

COMP2603 Assignment 1

Mark Scheme: 100 Marks

Class Design	Marks		Program	Marks
Passenger	16		Compiles	5
LuggageSlip	14		Runs (LuggageManagementSystem)	10
LuggageManifest	20		Passes JUnit Tests	14
Flight	16		Has clean code	5
Total	66		Total	34

Passenger Class: 16 marks

Attribute	Type	MARKS	CRITERIA
passportNumber	String	1	Private, instance variable, declared only.
flightNo	String	1	Private, instance variable, declared only.
firstName	String	1	Private, instance variable, declared only.
lastName	String	1	Private, instance variable, declared only.
numLuggage	int	1	Private, instance variable, declared only.
cabinClass	char	1	Private, instance variable, declared only.

Method Signature	Return Type	MARKS	CRITERIA
Passenger (String passportNumber, String firstName, String lastName, String flightNo)		5	Constructor. - Uses the 4 input parameters to set the state. (2 marks) - Sets the numLuggage variable randomly (2 marks) - Sets the cabinClass variable randomly (1 mark)
assignRandomCabinClass()	void	2	Sets the cabinClass variable to a random cabin class value : 'F', 'B', 'P', 'E'

Method Signature	Return Type	MARKS	CRITERIA
toString()	String	3	Returns a String representation of a Passenger object. See format below

Format for the toString() output for a Passenger, Joe Bean, with passport number TA890789:
PP NO. TA890789 NAME: J.BEAN NUMLUGGAGE: 3 CLASS: E

LuggageSlip Class: 14 marks

Attribute	Type	MARKS	Purpose
owner	Passenger	1	The Passenger who owns/checks in the luggage.
luggageSlipIDCounter	int	1	A <u>class</u> variable that starts at 1 and increments by 1 for each new LuggageSlip object created.
luggageSlipID	String	1	A unique identifier for the LuggageSlip, produced using Passenger's flight number, last name and the luggageSlipIDCounter. Example below
label	String	1	A string variable for recording any additional details on the luggage slip.

Method Signature	Return Type	MARKS	Purpose
LuggageSlip (Passenger p, Flight f)		3	Constructor. Initialises all state using the input parameters, and sets the label to an empty String.
LuggageSlip (Passenger p, Flight f, String label)		3	Overloaded constructor. Initialises all state using the input parameters (including label).
hasOwner (String passportNumber)	boolean	2	Return true if the owner of the LuggageSlip has the supplied passportNumber, false otherwise.
toString()	String	2	Returns a String representation of the LuggageSlip object. See format below.

Sample luggageSlipID values for a Passenger, Joe Bean, with three pieces of luggage on flight BW600:

BW600_Bean_1

BW600_Bean_2

BW600_Bean_3

Format for the toString() output for a LuggageSlip for Joe Bean

BW600_Bean_1 PP NO. TA890789 NAME: J.BEAN NUMLUGGAGE: 3 CLASS: E \$105

BW600_Bean_2 PP NO. TA890789 NAME: J.BEAN NUMLUGGAGE: 3 CLASS: E \$105

BW600_Bean_3 PP NO. TA890789 NAME: J.BEAN NUMLUGGAGE: 3 CLASS: E \$105

The additional token at the end (\$105) is the label (if set).

LuggageManifest Class: 20 marks

Attribute	Type	MARKS	CRITERIA
slips	ArrayList<LuggageSlip>	2	Private, instance variable, declared outside constructor, initialised in constructor

Method Signature	Return Type	MARKS	CRITERIA
LuggageManifest ()		1	Constructor. - Initialises the slips collection.
addLuggage(Passenger p, Flight f)	String	6	- Creates one or more new LuggageSlip objects - Adds the LuggageSlip object(s) to the slips collection based on the number of pieces of luggage that a Passenger has. - Checks the number of allowed pieces based on the Passenger's cabin class and - Calculates the cost of excess luggage. - Adds the excess luggage cost as a label on all luggage slips for the Passenger. The method returns the output indicated below*.
getExcessLuggageCost(int numPieces, int numAllowedPieces)	double	3	Calculates and returns the total cost of adding excess luggage based on the number of allowed pieces. Every excess piece of luggage costs \$35.00.
getExcessLuggageCostByPassenger(String passportNumber)	String	5	Returns the total cost of excess luggage (if any) on the manifest for a Passenger with a given passport number or "No Cost" otherwise.
toString()	String	3	Returns a String representation of the aggregated state of a LuggageManifest by traversing and returning the String representation of each LuggageSlip (if present).

*Sample output for the addLuggage(..) method for Passenger 1 - 2 pieces of luggage, 1 excess
PP NO. TA890789 NAME: J.BEAN NUMLUGGAGE: 2 CLASS: P
Pieces Added: (2). Excess Cost: \$35

*Sample output for the addLuggage(..) method for Passenger 2 - 0 pieces of luggage
PP NO. TA890789 NAME: J.BEAN NUMLUGGAGE: 0 CLASS: P
No Luggage to add.

*Sample output for the addLuggage(..) method for Passenger 3 - 1 piece of luggage, 0 excess
PP NO. TA890789 NAME: J.BEAN NUMLUGGAGE: 1 CLASS: B
Pieces Added: (1) Excess Cost: \$0

Sample toString() output for the LuggageManifest :

LUGGAGE MANIFEST:

BW600_Bean_1 PP NO. TA890789 NAME: J.BEAN NUMLUGGAGE: 2 CLASS: F
 BW600_Bean_2 PP NO. TA890789 NAME: J.BEAN NUMLUGGAGE: 2 CLASS: F
 BW600_Deer_3 PP NO. BA321963 NAME: L.DEER NUMLUGGAGE: 2 CLASS: P \$35
 BW600_Deer_4 PP NO. BA321963 NAME: L.DEER NUMLUGGAGE: 2 CLASS: P \$35

Flight Class: 16 marks

Attribute	Type	MARKS	CRITERIA
flightNo	String	1	Private, instance variable, declared only.
destination	String	1	Private, instance variable, declared only.
origin	String	1	Private, instance variable, declared only.
flightDate	LocalDateTime	1	Private, instance variable, declared only.
manifest	LuggageManifest	1	Private, instance variable, declared only.

Method Signature	Return Type	MARKS	CRITERIA
Flight (String flightNo, String destination, String origin, LocalDateTime flightDate)		2	Constructor. Initialises the state variables, with the supplied parameters if applicable, and creates a new LuggageManifest object
checkInLuggage(Passenger p)	String	5	Validates whether a Passenger can check in luggage for the flight (same flight number as the Passenger's). If this is true, the method adds the Passenger's luggage to the flight using the addLuggage(..) method and returns the String outcome. If the Passenger's flight number does not match the Flight, then the message "Invalid flight" is returned.
printLuggageManifest()	String	1	Returns a String representation of the manifest
getAllowedLuggage(char cabinClass)	int	2	A class method that returns the number of allowed pieces of luggage for a given cabin class, after which a cost is incurred. 'F': 3 pieces 'B': 2 pieces 'P': 1 piece 'E': 0 pieces
toString()	String	1	Returns a String representation of the Flight.