Roll Number:_____

Thapar Institute of Engineering and Technology, Patiala

Department of Computer Science and Engineering

END SEMESTER EXAMINATION

B. E. (Second Year): Semester-IV (2018/19) (COE)	Course Code: UCS310 Course Name: Database Management System
22 May, 2019	Wednesday, 9.00-12.00
Time: 3 Hours, M. Marks: 100	Name of Faculty: AV, GEK, SR, MAK, SUR,RR,HAR

Note: ***All questions are compulsory and attempt all parts of question at one place. Assume missing data, if any, suitably. Clearly specify your assumptions, if any.***

Q.1(a)	Consider the Schema of Books having the following attributes: *Book_id*, *Title*, *Publisher* and *Author(s)*. *Book_id* is the primary key, *Title* and *Publisher* are unique. *Author* field cannot be left empty. **i)** For the above schema, identify the category of SQL statements that can be mapped directly for describing the External and Logical Schema(s). ii) Support your answer with proper justification by writing the example(s) in SQL query, for the above schema, which helps to differentiate between the External and Logical Schemas.	(4)
Q1.(b)	Discuss the importance of Data Independence used in DBMS as opposed to the File System. Also explain different types of data independence.	(6)
Q.2(a)	Arnav has created a web application that allows the user to store and share files of approximate size between 1 MB - 10 MB. Now a user is trying to store a game Video files into the database using the following piece of SQL code: INSERT INTO GAME values (3, 'Terminator2', LOAD_FILE("C:\\Users\\Public\\Videos\\Sample Video\\test1.mpg")); INSERT INTO GAME values (4, 'Terminator3', LOAD_FILE("C:\\Users\\Public\\Videos\\Sample Video\\test2.mpg")); where the Game is schema having attributes(Serial Number, Game_Name, File_Path); i) Discuss the advantages and disadvantages between File Management System and DBMS approaches for the above mentioned scenario with respect to memory utilization, security, economically and processing time. **ii)** Based upon the above differentiation, which approach you would like to suggest to Arnav? Justify your answer.	(10)
Q.2(b)	Write the *SQL* code for creating a sequence which generates numbers in following order: 42, 44, 46, 48, 50, 40, 42, 44, 46, 48, 50 and so on…	(5)

P.T.O

Q.3(a)	Consider the following ***Employee*** relation having attributes(*empid, ename, dob, salary, pan_card*) A Bank employee has to access one million records for search. Frequent search query operations are either based on *empid* or ***pan_card***. Now, the Bank Employee wants to increase the system efficiency for fast data retrieval. As a database programmer, what change(s) would you like to implement on the above existing Schema, for faster data retrieval? Justify your answer.	(5)
Q.3(b)	Discuss any two differences between the following terms with appropriate examples: **i)** Trivial and Non Trivial Dependency. **ii)** Left Outer Join and Right Outer Join. **iii)** Correlated Query and Sub-query.	(10)
Q.4(a)	An agency called *InstantCover* supplies part-time/temporary staff to hotels throughout Scotland. The relation *R* along with data shown below lists the time spent by agency staff working at two hotels. The National Insurance Number (NIN) is unique for employee.	(10)

NIN	ContractNo	Hoursper week	Ename	Hotel_ No	HotelLocation
13567WD	C1024	16	John S	H25	Edinburg
34111XA	C1024	24	Diane H	H25	Edinburg
12670YD	C1025	28	Sarah W	H4	Glasgow
13567WD	C1025	16	John S	H4	Glasgow

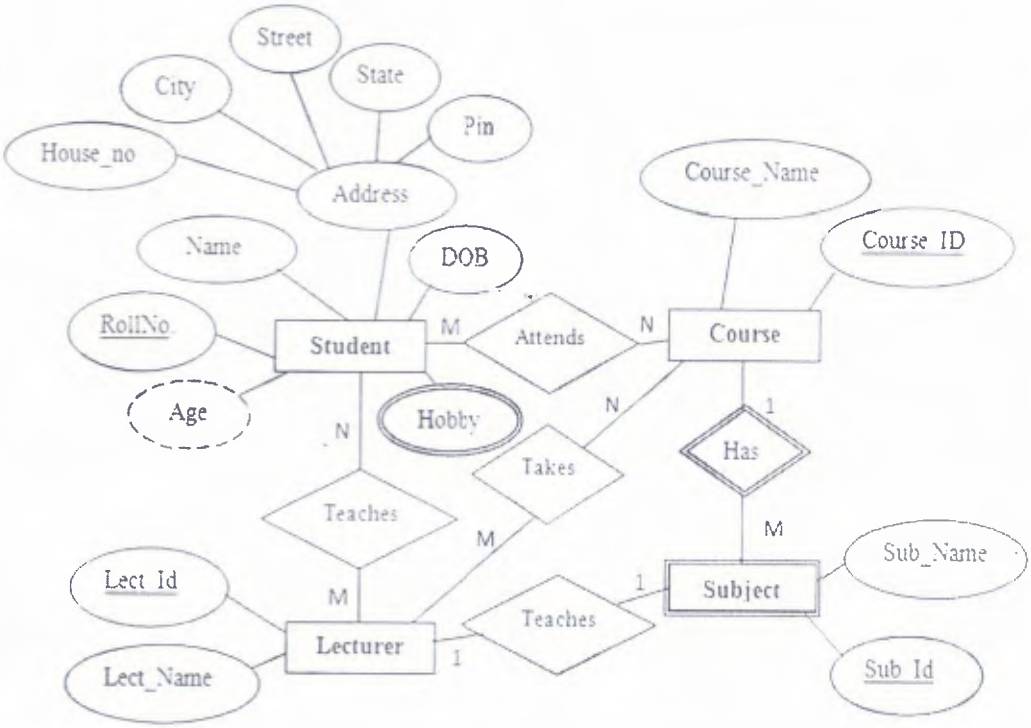
i) Find out all functional dependencies (*based only upon the existing data given above in relation R*). Justify your answer.

ii) Normalize this relation to 3NF, by applying each normal form one by one. Indicate all the intermediate steps.

