

GLA UNIVERSITY, MATHURA

DATABASE MANAGEMENT SYSTEM

ASSIGNMENT

Submitted to -

Abhishek Sharma Sir

Submitted by-

Ayushi Singh

2415000426

SC

15

QUERY -1

List the student's name, dob from student table

ASSIGNMENT 1_2415000893.sql* ▶ ⏪ ⏴ Aa ⌂

```
27     insert into Student (sID , sName ,GPA , sizeHS , DoB )values (543 , 'C
28
29
30
31     SELECT sName , Dob from Student ;
32
33     SELECT sName from Student where GPA > 3.7;
```

Query result Script output DBMS output Explain Plan SQL history

>Delete Download Execution time: 0.004 seconds

	SNAME	DOB
5	Amy	6/26/1996, 12:00:00
6	Bob	4/7/1995, 12:00:00
7	Craig	2/4/1995, 12:00:00
8	Doris	7/24/1997, 12:00:00
9	Edward	12/21/1996, 12:00:00
10	Fay	8/27/1996, 12:00:00
11	Gary	10/8/1996, 12:00:00
12	Helen	3/27/1997, 12:00:00
13	Irene	3/7/1996, 12:00:00
14	Jay	8/8/1998, 12:00:00

QUERY-2

List the name of student scoring more than 3.7 in GPA.

The screenshot shows a SQL query editor interface. At the top, there is a toolbar with various icons. Below the toolbar, the file name "ASSIGNMENT 1_2415000893.sql*" is displayed. The main area contains a code editor with the following SQL queries:

```
30
31  SELECT sName , Dob from Student ;
32
33  SELECT sName from Student where GPA > 3.7;
34
35  SELECT sName from Student where sizeHS >=1000 AND Dob > '31-DEC-1996';
36
37  SELECT sName from Student where GPA BETWEEN 2.9 AND 3.9;
```

Below the code editor, there are tabs for "Query result", "Script output", "DBMS output", "Explain Plan", and "SQL history". The "Query result" tab is selected, showing a table with the column "SNAME". The results are as follows:

	SNAME
1	Amy
2	Amy
3	Amy
4	Doris
5	Fay
6	Irene
7	Amy

At the bottom left of the results table, there are icons for trash, info, and download, followed by the text "Execution time: 0.002 seconds".

QUERY –3

List the name of student whose High School size is at least 1000 and born after 1996

The screenshot shows a SQL query editor interface. At the top, there is a toolbar with icons for file operations and a dropdown menu showing 'ASSIGNMENT 1_2415000893.sql*'. Below the toolbar is a code editor window containing the following SQL code:

```
32  SELECT sName from Student where GPA > 3.7;
33  SELECT sName from Student where sizeHS >=1000 AND Dob > '31-DEC-1996';
34
35
36
37  SELECT sName from Student where GPA BETWEEN 2.9 AND 3.9;
38
39
```

Below the code editor, there are tabs for 'Query result', 'Script output', 'DBMS output', 'Explain Plan', and 'SQL history'. The 'Query result' tab is selected, showing the following table:

	SNAME
1	Doris
2	Jay
3	Craig

At the bottom of the interface, there are buttons for 'Delete', 'Info', 'Download', and a message indicating an execution time of '0.001 seconds'.

QUERY –4

List the name of student who are scoring GPA in between 2.9 and 3.9

The screenshot shows a SQL query editor window with the following details:

- File:** ASSIGNMENT 1_2415000893.sql*
- Query:** (lines 35-39)

```
34
35     SELECT sName from Student where sizeHS >=1000 AND Dob > '31-DEC-1996'
36
37     SELECT sName from Student where GPA BETWEEN 2.9 AND 3.9;
38
39     SELECT * FROM College where State = 'MA'
```
- Execution Time:** 0.003 seconds
- Results:** A table titled "SNAME" showing student names.

