****

**Question 1 (40 Marks)**

The following code defines an abstract superclass called Player, which is contained in a package called Football. The Player class is part of a Football Manager System, which calculates the number of points earned based on goals scored, matches won and matches drawn.

package Football;

public abstract class Player {

String firstName, lastName;

static int totalPoints;

//Number of points awarded to the player when he scores a goal.

static final int goalPoints = 3;

//Number of points awarded to the player when his team wins.

static final int winPoints = 3;

//Number of points awarded to the player when his team draws.

static final int drawPoints = 1;

public Player(String firstName, String lastName) {

this.firstName = firstName;

this.lastName = lastName;

}

public int draws(int gmsDrwn) {

return totalPoints + (drawPoints \* gmsDrwn);

}

public int wins(int gmsWon) {

return totalPoints + (winPoints \* gmsWon);

}

public String toString() {

return firstName +" "+lastName;

}

public abstract int noGoalPointsEarned();

public abstract int totalPointsEarned();

}

}

Write the java code for the following:

1. A class called Striker, which is part of the Football package.

* Striker “is a “ player.
* The Striker class has three member variables, goalsScored, gamesWon and gamesDrawn.
* The Striker class should contain a method, which will calculate the number of goal points earned. NOTE : Goal points are those points gained by a player when he scores a goal. A player earns three points for every goal he scores.
* The Striker class should contain a method to calculate the total number of points earned, this number will be based on the no of points received for goals, games won and games drawn. The total points earned will be cumulative.
* Override the toString() method inherited from Player to display information about the Striker.

(10 Marks)

1. Write the code for a test class, TestFootball. This class should generate two instances of the Striker class

* John Doyle – scored 3 goals, won 4 games, drew 2 games
* Pat Brown - scored 4 goals, won 5 games, drew 1 game
* The class should generate output similar to the following for one striker.
* The name of the top goal scorer should also be displayed.

**Striker’s name is:** John Doyle

**No. of goal points earned :** 9

**Total points earned:**  23

### The Top Goal Scorer is: Pat Brown

(7 Marks)

### Write an interface called FootballPoints that could be implemented in the Strikerclass, instead of inheriting from the Player class.

(7 Marks)

1. What is the importance of Interfaces in Java?

(4 Marks)

1. The following is an example of a nested class, which has been added to the Striker class.

##### Create an instance of an object for the Striker class

1. Create an instance of an object for the No10Shirt class

(7 Marks)

private class No10Shirt

{

private 10ShirtDescription;

No10Shirt(String Description)

{

this.Description = Description;

}

public String toString()

{

}

}

1. Describe two properties of a local class in java.

(5 Marks)

**Question 2 (30 marks)**

The following code defines a class called Department which represents departments within a University. A University contains a number of different departments.

Write the Java code for the following:

**public class** Department {

**private** **int** Deptid;

**private** String DeptName;

**public** Department(**int** dept,String deptN){

Deptid=dept;

DeptName=deptN;

}

**public** **int** getDeptid() {

**return** Deptid;

}

**public** **void** setDeptid(**int** deptid) {

Deptid = deptid;

}

**public** String getDeptName() {

**return** DeptName;

}

**public** **void** setDeptName(String deptName) {

DeptName = deptName;

}

}

1. A class called University with the following implementation details:

* University has two data members: name (String) and the list (ArrayList) of departments within the University.
* A constructor that initialises both data members using a parameter list.
* A showList() method that displays relevant information for each department.

(16 marks)

1. A test class to perform the following tasks:

* Create a University object for a university called UCB that has the following departments:

|  |  |
| --- | --- |
| **Department Name** | **Department ID** |
| Computing | 1001 |
| Business | 1002 |
| Languages | 1003 |
| Engineering | 1004 |

* Display the university name and the relevant information for each department.

( 9 marks)

1. Explain what you understand by the term **"Composition"** in relation to the **University** and **Department** classes.

(5 marks)

**Question 3 (30 marks)**

1. The following code is a Cinema Booking class which, allows users to book cinema tickets and validates the details.

**public** **class** CinemaBooking {

**private** **final** **int** MAXNUM = 6;

**private** **int** ccNo;

**private** **int** noTickets;

**private** String filmName = **new** String("");

**public** CinemaBooking(**int** creditCard,**int** numTickets, String movie)

{

ccNo = creditCard;

noTickets = numTickets;

filmName = movie;

}

}

1. You are required to write the **validateDetails()** method for this class including the following restrictions:

* A user cannot buy more than 6 tickets in one go.
* The film booked must be one from the available list i.e. Pretty Woman or Star Wars.
* In the case where these restrictions are not upheld the system will throw an exception.
* The two exceptions to be used in this problem are TooManyTicketsException() and UnknowFilmException().

1. TooManyTicketsException alerts the cinema that a user has attampted to book more than the maximum allowed tickets i.e 6.
2. UnknowFilmException alerts the cinema that a user has attempted to book a film that is not showing.

* You can assume both of these are already defined.

(10 Marks)

1. Using the code provided below you are to write the cinema test class such that the TooManyTicketsException() and UnknowFilmException() are tried and caught.

(15 Marks)

**import** java.util.Scanner;

**public** **class** CinemaBookingTest {

**public** **static** **void** main(String args[])

{

**int** creditCard = 0;

**int** noTickets = 0;

String film = **new** String("");

Scanner in = **new** Scanner(System.*in*);

System.*out*.println("Please Enter your Credit Card Number");

creditCard = in.nextInt();

System.*out*.println("Please Enter the Number of Tickets you require: ");

noTickets = in.nextInt();

System.*out*.println("Please Enter the Film you wish to see, James Bond or

Star Wars :");

film = in.nextLine();

CinemaBooking MyMovie = **new** CinemaBooking(creditCard,noTickets,film);

}

}

1. Look at the following exception handling code:

**import** java.util.Scanner;

**public** **class** Exceptions {

**public** **static** **void** main(String args[])

{

String name = **new** String("");

Scanner in = **new** Scanner(System.*in*);

**try**

{

System.*out*.println("Enter You name Please");

name = in.nextLine();

}

**catch**(Exception e)

{

e.printStackTrace();

}

**catch** (IOException e)

{

e.printStackTrace();

}

**catch** (InterruptedIOException e)

{

e.printstack();

}

}

}

What is the problem with this code? Justify your answer.

(5 Marks)

**Question 4 (30 marks)**

1. In Java, inner classes can be static or non-static. Using an example, explain how they differ.

(10 marks)

1. What are the purposes of static variables and methods, and when would you use them?

(10 marks)

1. Outline the function of wrapper classes. Using one wrapper class, give an example of how it could be used.

(10 marks)