UTS Insearch
UNIVERSITY OF TECHNOLOGY SYDNEY

Project A

Diploma of IT

Programming Fundamentals

SEMESTER 3 2021

Weight: 15%

DUE : Week 6

ASSESSMENT TASK 1 : CREATING BASIC JAVA CLASSES AND METHODS

TASK DESCRIPTION

This project requires you to write a basic Java classes with attributes and methods. Methods required include an alternate constructor, methods to display output and methods to read user input. Code must be written that creates objects and then calls methods for these objects.

OBJECTIVES

- Demonstrate a working knowledge of the basic constructs in the object-oriented language Java
- Use the BlueJ programming environment to build simple software systems
- Design and code a software system that correctly implements a solution to a small problem defined by a specification, and follows specific design rules
- Comprehend and use basic program control constructs of sequence and selection

BRIEF

This is a take home assessment to evaluate your understanding of object oriented concepts taught so far in IPRG001. The assessment will be composed of three (3) parts: the scenario, the code and explanation (including proof of testing).

The scenario:

You will decide on the scenario and develop a back story for your program – it could be a game of some sort or simply display some information about your chosen things in a specific way.

Choose two objects – they could be your favourite things e.g. a book, a game, an animal (real or imaginary), a character (real or imaginary) etc.

The code:

Create two classes (class1 & class2, one for each thing – choose a good name for each class), for each class:

Identify at least 3 attributes – an int/double, a String, a boolean and a constant must be present in at least one of the classes.

Design an alternate constructor.

Create the accessors and mutators for each attribute.

Design a suitable toString()

Create a main class (choose a name for it). Create a Scanner object as a local variable in the main class – where ever it is needed. The main class creates an object of class1 and an object of class2 as attributes – choose appropriate names for them. The default constructor should initialise both objects. Each object's contents should be displayed using the toString().

User entry must change both objects in some way (using mutators).

User must be presented with a decision and the response processed appropriately using an if statement.

Use multiple methods in the main class - where appropriate.

The explanation:

You will explain the code you wrote and any problems you encountered and how you fixed them. At minimum you need to identify :

Basic Class structure – what does a class have

Attributes – what attributes did you choose for each class and why

Alternate constructors – what parameters did you chose and why

Non-void and void methods – which methods are void and non-void and why

Accessor – which methods are accessors

Mutators – which methods are mutators

Access modifiers – what modifiers are used in your code and why

Constants – which values are stored as constants, what modifier did you use and why.

toString() – what attributes did you include and why

Scanner - identify the scanner methods you used and why

if statements – document any if statements you used and how they support decisions.

You also need to include at least 1 screen shot of your code being testing on the object bench.

Please double check your code compiles and your submission document contains the updated version of all code prior to submission.

SUBMISSION

You will submit your assessment in MS Word document format – yourName-studNo.doc.

The document will have 3 sections:

The scenario: You will write the back story for your program – this will be assessed using LLO minimum 100 words

The Code: You will copy the code from your BlueJ project into this section

The explanation: you will explain your code choices and reasoning. You also need to include at least one screenshot of testing your code. Include any issues you had and how you resolved them. Minimum 300 words.

The deadline for submission of this project is Friday 11:59pm in week 6.

Draft Project idea: due Friday 11:59pm Week 4 for tutor approval

Draft Project A: due Tutorial B Week 6 – draft code review by tutor

All projects submitted after the deadline will incur late penalties of 1 grade band every 2 days (or part thereof).

ASSESSMENT CRITERIA

CRITERIA	WEIGHT	SLOs / LLOs	PLOs
Demonstrate a working knowledge of basic Java code	40%	1, 2	A1, A3, E1
Identify correct use of object oriented rules and patterns	40%	3, 4	A2, A3, B1, B2
Can use a more than adequate range of vocabulary, mostly appropriately, with developing flexibility and precision	20%	W4	

SLOs: subject learning outcomes
PLOs: program learning outcomes
LLOs: language learning outcomes

Please refer to the next page for the rubric definition for each criteria and grade.

		HD	D	C	P	F3	F2	F1	F0
1	**Demonstrate a working knowledge of basic Java code.**	The code is exceptionally well organised and variable names chosen clearly explain what the code is accomplishing and how. Camel case always used.	The code is easy to follow and well organised and variable names chosen are very useful in understanding what the code does and how it is accomplished. Camel case mostly used.	The code is fairly easy to follow and variable names chosen are useful in understanding the code. Camel case sometimes used.	The code is a little hard to read and variable names chosen are somewhat useful in understanding the code. Camel case used at least once.	The code is hard to read and variable names chosen at least once help understand what they are used for. Camel case not used.	The code is poorly organised and very difficult to follow. Variable names chosen do not help understand what they are used for. Reader must guess what variables are for. Camel case not used.	The code is poorly organised and very difficult to follow. Variable names chosen use names that will not compile e.g. with spaces. Camel case not used.	No valid code written
2	**Identify correct use of object oriented rules and patterns.**	The program works correctly in every way and meets all of the functional specifications.	The program works and produces correct results except in one area. It also meets most of the other specifications.	The program mostly works and produces mostly correct results. It also meets most of the other specifications.	The program is producing some incorrect results or a part of the program is incomplete.	The program is producing mostly incorrect results or a most of the program is incomplete.	The program is producing all incorrect results or mostly incomplete.	The program does not compile or run; Some code is valid	Program does no required tasks

W4	**Can use a more than adequate range of vocabulary, mostly appropriately, with developing flexibility and precision**	Scenario is an engaging, unique idea – clearly explained and well thought out. Explanation correctly identifies all required elements and clearly justifies coding choices. Work is always clear and comprehensive	Scenario is a unique idea – clearly explained and well thought out. Explanation correctly identifies almost all required elements and justifies most coding choices. Work is mostly clear and comprehensive	Scenario is a unique idea – mostly explained. Explanation correctly identifies most required elements and justifies some coding choices. Work is mostly clear and comprehensive	Scenario is a mostly unique idea –some parts explained. Explanation correctly identifies some required elements and justifies a few coding choices. Work is somewhat clear and comprehensive	Scenario is not a unique idea – some parts maybe explained. Explanation correctly identifies few required elements and justifies very few coding choices. Work is somewhat clear and comprehensive	Scenario is difficult to understand, is flawed or inadequate. Explanation correctly identifies very few required elements or justifies very few coding choices. Work is substandard	Scenario is confused or copied from another source. Explanation does not identify any elements or code choices; or is copied from another source. Work is not acceptable.	No Sub-mission