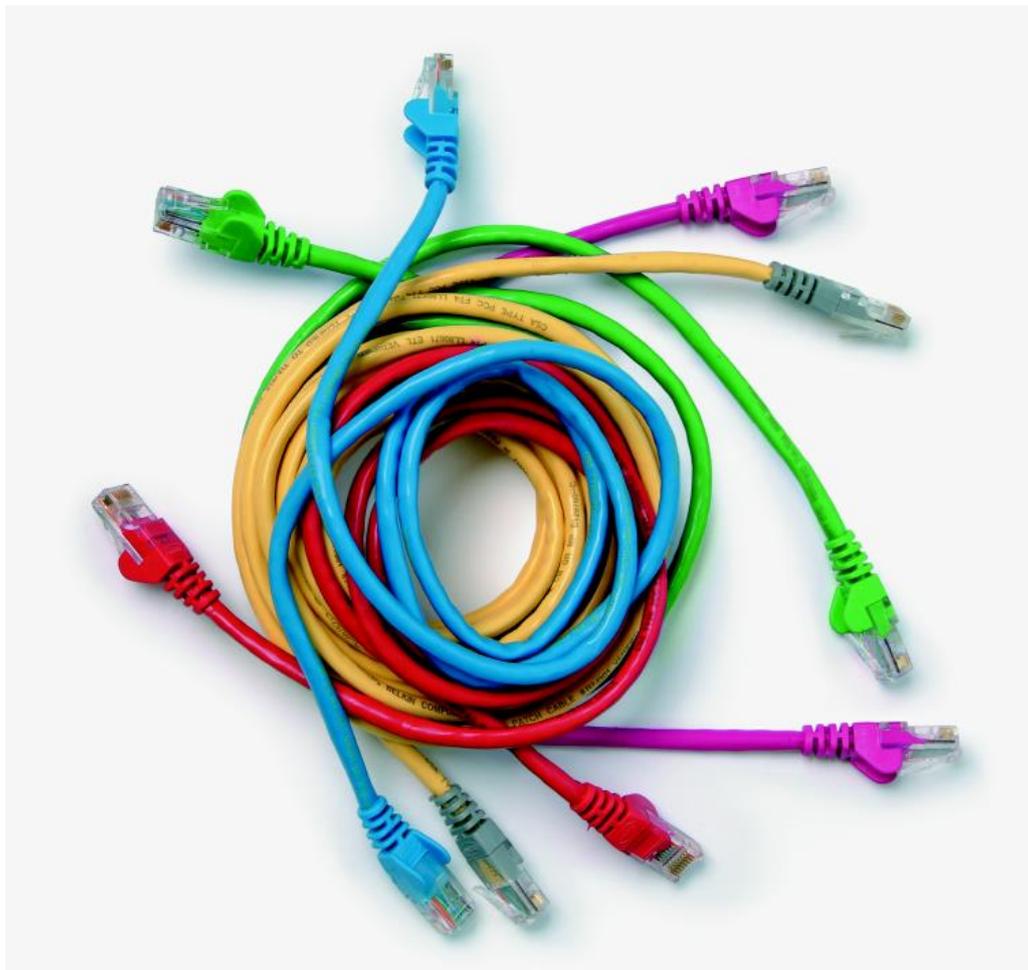


Level 3 Creating an object oriented computer program using Java (7540-037/7630-321)

Assignment guide for Candidates Assignment A



About City & Guilds

City & Guilds is the UK's leading provider of vocational qualifications, offering over 500 awards across a wide range of industries, and progressing from entry level to the highest levels of professional achievement. With over 8500 centres in 100 countries, City & Guilds is recognised by employers worldwide for providing qualifications that offer proof of the skills they need to get the job done.

City & Guilds Group

The City & Guilds Group includes City & Guilds, ILM (the Institute of Leadership & Management) which provides management qualifications, learning materials and membership services, NPTC which offers land-based qualifications and membership services, and HAB (the Hospitality Awarding Body). City & Guilds also manages the Engineering Council Examinations on behalf of the Engineering Council.

Equal opportunities

City & Guilds fully supports the principle of equal opportunities and we are committed to satisfying this principle in all our activities and published material. A copy of our equal opportunities policy statement is available on the City & Guilds website.

Copyright

The content of this document is, unless otherwise indicated, © The City and Guilds of London Institute 2010 and may not be copied, reproduced or distributed without prior written consent.

However, approved City & Guilds centres and learners studying for City & Guilds qualifications may photocopy this document free of charge and/or include a locked PDF version of it on centre intranets on the following conditions:

- centre staff may copy the material only for the purpose of teaching learners working towards a City & Guilds qualification, or for internal administration purposes
- learners may copy the material only for their own use when working towards a City & Guilds qualification

The *Standard Copying Conditions* on the City & Guilds website also apply.

Please note: National Occupational Standards are not © The City and Guilds of London Institute. Please check the conditions upon which they may be copied with the relevant Sector Skills Council.

Publications

City & Guilds publications are available on the City & Guilds website or from our Publications Sales department at the address below or by telephoning +44 (0)20 7294 2850 or faxing +44 (0)20 7294 3387.

Every effort has been made to ensure that the information contained in this publication is true and correct at the time of going to press. However, City & Guilds' products and services are subject to continuous development and improvement and the right is reserved to change products and services from time to time. City & Guilds cannot accept liability for loss or damage arising from the use of information in this publication.

