

(1)

Wasim-ur-Rahman

14524

BS (SE) 4<sup>th</sup> Sem  
Section 'A' -

DATA BASE FINAL PAPER



Question 1 :-

Sl. id	Student name	Student Address	Course ID	Course Name	Grade
01	Fawad	Karachi	SE-01	AI	A
02	walid	Lahore	SE-05	SQE	B
03	Saira	Peshawar	SE-02	DIP	C
04	Aiman	Karachi	SE-03	DB	A
05	Dangal	Lahore	SE-04	SRE	B
06	Emaan	Peshawar	SE-03	DB	C
			SE-01	AI	A
			SE-01	AI	B

• first the table should be changed into first Normal form because of the redundancy of data containing of

Std Id	Student name	Student Address	Course Id	Course Name	Grade
01	Fawad	Karachi	SE-01	AI	A
02	waleed	lahore	SE-02	SQE	B
03	Siara	Peshawar	SE-03	<del>DB</del> P	C
04	Aiman	Karachi	SE-04	DB	A
05	Danyal	lahore	SE-05	SRE	B
06	Emaan	Peshawar	SE-01	AI	C

Now as the upper table is in 2nd Normal Form we are moving on.

Course id	Course name
SE-01	AI
SE-02	SQE
SE-03	DIP
SE-04	DB
SE-05	SRE

\* Now in the upper course table "course Name" is only dependent on "Course Id".

normal Form, ~~which~~ <sup>here</sup> the upper normal table is in the 2nd normal form.

\* There should be no transitive dependancy for non prim attributes. So now first-

In the above table "student id" determines "student name" and "Course ID" determine "Course Name".

Therefore this implies that we have transitive dependency

we are dividing the table as show below-

STUDENT

student id	student name	student address	Grade
01	Fawad	Karachi	A
02	walkeed	lahore	B
03	Saira	peshawar	C
04	Aiman	Karachi	A
05	Danyal	lahore	B
06	Emaan	peshawar	C
			A
			B

\* Here in the upper table all columns are referring to one primary key "student id".

Question 2 :-

(5)

Answer :-

```
CREATE DATABASE Gallery;
```

```
CREATE TABLE Movies
```

```
( ID int primary key,
```

```
Movie-Name Varchar (255) NOT NULL,
```

```
Genre Varchar (255),
```

```
Year int CHECK (Year > 2020),
```

```
Rating int CHECK (Rating > 5),
```

```
);
```

### MOVIES

ID	Movie Name	Genre	Year	Rating
-	-	-	-	-
-	-	-	-	-
-	-	-	-	-

Question 3 :-

<del>student</del>	stnd Id	student name	Age	CGPA
-	-	-	-	-
-	-	-	-	-

(1) :- INSERTING DATA -  
CREATE TABLE ~~student~~ student

```
(
  student-id int primary key ,
  student-name varchar (255),
  Age int ,
  CGPA int,
);
```

```
INSERT INTO student ;
```

```
VALUES (14524 , 'wasim Rahman',
        21 , 2.8 );
```

```
INSERT INTO student ;
```

```
VALUES (14619, 'ABdul Aziz', 22, 3.1)
```

(7)

(8)

student id	Studentname	Age	CGPA
14524	waseem	21	2.8
14619	Abdul Aziz	22	3.1

(2)

Delete student record whose CGPA is greater 3 -

```
DELETE FROM student
WHERE CGPA > 3 ;
```

~ ~ ~ ~ ~

Question 4

part (1) :-

```
SELECT product-Name, product-id
FROM comteen-Table
WHERE Unit-Price < 50;
ORDER BY product-Name DESC,
Product-id ;
```

part 2 :-

product-List-Sorted Int;

(8)

```
SELECT * FROM canteen-table  
ORDER BY product Name;  
product - List - Sorted ;
```

```
SELECT * FROM canteentable  
ORDER BY product Name ;
```

part 3 :-

Category	Numbers
Junk	4
Not JUNK	2

part 4 :-

```
SELECT * FROM canteen-table  
INNER JOIN order-Details  
ON canteen-table . product-ID =  
order-Detail . product-ID ;
```

X ————— X