**CREATE DATABASE exercises;**

**USE exercises;**

**-----------------------------------------------------------**

**--**

**-- Table structure for table `countries`**

**--**

**CREATE TABLE IF NOT EXISTS `countries` (**

**`COUNTRY\_ID` varchar(2) NOT NULL,**

**`COUNTRY\_NAME` varchar(40) DEFAULT NULL,**

**`REGION\_ID` decimal(10,0) DEFAULT NULL,**

**PRIMARY KEY (`COUNTRY\_ID`),**

**KEY `COUNTR\_REG\_FK` (`REGION\_ID`)**

**) ENGINE=MyISAM DEFAULT CHARSET=latin1;**

**--**

**-- Dumping data for table `countries`**

**--**

**INSERT INTO `countries` (`COUNTRY\_ID`, `COUNTRY\_NAME`, `REGION\_ID`) VALUES**

**('AR', 'Argentina', '2'),**

**('AU', 'Australia', '3'),**

**('BE', 'Belgium', '1'),**

**('BR', 'Brazil', '2'),**

**('CA', 'Canada', '2'),**

**('CH', 'Switzerland', '1'),**

**('CN', 'China', '3'),**

**('DE', 'Germany', '1'),**

**('DK', 'Denmark', '1'),**

**('EG', 'Egypt', '4'),**

**('FR', 'France', '1'),**

**('HK', 'HongKong', '3'),**

**('IL', 'Israel', '4'),**

**('IN', 'India', '3'),**

**('IT', 'Italy', '1'),**

**('JP', 'Japan', '3'),**

**('KW', 'Kuwait', '4'),**

**('MX', 'Mexico', '2'),**

**('NG', 'Nigeria', '4'),**

**('NL', 'Netherlands', '1'),**

**('SG', 'Singapore', '3'),**

**('UK', 'United Kingdom', '1'),**

**('US', 'United States of America', '2'),**

**('ZM', 'Zambia', '4'),**

**('ZW', 'Zimbabwe', '4');**

**-- --------------------------------------------------------**

**--**

**-- Table structure for table `departments`**

**--**

**CREATE TABLE IF NOT EXISTS `departments` (**

**`DEPARTMENT\_ID` decimal(4,0) NOT NULL DEFAULT '0',**

**`DEPARTMENT\_NAME` varchar(30) NOT NULL,**

**`MANAGER\_ID` decimal(6,0) DEFAULT NULL,**

**`LOCATION\_ID` decimal(4,0) DEFAULT NULL,**

**PRIMARY KEY (`DEPARTMENT\_ID`),**

**KEY `DEPT\_MGR\_FK` (`MANAGER\_ID`),**

**KEY `DEPT\_LOCATION\_IX` (`LOCATION\_ID`)**

**) ENGINE=MyISAM DEFAULT CHARSET=latin1;**

**--**

**-- Dumping data for table `departments`**

**--**

**INSERT INTO `departments` (`DEPARTMENT\_ID`, `DEPARTMENT\_NAME`, `MANAGER\_ID`, `LOCATION\_ID`) VALUES**

**('10', 'Administration', '200', '1700'),**

**('20', 'Marketing', '201', '1800'),**

**('30', 'Purchasing', '114', '1700'),**

**('40', 'Human Resources', '203', '2400'),**

**('50', 'Shipping', '121', '1500'),**

**('60', 'IT', '103', '1400'),**

**('70', 'Public Relations', '204', '2700'),**

**('80', 'Sales', '145', '2500'),**

**('90', 'Executive', '100', '1700'),**

**('100', 'Finance', '108', '1700'),**

**('110', 'Accounting', '205', '1700'),**

**('120', 'Treasury', '0', '1700'),**

**('130', 'Corporate Tax', '0', '1700'),**

**('140', 'Control And Credit', '0', '1700'),**

**('150', 'Shareholder Services', '0', '1700'),**

**('160', 'Benefits', '0', '1700'),**

**('170', 'Manufacturing', '0', '1700'),**

**('180', 'Construction', '0', '1700'),**

**('190', 'Contracting', '0', '1700'),**

**('200', 'Operations', '0', '1700'),**

**('210', 'IT Support', '0', '1700'),**

**('220', 'NOC', '0', '1700'),**