City & Guilds

1 Giltspur Street

London EC1A 9DD

T +44 (0)844 543 0000 (Centres)

T +44 (0)844 543 0033 (Learners)

F +44 (0)20 7294 2413

www.cityandguilds.com

learnersupport@cityandguilds.com

Contents

Introduction – Information for Candidates	2
Candidate instructions	3

Level 3 Creating an object oriented computer program using Java (7540-037/7630-321)

Assignment A

Introduction – Information for Candidates

About this document

This assignment comprises all of the assessment for Level 3 Creating an object oriented computer program using Java (7540-037/7630-321).

Health and safety

You are asked to consider the importance of safe working practices at all times.

You are responsible for maintaining the safety of others as well as your own. Anyone behaving in an unsafe fashion will be stopped and a suitable warning given. You will **not** be allowed to continue with an assignment if you compromise any of the Health and Safety requirements. This may seem rather strict but, apart from the potentially unpleasant consequences, you must acquire the habits required for the workplace.

Time allowance

The recommended time allowance for this assignment is **4 hours**.

Level 3 Creating an object oriented computer program using Java (7540-037/7630-321)

Candidate instructions

Candidates are advised to read **all instructions** carefully before starting work and to check with your assessor, if necessary, to ensure that you have fully understood what is required.

Time allowance: 4 hours

Assignment set up: A scenario is provided for candidates in the form of a company specification for a service they require.

This assignment is made up of **three** tasks

- **Task A** - provides a detailed design specification that should be followed by candidates when developing their program.
- **Task B** - requires the candidate to test the program and provide documentation.
- **Task C** - provides criteria that should be followed by candidates when producing their work.

Scenario

An insurance company, EveryQuote, is developing software for validating and converting dates to use as a library routine available within the company. As a contracted employee of EveryQuote, you have been asked to create and test the software which will perform the following functions:

- Determine whether a date is valid.
- Determine whether a certain year is a leap year.
- Determine the Julian date for a specified date.
- Determine the number of days between two dates.
- Determine a person's age as at today's date given their birth date.

Task A

Candidates should use the following detailed specification to fulfil the company's requirements.

In this task you are required to create and test a new class called newDate.

- 1 Write the code for the following methods in the newDate class
 - a **public boolean validDate (int day, int month, int year)**
The month can only be in the range 1-12 and a month must have the correct number of days eg 30.2.1997 is invalid because February can only have 28 days, or 29 days if the year is a leap year.
This method must return true if the date is valid and false if the date is invalid.
 - b **public boolean leapYear (int year)**
If the year can be evenly divided by 4, but not 100, it is a leap year. However, years that are evenly divisible by 400 are also leap years.
This method must return true if the year is a leap year and false if the year is not a leap year.
 - c **public int julianDate (int day, int month, int year)**
The Julian calendar numbers each day of the year according to its ordinal value, from 1 to 365 (or, in a leap year, from 1 to 366). Therefore, February 1 has a Julian date of 32.
This method must return the value of the Julian date (ie the number of days).
 - d **public int daysElapsed (int day1, int month1, int year1, int day2, int month2, int year2)**
This method returns the actual number of days between any 2 dates.
 - e **public int ageDate (int dayBirth, int monthBirth, int yearBirth)**
This method must return a person's age in years as at today's date.
- 2 Write the code for a testnewDate class which tests the methods in the newDate class.
 - a A menu should be displayed on the screen with the following options:

MENU

1	Leap year
2	Julian date
3	Days elapsed
4	Age
5	Quit

- b If option 1 is selected a prompt should be displayed requesting a year (yyyy). LEAP YEAR should be displayed if the year is a leap year otherwise NOT A LEAP YEAR should be displayed.
- c If option 2 is selected a prompt should be displayed requesting a date (dd-mm-yyyy). If the date is valid the Julian date should be displayed otherwise INVALID DATE should be displayed.
- d If option 3 is selected a prompt should be displayed requesting a start date (dd-mm-yyyy) and an end date (dd-mm-yyyy). If both dates are valid the number of days between the two dates should be displayed otherwise INVALID DATE should be displayed.
- e If option 4 is selected a prompt should be displayed requesting a birth date (dd-mm-yyyy). If the date of birth is a valid date the age in years should be displayed otherwise INVALID DATE should be displayed.

- f If option 5 is selected the program should exit.
- g The option number must be in the range 1-5.

Task B

In this task you are required to test the program you have created and provide documentation.

- 1 Create test data to test the newDate class and determine the expected results.**
- 2 Prepare a test plan, test the software, compare the actual results to the expected results keeping a log for each test which identifies any discrepancies between actual and expected results and records any amendments made to correct errors. Use available debugging tools to help locate and resolve errors.
- 3 Produce technical documentation to describe the class interface and purpose and operation of the newDate class.
- 4 Print a listing of the code for both classes.

Task C

Candidates should follow the criteria below when producing their work:

- 1 The program conforms to the design specification.**
- 2 Meaningful names are used for classes, methods and attributes using consistent naming conventions.

Note

- Candidates should produce the following for their assessor:
 - A printed program listing of both classes (newDate and testnewDate).
 - Test data, test plan, expected results, actual results and the log of testing.
 - Technical documentation.
- At the conclusion of this assignment, hand all paperwork and removable media to the test supervisor.
- Ensure that your name is on the removable media and all documentation.

If the assignment is taken over more than one period, all removable media and paperwork must be returned to the test supervisor at the end of each sitting.

Published by City & Guilds
1 Giltspur Street
London
EC1A 9DD
T +44 (0)844 543 0000 (Centres)
T +44 (0)844 543 0033 (Learners)
F +44 (0)20 7294 2413
www.cityandguilds.com

City & Guilds is a registered charity
established to promote education
and training