Work Integrated Learning Programmes

BITS Pilani
Pilani | Dubai | Goa | Hyderabad

Assignment

Subject Name	**Database Design and Applications**
Subject Code	**SSZG518**
Submitted By	
Gandhi Shail Nitinkumar	**2020MT13184**
Umesh Modak	**2020MT13092**

1. **About web based software which is useful to many people.**

   Due to the current situation, people avoid going out and do work from home, even online education is also done in college, so I thought I will make an online **MCQ based simple online exam application** which will be useful for students as well as professor also.

2. **Description of online MCQ based online exam application**

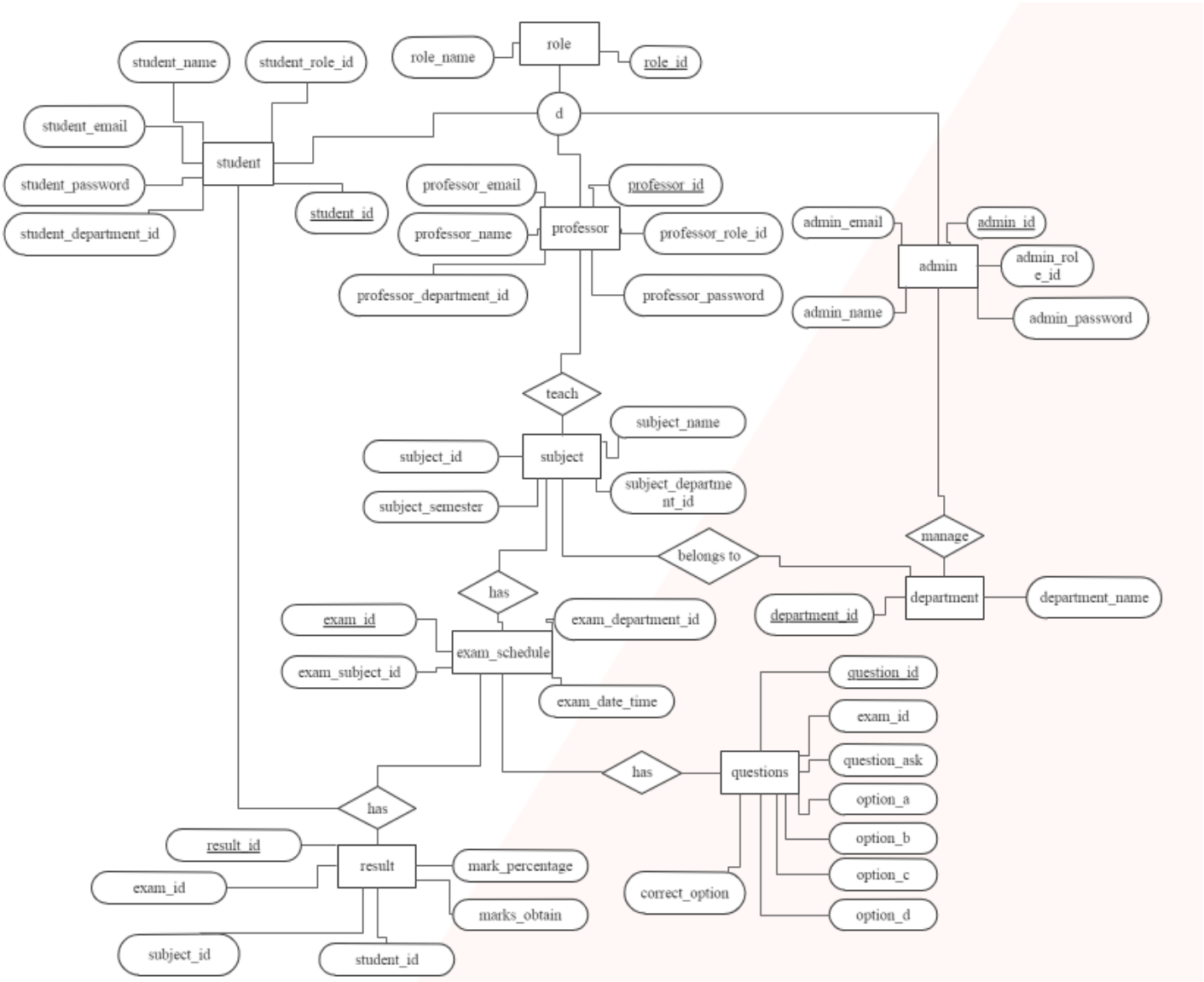
   - ***Users:*** Students, Professors and Admins are the users of this application.