	SNAME
5	Amy
6	Bob
7	Craig
8	Doris
9	Edward
10	Fay
11	Gary
12	Helen
13	Irene
14	Jay

QUERY-5

List all the details of colleges situated in MA.

ASSIGNMENT 1_2415000893.sql* ▾ ▶ ⏪ ⏴ ⏵ ⏷ Aa ⏹

```

37  SELECT sName from Student where GPA BETWEEN 2.9 AND 3.9;
38
39  SELECT * FROM College where State = 'MA'
40
● 41  SELECT * from Student where GPA >2.0 AND GPA <3.5;
42

```

Query result Script output DBMS output Explain Plan SQL history

Download Execution time: 0.002 seconds

	CNAME	STATE	ENROLLMENT
1	MIT	MA	10000
2	Harvard	MA	50040
3	MIT	MA	10000
4	Harvard	MA	50040
5	MIT	MA	10000
6	Harvard	MA	50040
7	MIT	MA	10000
8	Harvard	MA	50040

QUERY-6

List the students who are scored more than 2.0 but less than 3.5.

The screenshot shows a SQL developer interface with the following details:

- Query Editor:** The current file is "ASSIGNMENT 1_2415000893.sql*". The code contains two queries:

```
39  SELECT * FROM College where State = 'MA'  
40  
● 41  SELECT * from Student where GPA >2.0 AND GPA <3.5;  
42  
43
```
- Execution Results:** The results of the second query are displayed in a table:

	SID	SNAME	GPA	SIZEHS	DOB
1	567	Edward	2.9	2000	12/21/1996, 12:00:00
2	789	Gary	3.4	800	10/8/1996, 12:00:00
3	765	Jay	2.9	1500	8/8/1998, 12:00:00
4	543	Craig	3.4	2000	8/27/1998, 12:00:00
- Toolbars and Buttons:** Standard SQL developer toolbar buttons for executing queries, refreshing, and navigating.
- Right Panel:** Shows a small preview of the table data.

QUERY-7

List the students who have born after 1st Jul 96 in the order of the Date of Birth.

ASSIGNMENT 1_2415000893.sql* ▶ ⏪ ⏴ ⏵ Aa

```

39  SELECT * FROM College where State = 'MA'
40
41  SELECT * from Student where GPA >2.0 AND GPA <3.5;
42
43  SELECT *from Student where Dob >'01-JUL-1996' order by Dob;
44

```

Query result Script output DBMS output Explain Plan SQL history

Download Execution time: 0.005 seconds

	SID	SNAME	GPA	SIZEHS	DOB
1	678	Fay	3.8	200	8/27/1996, 12:00:00
2	789	Gary	3.4	800	10/8/1996, 12:00:00
3	567	Edward	2.9	2000	12/21/1996, 12:00:00
4	987	Helen	3.7	800	3/27/1997, 12:00:00
5	456	Doris	3.9	1000	7/24/1997, 12:00:00
6	765	Jay	2.9	1500	8/8/1998, 12:00:00
7	543	Craig	3.4	2000	8/27/1998, 12:00:00

QUERY – 8

List the sID , cName , decision of application that are accepted.

ASSIGNMENT 1_2415000893.sql* ▾ ▶ ⌂ ⌄ ⌅

```
95
96     SELECT major from APPLY ;
97
98     SELECT sID , cName , decision from APPLY where decision = 'Y'
```

Query result Script output DBMS output Explain Plan SQL history

>Delete Download ▾ Execution time: 0.005 seconds

	SID	CNAME	DECISION
1	123	Berkeley	Y
2	123	Cornell	Y
3	123	MIT	Y
4	123	Cornell	Y
5	123	Berkeley	Y
6	123	MIT	Y
7	123	Cornell	Y
8	123	Berkeley	Y
9	123	Cornell	Y
10	123	MIT	Y
11	123	Cornell	Y

QUERY – 9

List the sID , cName of applications which are filled by Stanford

ASSIGNMENT 1_2415000893.sql*

```
98  SELECT sID , cName , decision from APPLY where decision = 'Y';
99
100 SELECT sID,cName FROM APPLY where cName = 'Stanford';
```

Query result Script output DBMS output Explain Plan Schema browser



Download ▾

Execution time: 0.002 seconds

	SID	CNAME
1	123	Stanford
2	123	Stanford
3	123	Stanford
4	123	Stanford
5	123	Stanford
6	123	Stanford

QUERY – 10

List the colleges that have enrollment greater than 10001.

