



Coursework Assignment Brief

INST2002

Assignment Title	Coursework: An Online Dry-Food Store
Component/Module	Programming Module
Assignment Code/Number	B
Set by	Luke Dickens
Moderated by	Karen Stepanyan

Learning outcomes to be assessed:

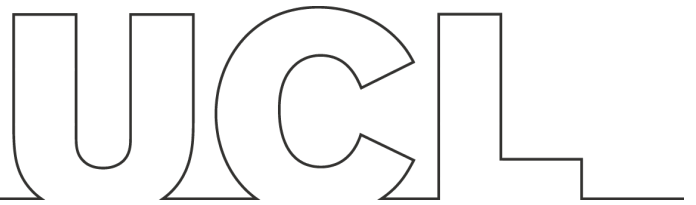
- Understanding of terms: class, method, return value, input argument, code reuse, superclass, subclass, interface, apparent type, actual type, and casting.
- Mastery of keywords/operators: **new, return, public, private, if, else, for, switch, case, this, final, static, ||, &&, ==, !=, !, <, >** and **[]** (square brackets).
- Use of new keywords/operators: **super, extends, interface, implements, instanceof, ()** - for casting, and **<>** for generics.
- Mastery of type definitions: int, boolean, double, String and user defined types, for locally defined variables and method input arguments.
- New uses of type definitions: upcasting, downcasting, List and ArrayList.
- Understanding of concepts:
 - different apparent and actual types
 - interfaces as a contract
 - superclass representing is-a relation
- Problem Solving:
 - using class hierarchies to minimise duplication
 - inheriting attributes and methods
 - reading and writing list elements
 - in place operations
 - opportunity to demonstrate understanding of recursion



Submission requirements:

- Before attempting the task, students are strongly encouraged to familiarise themselves with the specification.
- Students should compile regularly throughout to ensure that the final submission compiles.
- Students should package the files and the local folders into a zip file and submit as a single zip-archive.
- Submission is through the moodle submission page for INST2002 ([link](#)).

Basic Assignment Description	The student must complete an implementation of the online dry-food store in Java as described in the pdf document on moodle.
Conditions	Each student must work individually. A task description and code can be downloaded in advance, and a solution developed on the student's local machine. Submission is through moodle as a zip file. All work, including the submission, must be done by the student alone.
Marking Criteria and Weighting Rubric	<p>The students code will be assessed for correct functionality and coding style. The total mark for this assessment consists of 70% for correct functionality and 30% for coding style. Of the mark for functionality 40% is assigned to the product-list and associated classes and 30% to the basket and associated classes.</p> <p>If a student fails to meet the precise functionality for any part of the coursework, then partial credit will be given where the student goes some way towards meeting the requirements.</p> <p>Good coding style includes: good variable names, appropriate use of control flow (e.g. for loops, if statements and recursion), appropriate variables with good names, code reuse and minimising duplication of code, appropriate helper methods & additional classes, appropriate use & handling of exceptions, meaningful & appropriate comments, good package structure, and consistent formatting.</p>



The assignment is worth % of the overall assessment for this course

This assignment **must** be completed:

Date work set (provisional):

Date and time due in (provisional):

Standard lateness penalty will apply

Target date for return of marked work and full feedback (provisional):
(tentatively within 4 working weeks, according to DIS policy)

A detailed description of the assignment is: here."/>