   - ***Benefits:*** By using this application students give the exam across the globe and also professors can set the exam with less effort because all the evaluation will be done by the system so students will get immediate results after giving the examination. Also this system will reduce paper work also.

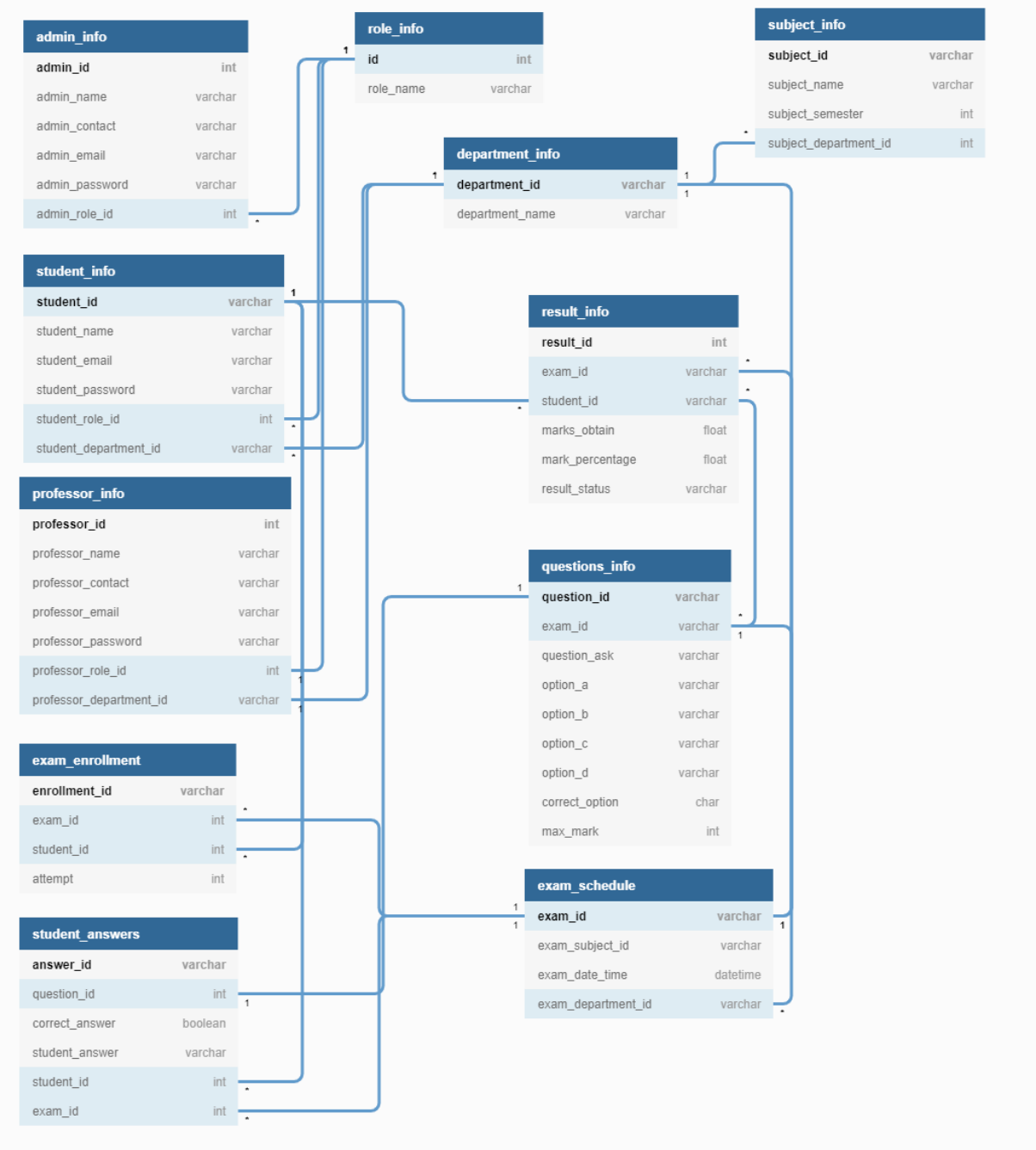
   - ***Functions and Features:***
     a. Quick evaluation of the exam.
     b. Saves time for the professor and students.
     c. Easy Schedule of exam.
     d. Store the examination archives.
     e. Students can verify the answers on their own.
     f. Grades are calculated automatically.