ASSIGNMENT 1_2415000893.sql* ▾ ▶ ⏪ ⏴ ⏵ ⏷ Aa ⏹

```
42
43  SELECT *from Student where Dob >'01-JUL-1996' order by Dob;
44
45  SELECT cName from College where enrollment > 10001;
46
47
```

Query result Script output DBMS output Explain Plan SQL history

Download ▾ Execution time: 0.008 seconds

	CNAME
1	Stanford
2	Berkeley
3	Cornell
4	Harvard
5	Stanford
6	Berkeley
7	Cornell
8	Harvard
9	Stanford
10	Berkeley
11	Cornell

QUERY – 11

List the colleges not in California

The screenshot shows a SQL query editor interface. At the top, there is a file menu with "ASSIGNMENT 1_2415000893.sql*" and various icons for running, saving, and deleting the script. Below the menu is the SQL code:

```
43  SELECT *from Student where Dob >'01-JUL-1996' order by Dob;
44
45  SELECT cName from College where enrollment > 10001;
46
47  SELECT cName from College where state !='CA';
48
```

Below the code, there are tabs for "Query result", "Script output", "DBMS output", "Explain Plan", and "SQL history". The "Query result" tab is selected, showing the following table:

	CNAME
1	MIT
2	Cornell
3	Harvard
4	MIT
5	Cornell
6	Harvard
7	MIT
8	Cornell
9	Harvard
10	MIT
11	Cornell

At the bottom of the results table, it says "Execution time: 0.004 seconds".

QUERY – 12

List names of all students who came from high school having size greater than 17000 and scored GPA less than 3.8.

The screenshot shows a SQL query editor interface. At the top, there is a toolbar with icons for file operations like save, open, and delete, and a font size dropdown. Below the toolbar, the title of the script is "ASSIGNMENT 1_2415000893.sql*". The code area contains the following SQL statements:

```
47  SELECT cName from College where state !='CA';
48
49  SELECT sName from Student where sizeHS >1700 and GPA <3.8;
50
51  DESCRIBE Student;
52
53  SELECT * FROM Student ;
```

The code editor has line numbers on the left and syntax highlighting for SQL keywords and identifiers. Below the code editor is a horizontal scroll bar. At the bottom of the interface, there is a navigation bar with tabs: "Query result" (which is selected and highlighted in green), "Script output", "DBMS output", "Explain Plan", and "SQL history". Under the "Query result" tab, there is a message bar with a trash icon, an info icon, a "Download" button with a dropdown arrow, and the text "Execution time: 0.066 seconds". Below the message bar is a table with two rows. The table has a single column header "SNAME" and two data rows, each containing a number (1 and 2) and a student name (Edward and Craig). The table is styled with alternating row colors.

	SNAME
1	Edward
2	Craig

QUERY-13

Display the description of the Student table.

ASSIGNMENT 1_2415000893.sql*

```
48
49     SELECT sName from Student where sizeHS >17000 and GPA <3.8;
50
51     DESCRIBE Student;
52
53
```

Query result Script output DBMS output Explain Plan SQL history

X ↓

SNAME

Amy
Amy
Amy
Doris
Fay
Irene
Amy

Elapsed: 00:00:00.005

7 rows selected.

