

CS3563 DBMS-II ASSIGNMENT 1

1)

```

Enter table name and value k
department 15

```

dept_name	building	budget
Civil Eng.	Chandler	255041.46
Biology	Candlestick	647610.55
History	Taylor	699140.86
Physics	Wrigley	942162.76
Marketing	Lambeau	210627.58
Pol. Sci.	Whitman	573745.09
English	Palmer	611042.66
Accounting	Saucon	441840.92
Comp. Sci.	Lamberton	106378.69
Languages	Linderman	601283.60
Finance	Candlestick	866831.75
Geology	Palmer	406557.93
Cybernetics	Mercer	794541.46
Astronomy	Taylor	617253.94
Athletics	Bronfman	734550.70

```

(base) pranav@XPS:~/Desktop/DBMS 2/Assign1/Assign1$

```

Fig1a. Department table

```

(base) pranav@XPS:~/Desktop/DBMS 2/Assign1/Assign1$ /usr/bin/env /usr/lib/jvm/jdk-19/bin/java @/tmp/cp_2g1w5st3skezzy/bwedojojdk.argsfile ex1
Enter table name and value k
section 15

```

course_id	sec_id	semester	year	building	room_number	time_slot_id
313	1	Fall	2010	Chandler	804	N
747	1	Spring	2004	Gates	314	K
443	1	Spring	2010	Whitman	434	O
893	1	Fall	2007	Fairchild	145	B
663	1	Spring	2005	Fairchild	145	D
457	1	Spring	2001	Saucon	844	D
445	1	Spring	2001	Alumni	547	J
559	1	Fall	2002	Lamberton	134	J
239	1	Fall	2006	Taylor	183	C
802	1	Spring	2003	Saucon	113	J
158	1	Fall	2008	Whitman	434	F
735	1	Spring	2003	Drown	757	D
237	1	Spring	2008	Power	717	D
338	1	Spring	2007	Fairchild	145	G
376	1	Fall	2006	Power	717	K

```

(base) pranav@XPS:~/Desktop/DBMS 2/Assign1/Assign1$

```

Fig1b. Section table.

```

(base) pranav@XPS:~/Desktop/DBMS 2/Assign1/Assign1$ /usr/bin/env /usr/lib/jvm/jdk-19/bin/java @/tmp/cp_2g1w5st3skezzy/bwedojojdk.argsfile ex1
Enter table name and value k
takes 15

```

id	course_id	sec_id	semester	year	grade
65901	401	1	Fall	2003	C-
24932	802	1	Spring	2003	B-
61332	200	1	Spring	2007	A-
73402	760	1	Spring	2004	A
65715	843	1	Fall	2010	B+
58300	408	1	Spring	2007	B-
760	571	1	Spring	2004	B
69730	313	1	Fall	2010	A
94836	626	1	Fall	2006	C+
49391	603	1	Fall	2003	B-
48850	457	1	Spring	2001	C-
68999	169	1	Spring	2007	A+
53469	169	1	Spring	2007	C
74016	747	1	Spring	2004	A-
49073	496	1	Fall	2001	C-

```

(base) pranav@XPS:~/Desktop/DBMS 2/Assign1/Assign1$

```

Fig1c. Takes table.

2)

```
(base) pranav@XPS:~/Desktop/DBMS_2/Assign1/A$  
Enter the course ID:  
276  
course_ID = 403, Title = Immunology  
  
course_ID = 345, Title = Race Car Driving  
  
course_ID = 352, Title = Compiler Design
```

Fig2a.Prerequisites for course_id = 276.

```
(base) pranav@XPS:~/Desktop/DBMS_2/Assign1/A$  
Enter the course ID:  
647  
course_ID = 792, Title = Image Processing  
  
course_ID = 814, Title = Compiler Design
```

Fig2a.Prerequisites for course_id = 647.

```
(base) pranav@XPS:~/Desktop/DBMS_2/Assign1/Assign1$ cd /home/pranav/desktop  
Enter the course ID:  
496  
course_ID = 489, Title = Journalism  
  
course_ID = 416, Title = Data Mining  
  
course_ID = 443, Title = Journalism  
  
course_ID = 852, Title = World History  
  
course_ID = 133, Title = Antidisestablishmentarianism in Modern America  
  
course_ID = 267, Title = Hydraulics
```

Fig2a.Prerequisites for course_id = 276.

