative evaluations. You will be able to make the necessary progress to acquire the targeted competency at the required level, according to the achievement context and pre-established performance criteria.

Below are some examples of formative evaluation methods that correspond to the targeted competencies for this course:

- Situation problems on data manipulation of database using java classes.
- Case studies
- Feedback from the teacher after students application exercises

Summative evaluation

Summative evaluations are less frequent. They take place later on, towards the middle and end of the semester. This gives you the time to integrate your learning and to learn how to apply it to situations related to the targeted competency. The summative evaluation material is prepared by your teacher according to the description of the course's targeted competency: its elements, achievement context and performance criteria.

The work completed in summative evaluations is graded. The purpose is to determine what you have learned.

Below is the information on the summative evaluation schedule and details for this course, as well as the weighting of marks:

Evaluations	***Weighting***
Mid-term exam	30%
Project	30%
Final exam	40%
Total	100 %

Institutional Requirements

Student's commitment

By registering for this course, you commit to:

- *obtaining the necessary course materials at the start of the semester;*
- *participating in the learning activities, formative and summative evaluation activities outlined in the course calendar;*
- *completing the work assigned to you and;*
- *submitting the work on time.*

Teacher's commitment

Your teacher commits to:

- *create varied learning situations that enable you to put into practice the knowledge, actions and professional behaviour of the targeted competency;*
- *plan sufficient and appropriate formative evaluation activities – involving correction and improvement – that provide frequent feedback, allowing you to be well informed of your progress;*
- *provide summative evaluations that correspond to the course's targeted competency and;*
- *evaluate work according to the applicable criteria, in a fair and equitable manner.*

The Institutional Policy on Evaluating Learning (IPEL) is applied to all institutional programs. Listed below are a few of its clauses:

Written language

The teacher is responsible for identifying spelling and grammar errors and for deducting the corresponding number of marks for any given summative evaluation.

Below is the % – based on language requirements – that can be deducted from the grade of each summative evaluation:

- Penalty of up to 15% for *Computer science – administrative data processing* (420.A0)
- Penalty of up to 10% for *Information technology – programmer analyst* (LEA.3Q).

Class attendance

Attendance and participation in classes and evaluations are mandatory for all students.

The teacher has the responsibility of monitoring attendance and of evaluating the reasons justifying student absences from classes.

A student whose absences exceed the allowable number for the course could be denied access to the final exam for that course.

Plagiarism and cheating

Plagiarism, attempts at plagiarism or complicity in plagiarism, whether in an exam or an assignment to be evaluated, constitutes an infraction. Plagiarism and cheating include:

- *using part or all of someone else's work and passing it off as one's own, without indicating the appropriate reference;*
- *having or using unauthorized documents, material or equipment during an exam;*
- *using the exam of another student during an exam;*
- *having another student do one's work for an evaluation;*
- *substituting a different person to write an exam or assignment to be evaluated and;*
- *using an evaluation already completed for another course.*

Plagiarism, attempts at plagiarism or fraud, or collaboration in plagiarism or fraud are prohibited and considered serious offences. Thus, any instances of plagiarism or fraud will lead to a grade of '0' for the assignment in question. In addition, a note will be made in the student's file and the student will receive a written notice from his or her Program Directorate to that effect.

In the case of recidivism, in the same course or in another course, the student will be given a grade of '0' for the course in question. A second note is made in the student's file and the student will receive a summons from his or her Program Directorate. For a third offence, he or she may be expelled from the College.

Submission of work and tests

All assignments must be submitted in class at the time designated by the teacher. Any late submissions result in a grade of zero (0).

Upon presentation of an official supporting document or valid reason for the absence, the student may request an extension from the teacher, who may accept or refuse the student's work and apply a penalty for the lateness.

Program Directorates do not accept student work. Assignments must be submitted directly to the teacher.

Rules and regulations to follow

Late arrivals

The teacher may refuse to admit to the classroom any student arriving late. A late arrival is considered an absence for that period.

Note: Students arriving late must recognize that the information they missed will not be repeated. Late students are therefore responsible for asking their peers about the material they missed. Arriving after the break, as well as leaving before the end of the class, may result in one or more hours of absence.

Eating and drinking in class

Eating and drinking are prohibited in the classrooms, locker rooms and Documentation Centre. Food may only be eaten in the cafeteria, vending machine areas and student lounges.

Mandatory Course Material

- USB key for backup work

Bibliography for this Course

Site web de java http://java.sun.com

Cay Horstmann, Big Java: Compatible with Java 5, 6 and 7, 4th Edition, WILEY, 2009

Academic Studies Directorate Approval: *Signature and date of approval*

Benjamin Bousquet-Lemieux Pour la DE

Signature numérique de Benjamin Bousquet-Lemieux Pour la DE
DN : cn=Benjamin Bousquet-Lemieux Pour la DE, o=College Lasalle, ou=Coordination au soutien pédagogique, email=benjamin.bousquet-lemieux@collegelasalle.com, c=CA
Date : 2015.06.15 15:54:16 -04'00'