**('230', 'IT Helpdesk', '0', '1700'),**

**('240', 'Government Sales', '0', '1700'),**

**('250', 'Retail Sales', '0', '1700'),**

**('260', 'Recruiting', '0', '1700'),**

**('270', 'Payroll', '0', '1700');**

**-- --------------------------------------------------------**

**--**

**-- Table structure for table `employees`**

**--**

**CREATE TABLE IF NOT EXISTS `employees` (**

**`EMPLOYEE\_ID` decimal(6,0) NOT NULL DEFAULT '0',**

**`FIRST\_NAME` varchar(20) DEFAULT NULL,**

**`LAST\_NAME` varchar(25) NOT NULL,**

**`EMAIL` varchar(25) NOT NULL,**

**`PHONE\_NUMBER` varchar(20) DEFAULT NULL,**

**`HIRE\_DATE` date NOT NULL,**

**`JOB\_ID` varchar(10) NOT NULL,**

**`SALARY` decimal(8,2) DEFAULT NULL,**

**`COMMISSION\_PCT` decimal(2,2) DEFAULT NULL,**

**`MANAGER\_ID` decimal(6,0) DEFAULT NULL,**

**`DEPARTMENT\_ID` decimal(4,0) DEFAULT NULL,**

**PRIMARY KEY (`EMPLOYEE\_ID`),**

**UNIQUE KEY `EMP\_EMAIL\_UK` (`EMAIL`),**

**KEY `EMP\_DEPARTMENT\_IX` (`DEPARTMENT\_ID`),**

**KEY `EMP\_JOB\_IX` (`JOB\_ID`),**

**KEY `EMP\_MANAGER\_IX` (`MANAGER\_ID`),**

**KEY `EMP\_NAME\_IX` (`LAST\_NAME`,`FIRST\_NAME`)**

**) ENGINE=MyISAM DEFAULT CHARSET=latin1;**

**--**

**-- Dumping data for table `employees`**

**--**

**INSERT INTO `employees` (`EMPLOYEE\_ID`, `FIRST\_NAME`, `LAST\_NAME`, `EMAIL`, `PHONE\_NUMBER`, `HIRE\_DATE`, `JOB\_ID`, `SALARY`, `COMMISSION\_PCT`, `MANAGER\_ID`, `DEPARTMENT\_ID`) VALUES**

**('100', 'Steven', 'King', 'SKING', '515.123.4567', '1987-06-17', 'AD\_PRES', '24000.00', '0.00', '0', '90'),**

**('101', 'Neena', 'Kochhar', 'NKOCHHAR', '515.123.4568', '1987-06-18', 'AD\_VP', '17000.00', '0.00', '100', '90'),**

**('102', 'Lex', 'De Haan', 'LDEHAAN', '515.123.4569', '1987-06-19', 'AD\_VP', '17000.00', '0.00', '100', '90'),**

**('103', 'Alexander', 'Hunold', 'AHUNOLD', '590.423.4567', '1987-06-20', 'IT\_PROG', '9000.00', '0.00', '102', '60'),**

**('104', 'Bruce', 'Ernst', 'BERNST', '590.423.4568', '1987-06-21', 'IT\_PROG', '6000.00', '0.00', '103', '60'),**

**('105', 'David', 'Austin', 'DAUSTIN', '590.423.4569', '1987-06-22', 'IT\_PROG', '4800.00', '0.00', '103', '60'),**

**('106', 'Valli', 'Pataballa', 'VPATABAL', '590.423.4560', '1987-06-23', 'IT\_PROG', '4800.00', '0.00', '103', '60'),**

**('107', 'Diana', 'Lorentz', 'DLORENTZ', '590.423.5567', '1987-06-24', 'IT\_PROG', '4200.00', '0.00', '103', '60'),**

**('108', 'Nancy', 'Greenberg', 'NGREENBE', '515.124.4569', '1987-06-25', 'FI\_MGR', '12000.00', '0.00', '101', '100'),**

**('109', 'Daniel', 'Faviet', 'DFAVIET', '515.124.4169', '1987-06-26', 'FI\_ACCOUNT', '9000.00', '0.00', '108', '100'),**

**('110', 'John', 'Chen', 'JCHEN', '515.124.4269', '1987-06-27', 'FI\_ACCOUNT', '8200.00', '0.00', '108', '100'),**

**('111', 'Ismael', 'Sciarra', 'ISCIARRA', '515.124.4369', '1987-06-28', 'FI\_ACCOUNT', '7700.00', '0.00', '108', '100'),**

**('112', 'Jose Manuel', 'Urman', 'JMURMAN', '515.124.4469', '1987-06-29', 'FI\_ACCOUNT', '7800.00', '0.00', '108', '100'),**