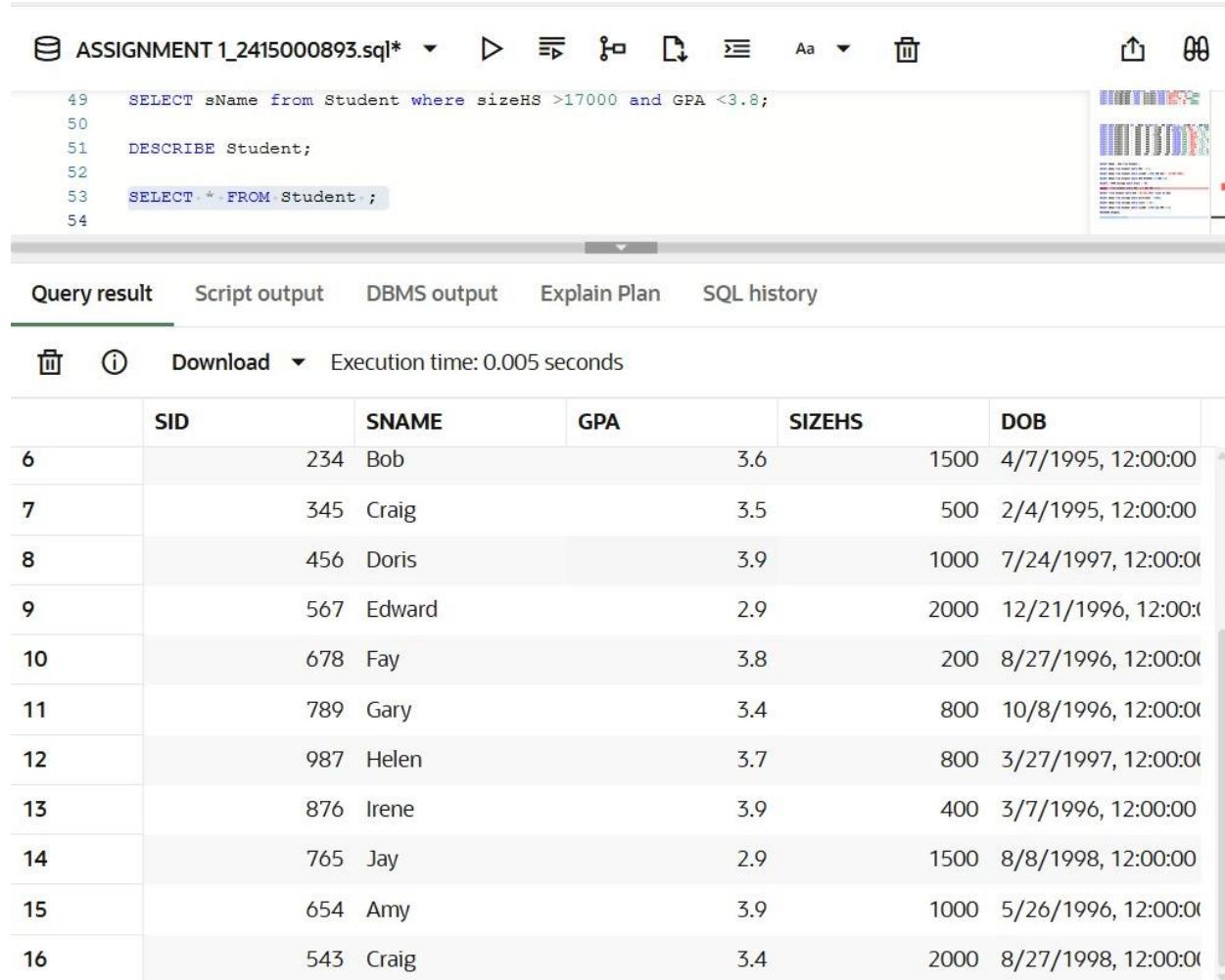
SQL> DESCRIBE Student

Name Null? Type

SID	NUMBER(38)
SNAME	VARCHAR2(10)
GPA	NUMBER(2,1)
SIZEHS	NUMBER(38)
DOB	DATE

QUERY-14

Display the details of all students.



