



eduresourceza.com



ST BENEDICT'S

SUBJECT

**INFORMATION
TECHNOLOGY**

PAPER

PRACTICAL

GRADE

11

DATE

JULY 2019

EXAMINER

D KENCH

MARKS

120

NAME

MODERATOR

TOOTH FAIRY

CLASS

DURATION

3 HOUR + 30 MINS
PRINTING

Question	1	2	3	4	5	6	Total
Possible Marks	40	29	27	5	14	5	120
Actual Marks							

SECTION A

SQL

40 MARKS

Parkrun organise free, weekly, 5km timed runs around the world. They are open to everyone, free, and are safe and easy to take part in. They keep records of the events and all members. St Benedict's has hosted four park runs and recorded the results. As an IT student you will need to process the data stored in table called **tblResults** in the database called **ParkRun**.

```
CREATE TABLE `tblResults` (
  `fDate` DATE,
  `Code` VARCHAR(255),
  `FullName` VARCHAR(255),
  `fTime` TIME,
  `Category` VARCHAR(255),
  INDEX (`Code`)
);
```

QUESTION 1

- 1.1 Write a SQL statement to sort the table **tblResults** according to date and then the finishing time. The first 20 rows of your output should appear as follows:

#	fDate	Code	FullName	fTime	Category
1	2019-06-15	504B	Katlego MATLABE	00:17:40	Senior
2	2019-06-15	396B	Anet COETZEE	00:17:58	Senior
3	2019-06-15	199B	John FRANKLIN	00:18:06	Senior
4	2019-06-15	2093E	Teboko SIAANI	00:18:24	Grand Master
5	2019-06-15	876D	Reegan ELLIS	00:18:37	Master
6	2019-06-15	1080A	Trish BAHLMANN	00:18:42	Junior
7	2019-06-15	684C	Martinique POTGIETER	00:18:47	Master
8	2019-06-15	1032B	Tayla KAVANAGH	00:19:28	Senior
9	2019-06-15	103D	Mavis JOHNSON	00:19:55	Master
10	2019-06-15	120B	Anika MEYER	00:25:01	Senior
11	2019-06-15	432A	Patrice NKOSI	00:27:19	Junior
12	2019-06-15	972A	Siphokazi MAPHOYI	00:28:35	Junior
13	2019-06-15	1092C	Tseleng SUPING	00:53:09	Veteran
14	2019-06-15	948B	Shelley YOUNG	00:53:27	Senior
15	2019-06-15	420B	Hanlie BOTHA	00:54:12	Senior
16	2019-06-15	1044E	Thalia SLATEM	01:10:23	Grand Master
17	2019-06-15	372C	Enie MANZINI	01:10:57	Veteran
18	2019-06-15	720B	Melanie DU PLESSIS	01:12:21	Senior
19	2019-06-22	1032B	Tayla KAVANAGH	00:18:19	Senior
20	2019-06-22	504B	Katlego MATLABE	00:18:32	Senior

(4)

- 1.2 Write a SQL statement to display the name(s) of all the "Masters" or "Grand Masters" that have a result that is greater than an hour. Do not show any duplicate names. Your output should be displayed as follows:

FullName
Thalia SLATEM

(6)

- 1.3 Write a SQL statement to count all the "Junior" runners who ran on 22 June 2019. Rename your new field **NumJuniors**. Your output should be displayed as follows:

NumJuniors
3

(5)

- 1.4 All codes should consist of 5 characters. Write a SQL statement to add a 0 to the beginning of any 4 character codes. Once the SQL statement is run, a sample of the first 10 records of the altered table **tblResults** is shown below:

#	fDate	Code	FullName	fTime	Category
1	2019-06-15	1080A	Trish BAHLMANN	00:18:42	Junior
2	2019-06-15	0420B	Hanlie BOTHA	00:54:12	Senior
3	2019-06-15	0396B	Anet COETZEE	00:17:58	Senior
4	2019-06-15	0720B	Melanie DU PLESSIS	01:12:21	Senior
5	2019-06-15	0876D	Reegan ELLIS	00:18:37	Master
6	2019-06-15	0199B	John FRANKLIN	00:18:06	Senior
7	2019-06-15	0103D	Mavis JOHNSON	00:19:55	Master
8	2019-06-15	1032B	Tayla KAVANAGH	00:19:28	Senior
9	2019-06-15	0372C	Ernie MANZINI	01:10:57	Veteran
10	2019-06-15	0972A	Siphokazi MAPHOYI	00:28:35	Junior

(5)

- 1.5 Write a SQL statement to find the winning time for each category. Sort the result from the shortest to the longest time.

Category	WinningTimes
Senior	00:17:40
Grand Master	00:18:24
Master	00:18:37
Junior	00:18:42
Veteran	00:21:07

(5)

- 1.6 Write a SQL statement to calculate the average time in seconds for each date. Convert each runner's time to seconds by multiplying the hours by 3600, the minutes by 60 and the adding these values to the seconds.

Date	AveTimeInSecs
15/06/2019	2076.7222
22/06/2019	1314.0556
29/06/2019	1897.6667

(7)

- 1.7 Write a SQL statement to count all the runners from each category. Only display the categories that have more than 5 runners.

NumCat	
	6
	9
	9
	24
	6

(5)

- 1.8 Write a SQL statement to insert a record for a runner using your name, the time of 18min and 18 seconds and today's date. Use an appropriate date function to determine today's date.

(3)

SECTION B

OOP

40 MARKS

A program needs to process the results of the results. A sample of the data has been extracted from the database and stored in a text file called **Results.txt** in the format runner code, runner name, finish time. The code of each runner ends with a letter indicating their age category. A = "Junior", B = "Senior", C = "Veteran", D = "Master" and E = "Grand Master". The data in the file is shown below:

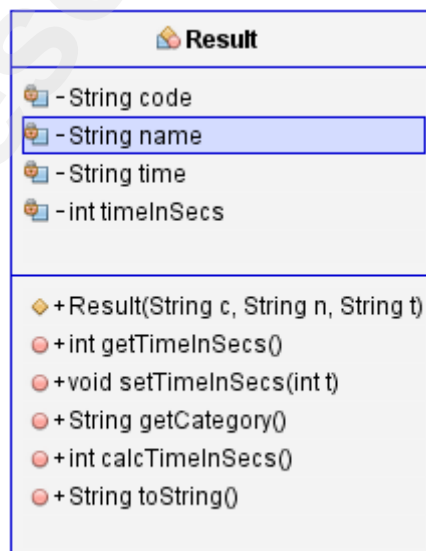
code#name#time#

```
1080A#Trish BAHLMANN#00:19:28#
420B#Hanlie BOTHA#00:18:19#
96B#Anet COETZEE#00:21:08#
720B#Melanie DU PLESSIS#00:19:55#
876D#Reegan ELLIS#00:19:18#
199B#John FRANKLIN#00:40:52#
103D#Mavis JOHNSON#01:10:23#
1032B#Tayla KAVANAGH#00:25:37#
372C#Enie MANZINI#00:22:01#
972A#Siphokazi MAPHOYI#00:18:42#
504B#Katlego MATLABE#00:21:07#
120B#Anika MEYER#00:24:37#
432A#Patrice NKOSI#00:53:09#
684C#Martinique POTGIETER#00:21:07#
2093E#Teboko SIAANI#01:02:01#
1044E#Thalia SLATEM#00:45:01#
1092C#Tseleng SUPING#00:23:52#
948B#Shelley YOUNG#00:19:21#
```

QUESTION 2

29 MARKS

Study the **Result** Class Diagram.