   - ***Simultaneously work:*** This is an online examination system and we use MySQL database in web applications so around 5000 students attempt the exam at a single time.

3. Entity Relation Diagram of MCQ based simple online exam application.



4. Entity Relation Diagram to Relational Database:



5. Normalize the Relational model

Sr. No	Relation	Candidate Key	Functional Dependency	1 NF	2 NF	3 NF	Reason for violation	BCNF	Reason for violation
1	role_Info	Id		✓	✓	✓			
2	admin_Info	admin_id		✓	✓	✓			
3	student_Info	student_id		✓	✓	✓			
4	professor_Info	professor_id		✓	✓	✓			
5	department_Info	department_id	department_id, department_name, subject_id subject_id subject_name	✓	✓	×	Transitive dependency exists as department_id --> subject_id and subject_id -->subject_name	×	As it is not in 3NF
6	exam_enrollment	enrollment_id		✓	✓	✓			
7	exam_schedule	exam_id		✓	✓	✓			
8	questions_Info	question_id		✓	✓	✓			
9	student_answers	answer_id		✓	✓	✓			
10	result_info	result_id		✓	✓	✓			

Normal Form	Conformance	Justification
1 NF	✓	All attributes hold atomic values
2 NF	✓	No Partial dependency exists
3 NF	✓	No Transitive dependency exists
BCNF	✓	Relation is in 3NF and in all FDs X-->Y, X is a super key

a. Normalizing the relation corresponding to violations

The FD subject_id-->subject_name in relation department_info is violating the 3 NF rule by having a transitive dependency. Hence following the normalize process breaking department_info into 2 relations as follows.

Relation	Attributes	Candidate Key	Functional Dependency
department_Info	department_id department_name	department_id	department_id --> department_name,college_name
subject_Info	subject_id, subject_name	subject_id	subject_id --> subject_name