The screenshot shows a SQL developer interface with the following details:

- Script Area:** Shows the SQL code:

```
49  SELECT sName from Student where sizeHS >17000 and GPA <3.8;
50
51  DESCRIBE Student;
52
53  SELECT * FROM Student ;
54
```
- Execution Result:** A table with 16 rows of student data. The columns are SID, SNAME, GPA, SIZEHS, and DOB.
- Table Data:**

	SID	SNAME	GPA	SIZEHS	DOB
6	234	Bob	3.6	1500	4/7/1995, 12:00:00
7	345	Craig	3.5	500	2/4/1995, 12:00:00
8	456	Doris	3.9	1000	7/24/1997, 12:00:00
9	567	Edward	2.9	2000	12/21/1996, 12:00:00
10	678	Fay	3.8	200	8/27/1996, 12:00:00
11	789	Gary	3.4	800	10/8/1996, 12:00:00
12	987	Helen	3.7	800	3/27/1997, 12:00:00
13	876	Irene	3.9	400	3/7/1996, 12:00:00
14	765	Jay	2.9	1500	8/8/1998, 12:00:00
15	654	Amy	3.9	1000	5/26/1996, 12:00:00
16	543	Craig	3.4	2000	8/27/1998, 12:00:00

QUERY-15

Display unique majors .

ASSIGNMENT 1_2415000893.sql* ▶ ⌂ [

```
92 UPDATE student SET GPA = GPA+1.5 where GPA<3.5 and
93
94 DELETE from Student where GPA < 3.2;
95
96 SELECT major from APPLY ;
```

Query result Script output DBMS output Explain Plan



Download ▾

Execution time: 0.005 seconds

	MAJOR
1	CS
2	EE
3	biology
4	bioengineering
5	bioengineering
6	CS
7	EE
8	CS
9	biology
10	marine biology
11	history

QUERY – 16

List the student names those are having 3 characters in their names.

ASSIGNMENT 1_2415000893.sql* ▾ ▶ ⏪ ⏴ ⏵

```
51  DESCRIBE Student;
52
53  SELECT * FROM Student ;
54
55  SELECT sName from Student where sName LIKE '_ _ _';
56
57
```

Query result Script output DBMS output Explain Plan



Download ▾

Execution time: 0.003 seconds

	SNAME
1	Amy
2	Bob
3	Amy
4	Bob
5	Fay
6	Jay
7	Amy
8	Amy
9	Bob
10	Amy

QUERY – 17

List the student names those are starting with ‘H’ and with 5 characters.



ASSIGNMENT 1_2415000893.sql*



```
51   DESCRIBE Student;
52
53   SELECT * FROM Student ;
54
55   SELECT sName from Student where sName LIKE '_ _';
56
57   SELECT sName from Student where sName LIKE 'H_ _';
```

[Query result](#)[Script output](#)[DBMS output](#)[Explain Plan](#)[S](#)

Download

Execution time: 0.003 seconds

[Show statement](#)

1	Helen
2	Helen
3	Helen

QUERY – 18

List the student names those are having third character and fifth character must be ‘e’.



ASSIGNMENT 1_2415000893.sql*



```
55    SELECT sName from Student where sName LIKE '____';  
56  
57    SELECT sName from Student where sName LIKE 'H____';  
58  
59    SELECT sName from Student where sName LIKE '__e__e';
```

[Query result](#)[Script output](#)[DBMS output](#)[Explain Plan](#)[S](#)[Download](#)

Execution time: 0.002 seconds

	SNAME
1	Irene
2	Irene
3	Irene

QUERY – 19

List the student names ending with ‘y’

ASSIGNMENT 1_2415000893.sql* ▶ ⌂ ⌂

```
55  SELECT sName from Student where sName LIKE '_ ___';
56
57  SELECT sName from Student where sName LIKE 'H ___';
58
59  SELECT sName from Student where sName LIKE '__e_e';
60
61  SELECT sName from Student where sName LIKE '%y';
```

Query result Script output DBMS output Explain Plan

Trash Download Execution time: 0.004 seconds

	SNAME
2	Amy
3	Fay
4	Gary
5	Jay
6	Amy
7	Amy
8	Amy
9	Fay
10	Gary
11	Jay

QUERY – 20

List the student in the order of their GPA .