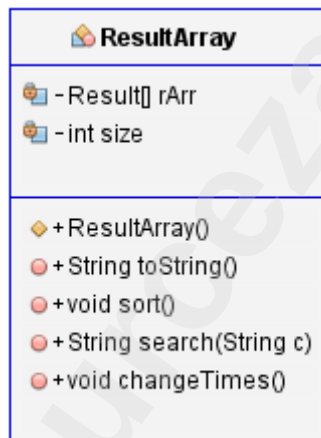
- 2.1 Create a class called **Result** with private fields to store the **code**, runner's full **name**, the **time** they crossed the finish line. and an integer called **timeInSecs** to store the time in seconds. (4)
- 2.2 Create a parameterised constructor to assign values to the **code**, **name** and **time** fields. (2)

- 2.3 Code a typed method called **calcTimeInSecs** to convert the time into seconds by multiplying the hours by 3600, the minutes by 60 and adding them to the seconds. Call this method from the constructor method to assign a value to the **timeInSecs** field. (8)
- 2.4 Code an accessor and mutator method for the **timeInSecs** field. (2)
- 2.5 Code a typed method called **getCategory** to extract the last character from the code and convert to the age category. A = "Junior", B = "Senior", C = "Veteran", D = "Master" and E = "Grand Master". The category description must be returned as a string. Marks will be awarded for efficient use of programming structures. (9)
- 2.6 Code a **toString** to return the fields code, name, time together with the time in seconds and the category separated by tabs. (4)

QUESTION 3

27 MARKS

Using the class diagram, code the **ResultArray** class to store many **Result** objects. The **search** and **changeTimes** methods will be coded in Question 5.



- 3.1 Add code to store 100 **Result** objects in an array called **rArr** and a field called **size** to record the number of objects in the array. (4)
- 3.2 Code a constructor method to instantiate the array of results using the text file **Results.txt**. The method must:
 - read in each line of the text file
 - separate each line into the code, name and time fields
 - instantiate an object of **rArr**
 - increment **size**
 - display a message if the text file does not exist (10)
- 3.3 Code a **toString** method to display each **Result** object in the format of the **toString** of the **Result** class. Each **Result** object must appear on a new line. (5)
- 3.4 Code a method called **sort** to sort the array of **Result** objects in ascending order of the time using the **calcTimeInSecs** method you created in 2.3. (8)

QUESTION 4**5 MARKS**

- 4.1 Code a class called **ResultUI**. (1)
- 4.2 Add code to instantiate a **ResultArray** object called **ra**. (1)
- 4.3 Add code to display all the runners using the **toString** method of the **ResultArray** class.

Your output should be as follows:

Trish BAHLMANN	1080A	00:19:28	1168	Junior	
Hanlie BOTHA	420B	00:18:19	1099	Senior	
Anet COETZEE	96B	00:21:08	1268	Senior	
Melanie DU PLESSIS	720B	00:19:55	1195	Senior	
Reegan ELLIS	876D	00:19:18	1158	Master	
John FRANKLIN	199B	00:40:52	2452	Senior	
Mavis JOHNSON	103D	01:10:23	4223	Master	
Tayla KAVANAGH	1032B	00:25:37	1537	Senior	
Enie MANZINI	372C	00:22:01	1321	Veteran	
Siphokazi MAPHOYI	972A	00:18:42	1122	Junior	
Katlego MATLABE	504B	00:21:07	1267	Senior	
Anika MEYER	120B	00:24:37	1477	Senior	
Patrice NKOSI	432A	00:53:09	3189	Junior	
Martinique POTGIETER	684C	00:21:07	1267	Veteran	
Teboko SIAANI	2093E	01:02:01	3721	Grand Master	
Thalia SLATEM	1044E	00:45:01	2701	Grand Master	
Tseleng SUPING	1092C	00:23:52	1432	Veteran	
Shelley YOUNG	948B	00:19:21	1161	Senior	

(1)

- 4.4 Add code to sort the array and then display the sorted array.