b. Final set of relations post normalization

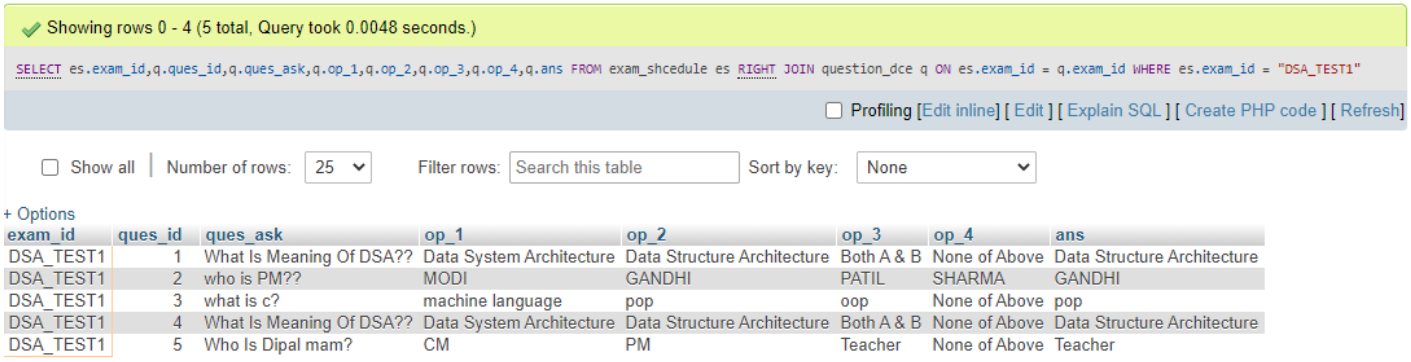
Sr.No	Relation	Candidate Key	Functional Dependency	1NF	2NF	3NF	BCNF
1	role_Info	Id		✓	✓	✓	✓
2	admin_Info	admin_id		✓	✓	✓	✓
3	student_Info	student_id		✓	✓	✓	✓
4	professor_Info	professor_id		✓	✓	✓	✓
5	department_Info	department_id	department_id --> department_name	✓	✓	✓	✓
6	subject_Info	department_id	subject_id-->subject_name	✓	✓	✓	✓
7	exam_enrollment	enrollment_id		✓	✓	✓	✓
8	exam_schedule	exam_id		✓	✓	✓	✓
9	questions_Info	question_id		✓	✓	✓	✓
10	student_answers	answer_id		✓	✓	✓	✓
11	result_info	result_id		✓	✓	✓	✓

6. SQL Queries

1. Get exam questions which is display during the examination

SELECT es.exam_id,q.ques_id,q.ques_ask,q.op_1,q.op_2,q.op_3,q.op_4,q.ans FROM exam_shcedule es RIGHT JOIN question_dce q ON es.exam_id = q.exam_id WHERE es.exam_id = "DSA_TEST1";

Result :



Showing rows 0 - 4 (5 total, Query took 0.0048 seconds.)

```