ASSIGNMENT 1_2415000893.sql*

```
57  SELECT sName from Student where sName LIKE 'H____';  
58  
59  SELECT sName from Student where sName LIKE '__e_e';  
60  
61  SELECT sName from Student where sName LIKE '%y';  
62  
63  SELECT sName from Student order by GPA;
```

Query result Script output DBMS output Explain Plan



Download ▾

Execution time: 0.005 seconds

	SNAME
1	Edward
2	Jay
3	Jay
4	Edward
5	Edward
6	Jay
7	Gary
8	Gary
9	Gary
10	Craig

QUERY – 21

List the details of the students in order of the ascending of GPA and descending of DoB.

The screenshot shows a SQL developer interface with the following details:

- Script Area:** Contains the following SQL code:

```
59  SELECT sName from Student where sName LIKE '__e_e';
60
61  SELECT sName from Student where sName LIKE '%y';
62
63  SELECT sName from Student order by GPA;
64
65  SELECT * from Student order by GPA, DoB DESC;
```
- Execution Results:** The results tab shows the output of the last query:

	SID	SNAME	GPA	SIZEHS	DOB
3	765	Jay	2.9	1500	8/8/1998, 12:00:00
4	567	Edward	2.9	2000	12/21/1996, 12:00:00
5	567	Edward	2.9	2000	12/21/1996, 12:00:00
6	567	Edward	2.9	2000	12/21/1996, 12:00:00
7	543	Craig	3.4	2000	8/27/1998, 12:00:00
8	543	Craig	3.4	2000	8/27/1998, 12:00:00
9	543	Craig	3.4	2000	8/27/1998, 12:00:00
10	789	Gary	3.4	800	10/8/1996, 12:00:00
11	789	Gary	3.4	800	10/8/1996, 12:00:00
12	789	Gary	3.4	800	10/8/1996, 12:00:00

QUERY – 22

List the sIDs of student who apply in either ‘Stanford’ , ‘Cornell’ or ‘MIT’ College .

ASSIGNMENT 1_2415000893.sql* ▶ ⏪ ⏴ ⏵ Aa

```
82  SELECT sID, cName, decision from Apply;
83
84  SELECT sID from APPLY where cName = 'CORNEL' , 'MIT' , .'STA FORD' ;
85
```

Query result Script output DBMS output Explain Plan SQL history

>Delete Download Execution time: 0.004 seconds

	SNAME	DOB
7	Edward	12/21/1996, 12:00:00
8	Fay	8/27/1996, 12:00:00
9	Gary	10/8/1996, 12:00:00
10	Helen	3/27/1997, 12:00:00
11	Irene	3/7/1996, 12:00:00
12	Jay	8/8/1998, 12:00:00
13	Amy	5/26/1996, 12:00:00
14	Craig	8/27/1998, 12:00:00
15	Amy	6/26/1996, 12:00:00
16	Bob	4/7/1995, 12:00:00
17	Craig	2/4/1995, 12:00:00

QUERY – 23

Delete all applications filled at Stanford.

```
ASSIGNMENT 1_2415000893.sql* ▶ ⏷ ⏸ ⏹ ⏺ ⏻
```

82 SELECT sID, cName, decision from Apply;
83
84 SELECT sID from APPLY where cName = 'CORNEL' , 'MIT' , . 'S'
85
86 DELETE from APPLY .. where cName = 'Stanford';
87