Your output should be as follows:

Hanlie BOTHA	420B	00:18:19	1099	Senior	
Siphokazi MAPHOYI	972A	00:18:42	1122	Junior	
Reegan ELLIS	876D	00:19:18	1158	Master	
Shelley YOUNG	948B	00:19:21	1161	Senior	
Trish BAHLMANN	1080A	00:19:28	1168	Junior	
Melanie DU PLESSIS	720B	00:19:55	1195	Senior	
Martinique POTGIETER	684C	00:21:07	1267	Veteran	
Katlego MATLABE	504B	00:21:07	1267	Senior	
Anet COETZEE	96B	00:21:08	1268	Senior	
Enie MANZINI	372C	00:22:01	1321	Veteran	
Tseleng SUPING	1092C	00:23:52	1432	Veteran	
Anika MEYER	120B	00:24:37	1477	Senior	
Tayla KAVANAGH	1032B	00:25:37	1537	Senior	
John FRANKLIN	199B	00:40:52	2452	Senior	
Thalia SLATEM	1044E	00:45:01	2701	Grand Master	
Patrice NKOSI	432A	00:53:09	3189	Junior	
Teboko SIAANI	2093E	01:02:01	3721	Grand Master	
Mavis JOHNSON	103D	01:10:23	4223	Master	

(2)

QUESTION 5**14 MARKS**

Return to the **ResultArray** class to complete the **search** and **changeTimes** methods listed in the class diagram.

- 5.1 Code a method called **search** to find all the **Results** in the Result array **ra**, given the category as a string. The method must return each result, of the given category, on a separate line. (6)
- 5.2 The organisers realise that they do not have any medals for the "Grand Master" category. They decide to deduct 10 mins (600 secs) from each Grand Master's time. Code a method called **changeTimes** to reduce the **timeInSecs** field for each Grand Master in the array called **rArr** by 600. Use any methods you have previously coded in your method. (8)

QUESTION 6**5 MARKS**

Return to the **ResultUI** class to call the methods you called in Question 5.

- 6.1 Add code to search for all "Seniors" and display the result using the method you coded in 5.1.

Your output should be as follows:

Hanlie BOTHA	420B	00:18:19	1099	Senior	
Shelley YOUNG	948B	00:19:21	1161	Senior	
Melanie DU PLESSIS	720B	00:19:55	1195	Senior	Senior
Katlego MATLABE	504B	00:21:07	1267	Senior	
Anet COETZEE	96B	00:21:08	1268	Senior	
Anika MEYER	120B	00:24:37	1477	Senior	
Tayla KAVANAGH	1032B	00:25:37	1537	Senior	
John FRANKLIN	199B	00:40:52	2452	Senior	

(2)

- 6.2 Add code to change the times of all the "Grand Master"s, then sort the array again and display the results.

Your output should be as follows – note that the time field does not change:

Hanlie BOTHA	420B	00:18:19	1099	Senior	
Siphokazi MAPHOYI	972A	00:18:42	1122	Junior	
Reegan ELLIS	876D	00:19:18	1158	Master	
Shelley YOUNG	948B	00:19:21	1161	Senior	
Trish BAHLMANN	1080A	00:19:28	1168	Junior	
Melanie DU PLESSIS	720B	00:19:55	1195	Senior	Senior
Martinique POTGIETER	684C	00:21:07	1267	Veteran	
Katlego MATLABE	504B	00:21:07	1267	Senior	
Anet COETZEE	96B	00:21:08	1268	Senior	
Enie MANZINI	372C	00:22:01	1321	Veteran	
Tseleng SUPING	1092C	00:23:52	1432	Veteran	
Anika MEYER	120B	00:24:37	1477	Senior	
Tayla KAVANAGH	1032B	00:25:37	1537	Senior	
Thalia SLATEM	1044E	00:45:01	2101	Grand Master	
John FRANKLIN	199B	00:40:52	2452	Senior	
Teboko SIAANI	2093E	01:02:01	3121	Grand Master	
Patrice NKOSI	432A	00:53:09	3189	Junior	
Mavis JOHNSON	103D	01:10:23	4223	Master	

(3)