SELECT es.exam_id,q.ques_id,q.ques_ask,q.op_1,q.op_2,q.op_3,q.op_4,q.ans FROM exam_shcedule es RIGHT JOIN question_dce q ON es.exam_id = q.exam_id WHERE es.exam_id = "DSA_TEST1"
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

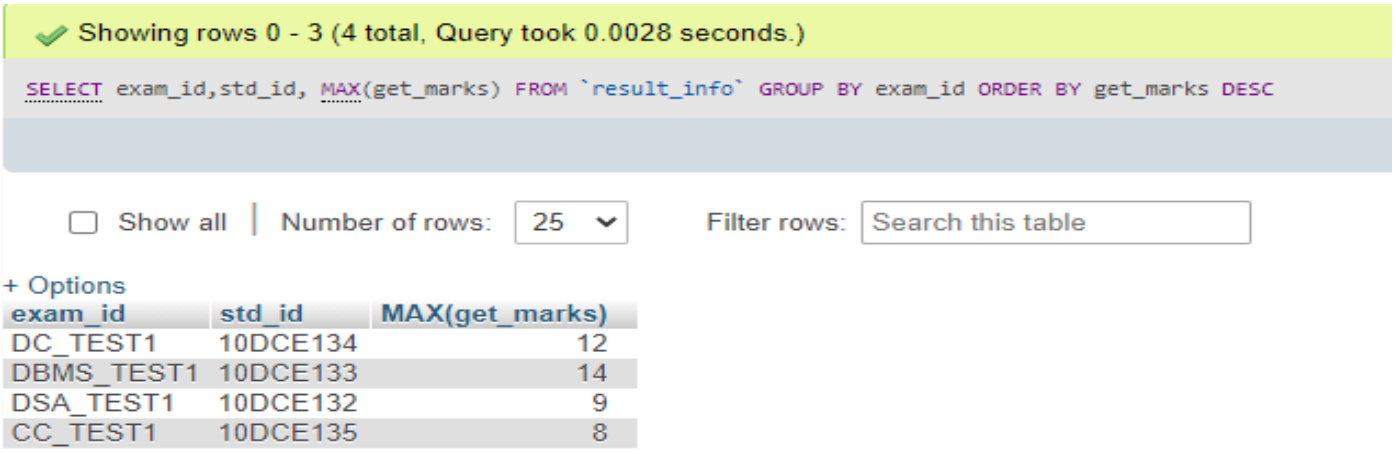
+ Options

exam_id	ques_id	ques_ask	op_1	op_2	op_3	op_4	ans
DSA_TEST1	1	What Is Meaning Of DSA??	Data System Architecture	Data Structure Architecture	Both A & B	None of Above	Data Structure Architecture
DSA_TEST1	2	who is PM??	MODI	GANDHI	PATIL	SHARMA	GANDHI
DSA_TEST1	3	what is c?	machine language	pop	oop	None of Above	pop
DSA_TEST1	4	What Is Meaning Of DSA??	Data System Architecture	Data Structure Architecture	Both A & B	None of Above	Data Structure Architecture
DSA_TEST1	5	Who Is Dipal mam?	CM	PM	Teacher	None of Above	Teacher

2. Exam wise highest marks with descending order

SELECT exam_id,std_id, MAX(get_marks) FROM `result_info` GROUP BY exam_id ORDER BY get_marks DESC;

Result :



Showing rows 0 - 3 (4 total, Query took 0.0028 seconds.)

