



eduresourceza.com

WORLD CUP SOCCER 2010

There were a lot of websites on the internet that kept track of the group stage games in the world cup. A copy of the results of group A and B games are stored in the text file called **GroupAB.txt** in the exam common folder on the exam folder on the server. The file consists of four fields separated by the # character: team name, group, overall points and goal difference.

GroupAB.txt

France#A#1#-3
 South Africa#A#4#-2
 Argentina#B#9#6
 Greece#B#3#-3
 Mexico#A#4#1
 South Korea#B#4#-1
 Uruguay#A#7#4
 Nigeria#B#1#-2

OBJECTS AND ARRAYS**79 marks**

Create a new Java project called **SoccerProj** to do the following:

- 1.1 Create a GUI application called **MainUI** where the user can select any of the following menu options:

Menu Options

Display an alphabetically sorted list of all the teams in group A and B
 Display a ranked sorted list (by overall points) of all the teams in group A and B
 Determine the teams to play in the knockout games
 Output list of all teams that won 2 or more games

(4)

- 2.1 Create a **Team** class with instance variables for team name, group, overall points and goal difference. Declare each field using an appropriate type and ensure the fields are not accessible outside the class. (4)

- 2.2 Code the following methods for the Team class:

- 2.2.1 A parameterised constructor. (4)
 2.2.2 Accessor methods for all instance variables. (4)
 2.2.3 Mutator method for the 'group' instance variable. (3)
 2.2.4 **toString** method – which creates and returns a string of all the fields in the class separated by tabs. (4)

- 3.1 Create a class called **TeamAB** to hold all the team information stored in the text file called **GroupAB.txt**. It has an instance array called **teamArr** to store objects of type **Team** and a field called **size** to store the number of teams in the array. You may assume that there will never be more than 32 teams in the array teamArr. (5)

3.2 Code the following methods for the **TeamAB** class:

- 3.2.1 Constructor that initialises the array called **teamArr** using the data in the text file called **GroupAB.txt**. The constructor must read the data from the text file, extract each field and instantiate each element in the array called **teamArr** using the constructor in the **Team** class. The field called **size** must contain the number of elements instantiated in the array. (10)
- 3.2.2 Code a method called **SortAlph** to sort all teams in the array alphabetically by team name (6)
- 3.2.3 Code a method called **SortPts** to sort all teams in the array descending according to overall points (3)
- 3.2.4 Code a method called **Search** that will find the top 2 teams in each group. The method must determine which teams will play each other in the matches 1A vs 2B and 1B vs 2A. 1A means the team coming top in the A group and 2B means the team coming second in group B. The position in the groups is determined by the greater number of points and then by the higher goal difference (if the number of points is the same). (10)
- 3.2.5 Code a method called **allString** that creates a string of all the spies using the **Teams'** **toString** method. Each team should be on a new line. (4)
- 3.2.6 Code a method called **Win** to output a list of all teams that won 2 or more games. Note that a win means 3 points, draw means 1 point and a loss means 0 points. (4)

4. Menu Options

Ensure that you can access the relevant methods from the **TeamAB** class in the **MainUI** class. (2)

4.1 - Display alphabetically sorted list of all teams

Add code in the **MainUI** class/unit to call the **SortAlph** method in the **TeamAB** class and to display the sorted list of spies using the **allString** method in an appropriate component on the screen. (4)

4.2 - Display a ranked sorted list (by overall points) of all the teams in group A and B

Add code in the **MainUI** class/unit to call the **SortPts** method in the **TeamAB** class and to view all the teams information ranked by points using the **allString** method. (4)

4.3 - Display the teams to play each other in the knockout games

Add code in the **MainUI** class/unit to call the **Search** method in the **TeamAB** class and to view the teams in the knockout games. (2)

4.4 - Display list of all teams that won 2 or more games

Add code in the **MainUI** class/unit to call the **Win** method in the **TeamAB** class and to view all the teams that won 2 or more games. (2)

SQL STATEMENTS

41 marks

Migrate the Access database TourInfo in your exam folder to MySQL
Connect Netbeans to the TourInfo MySQL database.

Mr Sileno has organised a tour to East London. He has stored the participants information in a table called Participants. The table has fields to store the flight number to East London, the return flight number, the surname, firstname, ID number, cell phone number and the amount the participant has paid toward the tour.

In a second table called Staff Salaries he has stored the details of staff, their tax code, how much their salary is and the amount of tax they owe based on their salary.

