1. Write a SQL statement to create a simple table countries including columns country\_id,country\_name and region\_id.

+--------------+---------------+------+-----+---------+-------+

| Field | Type | Null | Key | Default | Extra |

+--------------+---------------+------+-----+---------+-------+

| COUNTRY\_ID | varchar(2) | YES | | NULL | |

| COUNTRY\_NAME | varchar(40) | YES | | NULL | |

| REGION\_ID | decimal(10,0) | YES | | NULL | |

+--------------+---------------+------+-----+---------+-------+

1. Write a SQL statement to create a table named jobs including columns job\_id, job\_title, min\_salary, max\_salary and check whether the max\_salary amount exceeding the upper limit 25000.
2. Write a SQL statement to create a table named countries including columns country\_id, country\_name and region\_id and make sure that no countries except Italy, India and China will be entered in the table.
3. Write a SQL statement to create a table named countries including columns country\_id,country\_name and region\_id and make sure that no duplicate data against column country\_id will be allowed at the time of insertion.
4. Write a SQL statement to create a table named jobs including columns job\_id, job\_title, min\_salary and max\_salary, and make sure that, the default value for job\_title is blank and min\_salary is 8000 and max\_salary is NULL will be entered automatically at the time of insertion if no value assigned for the specified columns.
5. Write a SQL statement to create a table countries including columns country\_id, country\_name and region\_id and make sure that the column country\_id will be unique and store an auto incremented value.
6. Write a SQL statement to create a table countries including columns country\_id, country\_name and region\_id and make sure that the combination of columns country\_id and region\_id will be unique.
7. Write a SQL statement to create a table job\_history including columns employee\_id, start\_date, end\_date, job\_id and department\_id and make sure that, the employee\_id column does not contain any duplicate value at the time of insertion and the foreign key column job\_id contain only those values which are exists in the jobs table.

Here is the structure of the table jobs;

+------------+--------------+------+-----+---------+-------+

| Field | Type | Null | Key | Default | Extra |

+------------+--------------+------+-----+---------+-------+

| JOB\_ID | varchar(10) | NO | PRI | | |

| JOB\_TITLE | varchar(35) | NO | | NULL | |

| MIN\_SALARY | decimal(6,0) | YES | | NULL | |

| MAX\_SALARY | decimal(6,0) | YES | | NULL | |

+------------+--------------+------+-----+---------+-------+

1. Write a SQL statement to create a table employees including columns employee\_id, first\_name, last\_name, job\_id, salary and make sure that, the employee\_id column does not contain any duplicate value at the time of insertion, and the foreign key column job\_id, referenced by the column job\_id of jobs table, can contain only those values which are exists in the jobs table. The InnoDB Engine have been used to create the tables. The specialty of the statement is that, The ON DELETE SET NULL action will set the foreign key column values in the child table(employees) to NULL when the record in the parent table(jobs) is deleted, with a condition that the foreign key column in the child table must accept NULL values and the ON UPDATE SET NULL action resets the values in the rows in the child table(employees) to NULL values when the rows in the parent table(jobs) are updated.  
   Assume that the structure of two table jobs and InnoDB Engine have been used to create the table jobs.

CREATE TABLE IF NOT EXISTS jobs (

JOB\_ID integer NOT NULL UNIQUE PRIMARY KEY,

JOB\_TITLE varchar(35) NOT NULL DEFAULT ' ',

MIN\_SALARY decimal(6,0) DEFAULT 8000,

MAX\_SALARY decimal(6,0) DEFAULT NULL

)ENGINE=InnoDB;

+------------+--------------+------+-----+---------+-------+

| Field | Type | Null | Key | Default | Extra |

+------------+--------------+------+-----+---------+-------+

| JOB\_ID | int(11) | NO | PRI | NULL | |

| JOB\_TITLE | varchar(35) | NO | | | |

| MIN\_SALARY | decimal(6,0) | YES | | 8000 | |

| MAX\_SALARY | decimal(6,0) | YES | | NULL | |

+------------+--------------+------+-----+---------+-------+

1. Write a SQL statement to create a table employees including columns employee\_id, first\_name, last\_name, job\_id, salary and make sure that, the employee\_id column does not contain any duplicate value at the time of insertion, and the foreign key column job\_id, referenced by the column job\_id of jobs table, can contain only those values which are exists in the jobs table. The InnoDB Engine have been used to create the tables. The specialty of the statement is that, The ON DELETE NO ACTION and the ON UPDATE NO ACTION actions will reject the deletion and any updates.

Assume that the structure of two table jobs and InnoDB Engine have been used to create the table jobs.

CREATE TABLE IF NOT EXISTS jobs (

JOB\_ID integer NOT NULL UNIQUE PRIMARY KEY,

JOB\_TITLE varchar(35) NOT NULL DEFAULT ' ',

MIN\_SALARY decimal(6,0) DEFAULT 8000,

MAX\_SALARY decimal(6,0) DEFAULT NULL

)ENGINE=InnoDB;

+------------+--------------+------+-----+---------+-------+

| Field | Type | Null | Key | Default | Extra |

+------------+--------------+------+-----+---------+-------+

| JOB\_ID | int(11) | NO | PRI | NULL | |

| JOB\_TITLE | varchar(35) | NO | | | |

| MIN\_SALARY | decimal(6,0) | YES | | 8000 | |

| MAX\_SALARY | decimal(6,0) | YES | | NULL | |

+------------+--------------+------+-----+---------+-------+