```
SELECT exam_id,std_id, MAX(get_marks) FROM `result_info` GROUP BY exam_id ORDER BY get_marks DESC
```

Show all | Number of rows: 25 Filter rows: Search this table

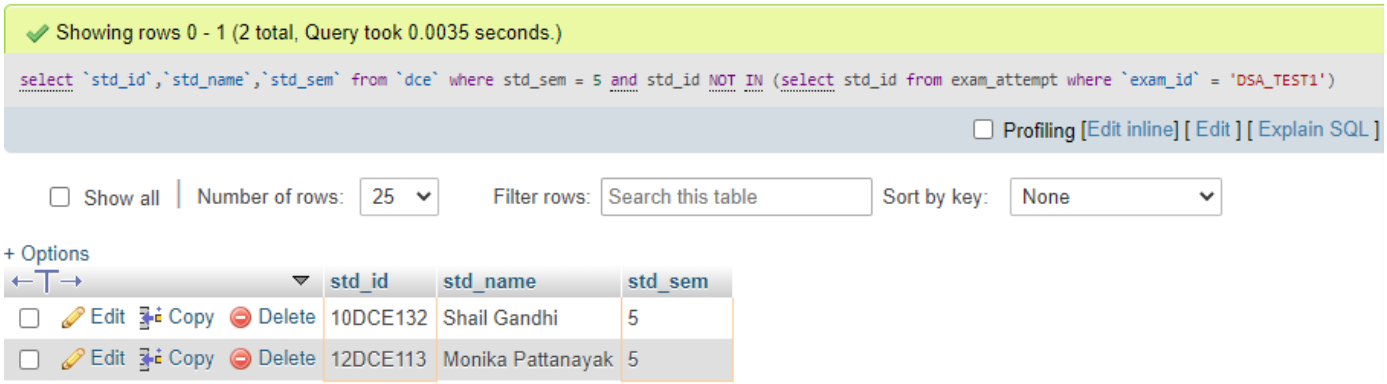
+ Options

exam_id	std_id	MAX(get_marks)
DC_TEST1	10DCE134	12
DBMS_TEST1	10DCE133	14
DSA_TEST1	10DCE132	9
CC_TEST1	10DCE135	8

3. List of students who is in 5th semester and they not attempt the DSA exam

SELECT`std_id`,`std_name`,`std_sem` from `dce` where std_sem = 5 and std_id NOT IN (select std_id from exam_attempt where `exam_id` = 'DSA_TEST1');

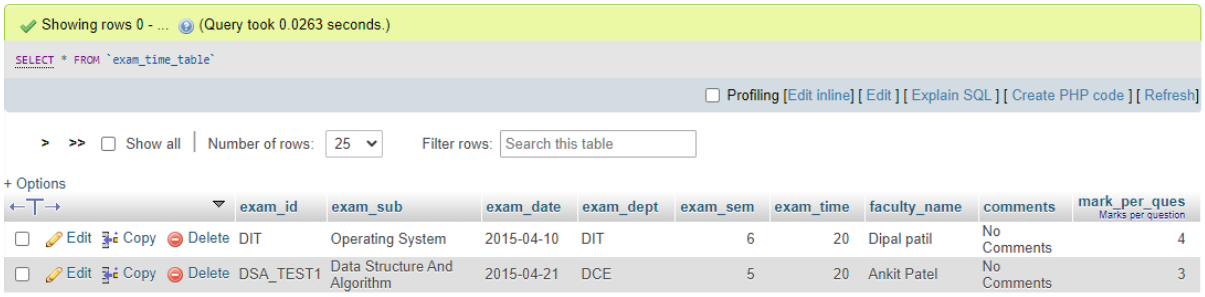
Result :

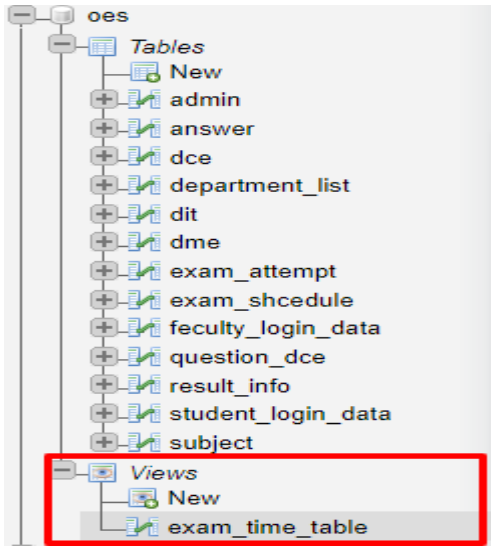


4. Create view for exam time table

CREATE VIEW ‘exam_time_table’ AS SELECT * FROM ‘exam_schedule’

Result:





5. Modify exam_schedule table and change data type and its size and make the comment column mandatory.

ALTER TABLE `exam_shcedule` CHANGE `comments` `comments` VARCHAR(255) NOT NULL;

Result :

MySQL returned an empty result set (i.e. zero rows). (Query took 0.0143 seconds.)

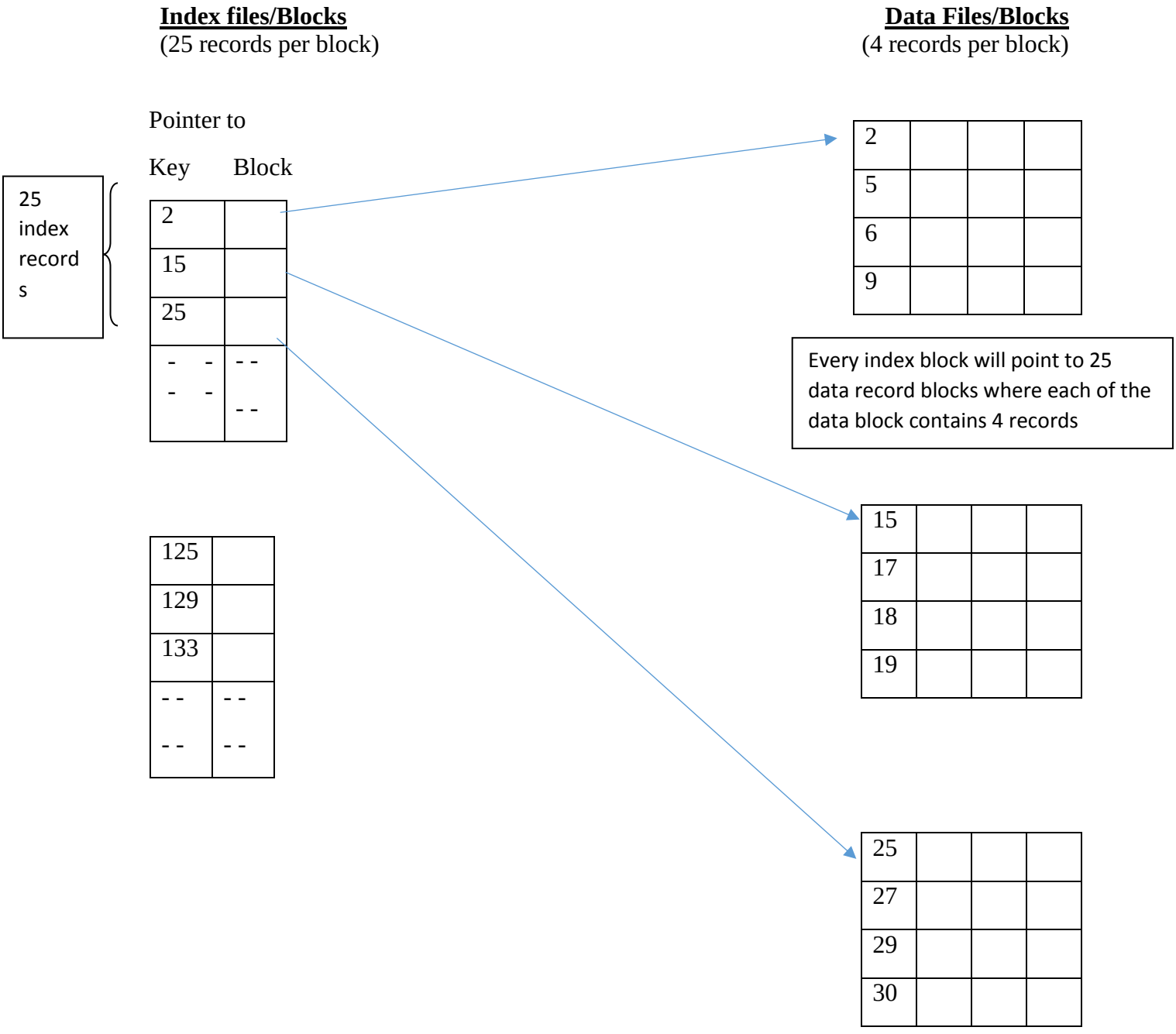
```
ALTER TABLE `exam_shcedule` CHANGE `comments` `comments` VARCHAR(255) NOT NULL
```

#	Name	Type	Collation	Attributes	Null	Default	Comments
1	**exam_id**	varchar(15)	latin1_swedish_ci		No	*None*	
2	**exam_sub**	varchar(200)	latin1_swedish_ci		No	*None*	
3	**exam_date**	date			No	*None*	
4	**exam_dept**	varchar(5)	latin1_swedish_ci		No	*None*	