3) Trigger on TEACHES TABLE:

INSERT:

- Incorrect Input:
insert into teaches(id,course_id,sec_id,semester,year)
values('99052','408','2','Spring',2003);
-> the course '408' with sec_id '2' in Spring semester 2003 belong to slot J, but the instructor '99052' already have a course '802' in year 2003 semester Spring.
Thus, Trigger do not allow user to add the above Query.
- Correct Input:
insert into teaches(id,course_id,sec_id,semester,year)
values('99052','137','1','Spring',2002);
-> instructor have no courses in year 2002 and Spring semester, Thus trigger allow user to add above Query

UPDATE:

- Incorrect Input:
update temp_teaches SET id = '6569', course_id = '349',
sec_id = '3',semester = 'Spring',year = '2008'
WHERE id = '6569'and course_id = '362'and
sec_id = '3' and semester = 'Spring';
-> instructor '6569' already teaches course '349' in same year and semester
thus when tried to replace course '362', trigger donot allow user to update.
- Correct Input:
update temp_teaches SET id = '77346', course_id = '443',
sec_id = '1',semester = 'Spring',year = '2010' WHERE
id = '77346'and course_id = '493'and sec_id = '1' and semester = 'Spring';
-> instructor '77346' teaches 2 courses in year 2010 and semester Spring,
but in slots H and D, the input course '443' is in slot O, as there's no clash,
thus trigger allows user to add above Query.

Trigger on SECTION TABLE:

UPDATE:

- Incorrect Input:
update section set time_slot_id ='M' where
course_id = '338' and sec_id = '1' and semester = 'Spring' and year = 2007 ;
-> in Spring and year 2007, course '400' also is in same slot 'M' taught by
instructor = '22591'. Thus trigger donot allow user to Query above input.
- Correct Input:
update section set time_slot_id ='J' where course_id = '338' and sec_id = '1'
and semester = 'Spring' and year = 2007 ;
-> there's no ther course in year 2007 and Spring semester with slot 'J' thus,
course '338' can be shifted from slot 'G' to 'J'.

4)

```
Enter Roll number to get CGPA :  
76672  
CGPA of 76672 = 5.081081  
(base) pranav@XPS:~/Desktop/DBMS_2/A  
Enter Roll number to get CGPA :  
90567  
CGPA of 90567 = 5.84127  
(base) pranav@XPS:~/Desktop/DBMS_2/A  
Enter Roll number to get CGPA :  
4582  
CGPA of 4582 = 5.361111  
(base) pranav@XPS:~/Desktop/DBMS_2/A  
Enter Roll number to get CGPA :  
81258  
CGPA of 81258 = 6.4761906
```

Fig4. CGPAs for Roll Num:

- i)7762.
- ii)90657.
- iii)4582.
- iv)81258.

5)

```
Enter value of k, to print Top K student ids with highest CGPA  
5  
  
ids-  
  
19362  
81207  
78637  
98619  
18286
```

Fig5a. Top 5 ids with Highest CGPA

```
Enter value of dept name and k respectively , to print Top K student ids with highest CGPA in given department
Psychology
5
ids-

17676
40116
32886
26494
51093
```

Fig5b_a. Top 5 ids with Highest CGPA
in 'Psychology' dept.

```
Enter value of dept name and k respectively , to print Top K student ids with highest CGPA in given department
Elec. Eng.
5
ids-

29140
64401
52057
59517
38712
```

Fig5b_b. Top 5 ids with Highest CGPA
in 'Elec. Eng. ' dept.

```
Enter value of dept name and k respectively , to print Top K student ids with highest CGPA in given department
Civil Eng.
5
ids-

98619
13753
40677
99660
32419
```

Fig5b_c. Top 5 ids with Highest CGPA
in 'Civil Eng.' dept.

```
(base) prandya@S: /Desktop/SDS_2/Assign1/Assign1$ ./ds1/bin/cmv ./ds1/cib/jvm/jdk-15/bin/java @/c
Enter value of k and dept name, to print Top K student ids with highest CGPA in given department
5 237
ids-

78637
66469
64196
77898
99289
```

Fig5c_a. Top 5 ids with Highest CGPA
with course_id '237'.

```
(base) pranshu@XPS: /Desktop/DBMS_2/Assign1/Assign1$  
Enter value of k and dept name, to print Top K student ids with highest CGPA in given department  
5 349  
ids-  
  
23506  
40116  
16311  
70021  
14094  
(base) pranshu@XPS: /Desktop/DBMS_2/Assign1/Assign1$
```

Fig5c_b. Top 5 ids with Highest CGPA
with course_id '349'.

```
(base) pranshu@XPS: /Desktop/DBMS_2/Assign1/Assign1$  
Enter value of k and dept name, to print Top K student ids with highest CGPA in given department  
5 735  
ids-  
  
18286  
77898  
52707  
11441  
39901  
(base) pranshu@XPS: /Desktop/DBMS_2/Assign1/Assign1$
```

Fig5c_a. Top 5 ids with Highest CGPA
with course_id '735'.