**('113', 'Luis', 'Popp', 'LPOPP', '515.124.4567', '1987-06-30', 'FI\_ACCOUNT', '6900.00', '0.00', '108', '100'),**

**('114', 'Den', 'Raphaely', 'DRAPHEAL', '515.127.4561', '1987-07-01', 'PU\_MAN', '11000.00', '0.00', '100', '30'),**

**('115', 'Alexander', 'Khoo', 'AKHOO', '515.127.4562', '1987-07-02', 'PU\_CLERK', '3100.00', '0.00', '114', '30'),**

**('116', 'Shelli', 'Baida', 'SBAIDA', '515.127.4563', '1987-07-03', 'PU\_CLERK', '2900.00', '0.00', '114', '30'),**

**('117', 'Sigal', 'Tobias', 'STOBIAS', '515.127.4564', '1987-07-04', 'PU\_CLERK', '2800.00', '0.00', '114', '30'),**

**('118', 'Guy', 'Himuro', 'GHIMURO', '515.127.4565', '1987-07-05', 'PU\_CLERK', '2600.00', '0.00', '114', '30'),**

**('119', 'Karen', 'Colmenares', 'KCOLMENA', '515.127.4566', '1987-07-06', 'PU\_CLERK', '2500.00', '0.00', '114', '30'),**

**('120', 'Matthew', 'Weiss', 'MWEISS', '650.123.1234', '1987-07-07', 'ST\_MAN', '8000.00', '0.00', '100', '50'),**

**('121', 'Adam', 'Fripp', 'AFRIPP', '650.123.2234', '1987-07-08', 'ST\_MAN', '8200.00', '0.00', '100', '50'),**

**('122', 'Payam', 'Kaufling', 'PKAUFLIN', '650.123.3234', '1987-07-09', 'ST\_MAN', '7900.00', '0.00', '100', '50'),**

**('123', 'Shanta', 'Vollman', 'SVOLLMAN', '650.123.4234', '1987-07-10', 'ST\_MAN', '6500.00', '0.00', '100', '50'),**

**('124', 'Kevin', 'Mourgos', 'KMOURGOS', '650.123.5234', '1987-07-11', 'ST\_MAN', '5800.00', '0.00', '100', '50'),**

**('125', 'Julia', 'Nayer', 'JNAYER', '650.124.1214', '1987-07-12', 'ST\_CLERK', '3200.00', '0.00', '120', '50'),**

**('126', 'Irene', 'Mikkilineni', 'IMIKKILI', '650.124.1224', '1987-07-13', 'ST\_CLERK', '2700.00', '0.00', '120', '50'),**

**('127', 'James', 'Landry', 'JLANDRY', '650.124.1334', '1987-07-14', 'ST\_CLERK', '2400.00', '0.00', '120', '50'),**

**('128', 'Steven', 'Markle', 'SMARKLE', '650.124.1434', '1987-07-15', 'ST\_CLERK', '2200.00', '0.00', '120', '50'),**

**('129', 'Laura', 'Bissot', 'LBISSOT', '650.124.5234', '1987-07-16', 'ST\_CLERK', '3300.00', '0.00', '121', '50'),**

**('130', 'Mozhe', 'Atkinson', 'MATKINSO', '650.124.6234', '1987-07-17', 'ST\_CLERK', '2800.00', '0.00', '121', '50'),**

**('131', 'James', 'Marlow', 'JAMRLOW', '650.124.7234', '1987-07-18', 'ST\_CLERK', '2500.00', '0.00', '121', '50'),**

**('132', 'TJ', 'Olson', 'TJOLSON', '650.124.8234', '1987-07-19', 'ST\_CLERK', '2100.00', '0.00', '121', '50'),**

**('133', 'Jason', 'Mallin', 'JMALLIN', '650.127.1934', '1987-07-20', 'ST\_CLERK', '3300.00', '0.00', '122', '50'),**

**('134', 'Michael', 'Rogers', 'MROGERS', '650.127.1834', '1987-07-21', 'ST\_CLERK', '2900.00', '0.00', '122', '50'),**

**('135', 'Ki', 'Gee', 'KGEE', '650.127.1734', '1987-07-22', 'ST\_CLERK', '2400.00', '0.00', '122', '50'),**

**('136', 'Hazel', 'Philtanker', 'HPHILTAN', '650.127.1634', '1987-07-23', 'ST\_CLERK', '2200.00', '0.00', '122', '50'),**