Q.4(b)	Find the minimal cover of the below given set of functional dependencies, *FD*, of a relation *R* (*A, B, C, D, E, F*) and then determine the candidate keys using the closure property. $FD=\{A\rightarrow BC, CD\rightarrow E, E\rightarrow C, D\rightarrow AEF, ABF\rightarrow BD, DF\rightarrow BC\}$	(5)
Q.5(a)	Consider following relations : **i)** **Client_master**(*client_no, name, address, city, pincode, state, bal_due*); **ii)** **Sales_order**(*order_no, order_date, client_no* foreign key references *client_no* of the *Client_master* table); **iii)** **Product_master**(*product_no, description, quantity_on_hand, selling_price, cost_price*); **iv)** **Sales_order_details**(*order_no* (primary key/foreign key references *order_no* of the *sales_order* table), *product_no* (primary key/ foreign key references *product_no* of the *Product_master* table), quantity_*ordered, quantity_disp, product rate*) ; **Note:** Underlined attributes represent the primary key.	(10)

P.T.O

	Write SQL queries for the following: **i)** Find the names of all clients having '*a*' as the second letter in their name. **ii)** Display the *order number* and *day* on which client placed their order. **iii)** List the *names, city* and *state* of clients who are not in the state of '*Maharashtra*'. **iv)** Print the description and total quantity sold for each product. **v)** Retrieve the *product no* and total quantity ordered for products '*P0001*' and '*P0002*' from sales_order_details.	
Q.5(b)	Consider the following relations: **Faculty**(*Faculty_id, F_name, Salary, Dept_id*) **Dept**(*Dep_No, Dname*) **Project**(*Proj_no, Pname, Dp_no*) **Write relational algebra queries for the following:** **i)** Find the *name* and *salary* of all the faculties. **ii)** Obtain the Faculty *names* of all the faculties having *salary* between *10000* and *20000.* **iii)** Find the details of Faculties working on the "*Website project*". **iv)** Find the *name, Faculty id* and *salary* of all faculties working for "*Computer science*"" department. **v)** Find the *project number*, controlling *department name* and *faculties* working in that project. **Note:** Underlined attributes represent the primary key.	(5)
Q.6(a)	**Design** and draw an ER diagram that captures all the following requirements about *Soccer team*. Use only basic ER model here; that is *entities, relationships* and *attributes.* Assume we have the following application that models soccer teams, the games they play, and the players in each team. In the design, we want to capture the following: • We have a set of teams, each team has an ID (unique identifier), name, main stadium, and to which city this team belongs. • Each team has many players, and each player belongs to one team. Each player has a number (unique identifier), name, DoB, start year, and shirt number that he uses. • Teams play matches, in each match there is a host team and a guest team. The match takes place in the stadium of the host team. • For each match, we need to keep track of the following: o The date on which the game is played o The final result of the match o The players participated in the match. For each player, how many goals he scored, whether or not he took yellow card, and whether or not he took red card. o During the match, one player may substitute another player. We want to capture this substitution and the time at which it took place. Each match has exactly three referees. For each referee, we have an ID (unique	(10) P.T.O.

	identifier), name, DoB, years of experience. One referee is the main referee and the other two are assistant referees. Make sure cardinalities and primary keys are clearly stated.	
Q.6(b)	Convert the following ER Diagram to database tables. Explicitly mention primary and foreign keys for each of the table. 	(5)
Q7	Consider the below relation named ***EMP*** and answer the following PL/SQL block questions (Question 7 (a), (b), (c)). ***EMP***(e_id, e_name, sal, dept_id, hire_date)	
(a)	Write a stored ***function*** TAX that accepts salary as input parameter and returns the tax information's of an employee from ***EMP*** relation. **Salary** — **Tax** Less Than or equal to 1000 — No Tax Greater Than 1000 but <=2000 — 10% of salary above 1000 Greater Than 2000 — 20% of salary above 2000	(5)
(b)	Write a PL/SQL program to create a ***trigger*** that ensures no changes can be made to ***EMP*** relation before 6AM and after 10 PM in a day.	(5)
(c)	Create a PL/SQL block to declare a cursor **EMP_CUR** to select the employee name, salary and hire date from the ***EMP*** relation. Process each row from the cursor, if the *salary* is greater than *15,000* and the *hire date* is greater than *01-FEB-2009*. Display the employee name, salary and hire date in the following format : *Mr Beans earns 75000 and joined the organization on 22 oct, 2010.* *Mr Varun earns 55000 and joined the organization on 13 sept, 2013.*	(5)