Design View of table **Participants**

PARTICIPANTS	
Field Name	Data Type
FlightNoToEL	Text
FlightNoreturn	Text
Surname	Text
FirstName	Text
ID	Text
CelNo	Text
PaidTourCost	Currency

Design View of table **Staff Salaries**

STAFF SALARIES	
Field Name	Data Type
Surname	Text
FirstName	Text
TaxCode	Text
SalaryToDate	Currency
TaxesToDate	Currency

Sample data of the table **Participants**

PARTICIPANTS						
FlightNoTo	FlightNor	Surname	FirstName	ID	CelNo	PaidTourCo
SA532	BA315	Anelke	Nicolas	881206 5043 085	0766619526	R 700.00
SA532	SA475	Kelly	R	890414 5038 084	0762426044	R 1 200.00
1T101	BA315	Williamson	Graeme	890626 5191 080	0793962850	R 800.00
BA123	SA475	Le Grange	Donald	891222 0072 084	0828906660	R 800.00
SA532	IT201	Van den Bergh	Donovan	891229 5089 088	0828492349	R 1 200.00
SA532	SA475	Van Zyl	Evan	900111 0121 083	0768340154	
1T101	SA475	Slater	Gerald	900213 5373 006	0823032045	R 500.00
SA532	IT201	Samuel	Peter	900326 0199 083	0723216129	R 950.00
BA123	BA315	Stevens	Ettienne	900806 0130 089	0763423525	
1T101	IT201	Steyn	Francois	900826 0083 005	0764190033	R 1 200.00
SA532	IT201	Potgieter	Tanille	901216 0235 087	0761264032	R 1 500.00
BA123	SA475	Du Plessis	Fanie	910130 0251 086	0821340604	R 600.00
1T101	SA475	Laidlaw	Daniel	910215 5052 082	0729549741	R 750.00
SA532	SA475	Terblanche	Eddy	910226 5040 084	0761326564	R 1 000.00
SA532	SA475	Poonsamy	Kevin	910418 5107 084	0764051908	R 1 500.00
SA532	SA475	Edwards	Russel	910621 5008 083	0838977425	
SA532	SA475	Torres	Fernando	910812 5048 084	0825503159	R 800.00
SA532	IT201	Smit	John	910905 0021 088	0726088803	
SA532	SA475	Strydom	Dane	910918 5210 085	0729970328	R 1 000.00
SA532	SA475	Prince	Neville	920116 5204 087	0795764557	R 1 000.00
SA532	SA475	De Villiers	Div	920424 5286 082	0839277637	R 1 250.00
BA123	SA475	Walker	Matthew	920430 5020 082	0764614178	
SA532	SA475	Daddy	Puff	920526 0020 082	0794076535	
BA123	BA315	Barnard	Roy	920925 0392 089	0837601611	R 1 500.00
SA532	BA315	Van Kampen	Fanie	921022 5011 085	0825689216	R 1 000.00
SA532	SA475	Daniel	Mark	930323 5237 087	0794271614	R 1 500.00
SA532	IT201	Nicolas	Ernesto	930432 0030 083	0837333533	R 1 500.00

Sample data of the table **Staff Salaries**

STAFF SALARIES				
Surname	FirstName	TaxCode	SalaryTotat	TaxesToDat
Alba	Jessica	A547261	R 0.00	R 0.00
Anderson	Pamela	A764322	R 0.00	R 0.00
Cruise	Tom	C457234	R 0.00	R 0.00
Deans	Robbie	D110112	R 0.00	R 0.00
Enslin	Kyle	E345267	R 0.00	R 0.00
Fudd	Elmer	F398512	R 0.00	R 0.00
Koekemoer	Vernon	K201000	R 0.00	R 0.00
White	Jake	W692536	R 0.00	R 0.00
*			R 0.00	R 0.00

Write SQL statements in Netbeans and copy the results into the Answer Sheet provided for you in your folder.

1. Alphabetical order of surname then first name. (3)
2. Update the staff salaries so that each staff member receives a salary of R10000, (3)
3. If a staff member has a tax code starting with an A then add R5000 to the staff member's salary. (3)
4. Each staff member will pay 25% tax. Update the TaxesToDate field of each staff member so that their tax amount is 25% of their salary. (3)
5. Display all the participants who are listed on a BA flight either going to or returning from East London. (4)
6. Show all the details for Mark Daniels. (3)
7. Generate a code for each participant by taking their initial, first three letters of their surname and last three digits of their ID number. Name this new field Code. (6)
8. Display all the participants who have not paid. (2)
9. Delete Fanie Du Plessis from the Participants table. (3)
10. Count the number of people who have paid. (4)
11. Find the person who has paid the most. (3)
12. Find the average amount each person has paid per flight to East London. (4)