Query result Script output **DBMS output** Explain Plan SQL h

*Action: Take the action that corresponds with the Cause
1. Check that your SQL statement has no typos.
2. Check Oracle Database documentation to find the correct syntax for the clause and update the problematic clause appropriately.
3. Check Oracle Database documentation to find the correct syntax for the statement and remove the unsupported clause.
4. Enter two single quotes instead of one to represent an apostrophe within a string.
*Params:
1) keyword_value
keyword near the keyword causing the error. The keyword value may be truncated for readability if it is too long.
2) previous_keyword_values: a sequence of keyword values that immediately precede the keyword_value in the command. The sequence of previous keyword values may be truncated if it is too long.
Error at Line: 1 Column: 44

SQL> DELETE from APPLY where cName = 'Stanford'

18 rows deleted.
Elapsed: 00:00:00.006

QUERY – 24

Delete the college Stanford from college table.

```
ASSIGNMENT 1_2415000893.sql* ▾ ▶ ⏷ ⏸ ⏹
```

84 SELECT sID from APPLY where cName = 'CORNEL' , 'MIT'
85
86 DELETE from APPLY where cName = 'Stanford';
87
88 DELETE from College where cName = 'Stanford';
89

Query result Script output DBMS output Explain Plan

Download ▾ Execution time: 0.004 seconds

	SNAME	DOB
1	Amy	6/26/1996, 12:00:00
2	Bob	4/7/1995, 12:00:00
3	Amy	6/26/1996, 12:00:00
4	Bob	4/7/1995, 12:00:00
5	Craig	2/4/1995, 12:00:00
6	Doris	7/24/1997, 12:00:00
7	Edward	12/21/1996, 12:00:00
8	Fay	8/27/1996, 12:00:00
9	Gary	10/8/1996, 12:00:00
10	Helen	3/27/1997, 12:00:00
11	Irene	3/7/1996, 12:00:00

QUERY – 25

Modify the GPA of all students by giving 10% raise in their GPA .

```
ASSIGNMENT 1_2415000893.sql* ▾ ▷ ⏺ ⏹
82   SELECT sID, cName, decision from Apply;
83
84   SELECT sID from APPLY where cName = 'CORNEL' , 'M
85
86   DELETE from APPLY    where cName = 'Stanford';
87
88   DELETE from College where cName = 'Stanford';
89
90   UPDATE Student SET GPA = GPA * 1.10;
91
```

Query result **Script output** DBMS output Explain Plan



10 rows deleted.

Elapsed: 00:00:00.006

SQL> UPDATE Student SET GPA = GPA * 1.10

64 rows updated.

Elapsed: 00:00:00.004

SQL> UPDATE Student SET GPA = GPA * 1.10

64 rows updated.

Elapsed: 00:00:00.001

QUERY – 26

Increment GPA by 1.5 for students whose GPA < 3.5 and HSsize >1500

```
ASSIGNMENT 1_2415000893.sql* ▶ ⏪ ⏴ ⏵ ⏷
88 DELETE FROM College WHERE CNAME = 'Stanford';
89
90 UPDATE Student SET GPA = GPA * 1.10;
91
92 UPDATE student SET GPA = GPA+1.5 WHERE GPA<3.5 AND sizeHS >
93
```

Query result Script output DBMS output Explain Plan SQL his

Rows updated.

Elapsed: 00:00:00.003

SQL> UPDATE student SET GPA = GPA+1.5 WHERE GPA<3.5 AND sizeHS >150

0 rows updated.

Elapsed: 00:00:00.003

SQL> UPDATE student SET GPA = GPA+1.5 WHERE GPA<3.5 AND sizeHS >1500

0 rows updated.

Elapsed: 00:00:00.001

QUERY – 27

Delete students who have scored less than 3.2 GPA.

```
ASSIGNMENT 1_2415000893.sql* ▶

92 UPDATE student SET GPA = GPA+1.5 where
93
94 DELETE from Student where GPA < 3.2;

Query result Script output DBMS output
[Delete] [Save]
0 rows updated.

Elapsed: 00:00:00.001

SQL> DELETE from Student where GPA <3.2

0 rows deleted.

Elapsed: 00:00:00.003

SQL> DELETE from Student where GPA < 3.2

0 rows deleted.

Elapsed: 00:00:00.003
```