5	**exam_sem**	int(3)			No	*None*	
6	**exam_time**	int(3)			Yes	*None*	
7	**faculty_name**	varchar(30)	latin1_swedish_ci		Yes	*None*	
8	**comments**	varchar(255)	latin1_swedish_ci		No	*None*	
9	**mark_per_ques**	int(4)			No	*None*	Marks per question

7. Design indices

In our Online Exam Application, one of the most frequently accessed table is ‘student_Info’ that is used to validate the login of all the students, who frequently access the Web App for all the purpose like exam enrolment, to appear in exam by recording answers, result inquiry etc. Therefore, an index needs to be created on the field ‘student_id’ to improve the query performance thereby improving the overall user experience of the application.

Since the field ‘student_id’ is the Primary key of the table “student_info” , the index created based on this field will be a Primary index.Since the data is already ordered on Primary key in the data file, the Primary key index is a sparse index as shown in the figure below:



b. **Argue on the space utilized and time saved on the retrieval. (You may take some random values to prove the same)**

Assume that we have an ordered file with 10000 records stored on disk.
Block size is 256 Bytes. Record length is fixed and it is 64 Bytes.
Key field (PK) length is 6 Bytes and block pointer is 4 Bytes.
Assume unspanned record organization.

Designing a Primary index on the field student_id.

