COMP503 Programming 2/ENSE502 Object-Oriented Applications
Assignment 1 – The Cinema Ticket Booking Application

Due date: Friday, 18th March 2016 at 5pm on AUTOnline.
This assignment is worth 10% of your final grade.

Brief

For this assignment, you will individually develop a Cinema Ticket Booking Application in Java with the following functionality, available from the console:

- Requests the Customer's name, age and asks if they are a student
- Lists films that are currently available to watch at the cinema
- Asks the customer to select the film to watch from a menu
- Issues a ticket if age appropriate
- Calculates the cost of the ticket and prints the ticket details to the console with the customers details, the film's detail and the cost

Methodology and Marking Scheme

This assignment has 100 marks in total
You will develop five classes with the following instance variables and methods

- **Film** class **(20 marks in total)**
    - Stores the title of a film and the rating of the film **(4 marks)**
    - Has a constructor with inputs for film title and rating **(4 marks)**
    - Has a default constructor with suitable default values **(4 marks)**
    - Has appropriate data encapsulation methods **(4 marks)**
    - toString method returns a string representation of a Film object **(4 marks)**

- **Customer** class **(20 marks in total)**
    - Stores the customer's name, age, and if they are currently a student **(4 marks)**
    - Has a constructor with inputs for name, age and student **(4 marks)**
    - Has a default constructor with suitable default values **(4 marks)**
    - Has appropriate data encapsulation methods **(4 marks)**
    - toString method returns a string representation of a Customer object **(4 marks)**

- **RATING** enumerated type **(16 marks in total)**
    - Has elements corresponding to New Zealand's film rating systems: **(8 marks)**
        - General, films appropriate for all ages
        - Parental Guidance, ages 12 and above
        - Mature, ages 16 and above
    - Has an instance variable storing the minimal age for the film rating **(4 marks)**
    - Has a constructor with input for minimal age **(4 marks)**

Dr. Kenneth Johnson, Auckland University of Technology, kenneth.johnson@aut.ac.nz

- **Ticket** class **(24 marks in total)**
    - Stores a Customer object and Film object **(4 marks)**
    - Has a method double cost() which calculates ticket cost as follows: **(10 marks)**
        - Standard ticket for adults: (aged 18 years or more) $10
        - Standard tickets for children: (aged less than 18 years) $7
        - Students between the ages of 10-25 have a 15% discount of their ticket
        - Students above 25 have a 10% discount off their ticket
        - Seniors (above the age of 64) and are not students have a 7% discount from the cost of their ticket
    - Has a constructor with inputs for Customer and Film **(2 marks)**
    - Is *immutable* but has appropriate get methods **(2 marks)**
    - toString method returns a string representation of the printed ticket with details for customer and film and total cost, with correct decimal format for currency **(6 marks)**

- **CinemaTicketBookingApplication** class **(20 marks in total)**
    - static (e.g. class) variables for: keyboard input, a primitive array of Film objects and a constant *NUMBER_OF_FILMS* that specifies the size of the array **(4 marks)**
    - static main method initialises a primitive array of films (these may be hard-coded) and repeatedly issues tickets in the following manner:
        - invokes the method private static Customer customerDetailsInput() which prompts the user to enter their customer information, constructs and returns a Customer object **(5 marks)**
        - invokes the method private static Film filmSelection() prompting the user to type the number corresponding to the film they want to watch, returning the appropriate Film object **(5 marks)**
        - public static Ticket issueTicket(Customer aCustomer, Film aFilm) which constructs and returns a Ticket object if the customer's age is appropriate to the film they wish to view. If not, null is returned and the customer is advised that the purchase could not be completed. **(2 marks)**
        - The Ticket is issued (e.g. displayed to the console) and the customer is asked if they wish to purchase another ticket **(4 marks)**

Dr. Kenneth Johnson, Auckland University of Technology, kenneth.johnson@aut.ac.nz

Program Interaction

Sample Usage of CinemaTicketBookingApplication:

```
Welcome to the Cinema Ticket Purchasing System
Please enter your name:
Jack Smith
Please enter your age:
20
Are you a student? (Y/N)
Y
Which film would you like to watch:
1 Ant-Man rating: (P)
2 Minions rating: (P)
3 Jurassic World rating: (M)
4 Inside Out rating: (G)
1
Your ticket is ready to be collected!
------------------------------------------------------------
FILM TICKET for: Ant-Man rating: (P)
CUSTOMER DETAILS: Jack Smith age: 20 Student? Yes
TOTAL COST: $8.50
------------------------------------------------------------

Issue another ticket? (Y/N)
Y
Please enter your name:
Holly Day
Please enter your age:
10
Are you a student? (Y/N)
Y
Which film would you like to watch:
1 Ant-Man rating: (P)
2 Minions rating: (P)
3 Jurassic World rating: (M)
4 Inside Out rating: (G)
3
Sorry, cannot issue ticket for age restricted film
Issue another ticket? (Y/N)
N
```

Authenticity

Remember, it is unacceptable to hand in any code which has previously been submitted for assessment (for any paper, including Programming 2), and all work submitted must be unique and your own!

Dr. Kenneth Johnson, Auckland University of Technology, kenneth.johnson@aut.ac.nz

Submission Instructions

Submit the following documents as an archive **`.zip`** file of your documents on AUTOnline before the deadline:

- Your source code (**`.java`** files).
- Sample console output demonstrating your program in use (**`.txt`** file)

Zip structure and file naming requirements. Please ensure your submission matches the following:

- **`lastname-firstname-studentid.zip`**
  - **`*.java`**
  - **`studentid-console-sample.txt`**

Replace the underlined text with your personal details.

Late submissions will receive a grade of 0.

An extension will only be considered with a Special Consideration Form approved by the School Registrar. These forms are available at the School of Computer and Mathematical Science located in the WT Level 1 Foyer.

You will receive your marked assignment via AUTonline. Please look over your entire assignment to make sure that it has been marked correctly. If you have any concerns, you must raise them with the lecturer. You have **one week** to raise any concerns regarding your mark. After that time, your mark cannot be changed.

Do not go to the lecturer because you do not like your mark. Only go if you feel something has been marked incorrectly.

Dr. Kenneth Johnson, Auckland University of Technology, kenneth.johnson@aut.ac.nz