No of Blocks needed for the 10000 Data Records

Size of disk block = 256 Bytes; record length = 64 Bytes

No. records per block (Bfr) = floor (256/64) = 4

No. of data blocks needed = ceil (10000/4) = 2500

No of Blocks needed for the 10000 Data Records

Block pointer = 4 Bytes. Key field = 6 bytes; total records = 80000

Index record length = key + pointer = 6+4 = 10 Bytes

Blocking factor for index (Bfr) = floor(256/10) = 25

No. of index blocks = Ceil (2500/25) = 100

Space requirement for index

Space requirement for index = No of Index Blocks*No of records per block * record length

Space requirement for index = 100*25 * 10

Space requirement for index = 25000 bytes = approx. 25KB

Time saved for retrieval

No. of block accesses = floor of (log2 100) + 1 = 6+1=7

This is significant time saving as just in 7 block access one can access the data block

c. **Take a small sample of at least 6 values on any attribute of the any table and construct the B+ tree.**

Let us have student_id key values to be arranged as B+ Tree for order 3 for the following values

8, 5, 1, 7, 3, 12, 9, 6

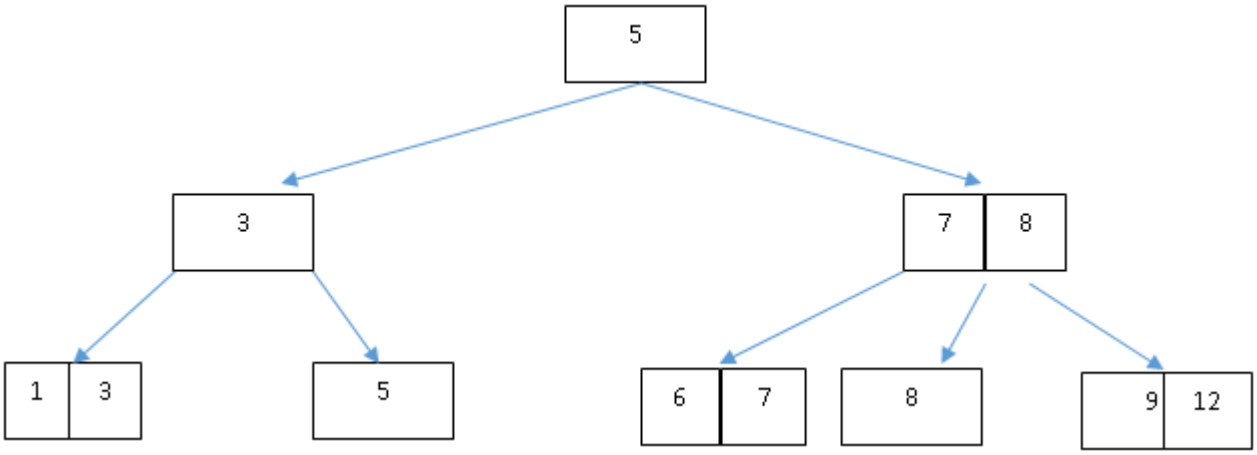
Assume number of pointers/children = p = 3

Max number of children = 3

Min number of children = Ceiling of (3/2) = 2

Max number of key fields = Max number of children – 1 = 3 – 1 = 2

Min number of key fields = Min number of children – 1 = 2 – 1 = 1



-------- END OF ASSIGNEMT --------