**('137', 'Renske', 'Ladwig', 'RLADWIG', '650.121.1234', '1987-07-24', 'ST\_CLERK', '3600.00', '0.00', '123', '50'),**

**('138', 'Stephen', 'Stiles', 'SSTILES', '650.121.2034', '1987-07-25', 'ST\_CLERK', '3200.00', '0.00', '123', '50'),**

**('139', 'John', 'Seo', 'JSEO', '650.121.2019', '1987-07-26', 'ST\_CLERK', '2700.00', '0.00', '123', '50'),**

**('140', 'Joshua', 'Patel', 'JPATEL', '650.121.1834', '1987-07-27', 'ST\_CLERK', '2500.00', '0.00', '123', '50'),**

**('141', 'Trenna', 'Rajs', 'TRAJS', '650.121.8009', '1987-07-28', 'ST\_CLERK', '3500.00', '0.00', '124', '50'),**

**('142', 'Curtis', 'Davies', 'CDAVIES', '650.121.2994', '1987-07-29', 'ST\_CLERK', '3100.00', '0.00', '124', '50'),**

**('143', 'Randall', 'Matos', 'RMATOS', '650.121.2874', '1987-07-30', 'ST\_CLERK', '2600.00', '0.00', '124', '50'),**

**('144', 'Peter', 'Vargas', 'PVARGAS', '650.121.2004', '1987-07-31', 'ST\_CLERK', '2500.00', '0.00', '124', '50'),**

**('145', 'John', 'Russell', 'JRUSSEL', '011.44.1344.429268', '1987-08-01', 'SA\_MAN', '14000.00', '0.40', '100', '80'),**

**('146', 'Karen', 'Partners', 'KPARTNER', '011.44.1344.467268', '1987-08-02', 'SA\_MAN', '13500.00', '0.30', '100', '80'),**

**('147', 'Alberto', 'Errazuriz', 'AERRAZUR', '011.44.1344.429278', '1987-08-03', 'SA\_MAN', '12000.00', '0.30', '100', '80'),**

**('148', 'Gerald', 'Cambrault', 'GCAMBRAU', '011.44.1344.619268', '1987-08-04', 'SA\_MAN', '11000.00', '0.30', '100', '80'),**

**('149', 'Eleni', 'Zlotkey', 'EZLOTKEY', '011.44.1344.429018', '1987-08-05', 'SA\_MAN', '10500.00', '0.20', '100', '80'),**

**('150', 'Peter', 'Tucker', 'PTUCKER', '011.44.1344.129268', '1987-08-06', 'SA\_REP', '10000.00', '0.30', '145', '80'),**

**('151', 'David', 'Bernstein', 'DBERNSTE', '011.44.1344.345268', '1987-08-07', 'SA\_REP', '9500.00', '0.25', '145', '80'),**

**('152', 'Peter', 'Hall', 'PHALL', '011.44.1344.478968', '1987-08-08', 'SA\_REP', '9000.00', '0.25', '145', '80'),**

**('153', 'Christopher', 'Olsen', 'COLSEN', '011.44.1344.498718', '1987-08-09', 'SA\_REP', '8000.00', '0.20', '145', '80'),**

**('154', 'Nanette', 'Cambrault', 'NCAMBRAU', '011.44.1344.987668', '1987-08-10', 'SA\_REP', '7500.00', '0.20', '145', '80'),**

**('155', 'Oliver', 'Tuvault', 'OTUVAULT', '011.44.1344.486508', '1987-08-11', 'SA\_REP', '7000.00', '0.15', '145', '80'),**

**('156', 'Janette', 'King', 'JKING', '011.44.1345.429268', '1987-08-12', 'SA\_REP', '10000.00', '0.35', '146', '80'),**

**('157', 'Patrick', 'Sully', 'PSULLY', '011.44.1345.929268', '1987-08-13', 'SA\_REP', '9500.00', '0.35', '146', '80'),**

**('158', 'Allan', 'McEwen', 'AMCEWEN', '011.44.1345.829268', '1987-08-14', 'SA\_REP', '9000.00', '0.35', '146', '80'),**

**('159', 'Lindsey', 'Smith', 'LSMITH', '011.44.1345.729268', '1987-08-15', 'SA\_REP', '8000.00', '0.30', '146', '80'),**

**('160', 'Louise', 'Doran', 'LDORAN', '011.44.1345.629268', '1987-08-16', 'SA\_REP', '7500.00', '0.30', '146', '80'),**

**('161', 'Sarath', 'Sewall', 'SSEWALL', '011.44.1345.529268', '1987-08-17', 'SA\_REP', '7000.00', '0.25', '146', '80'),**

**('162', 'Clara', 'Vishney', 'CVISHNEY', '011.44.1346.129268', '1987-08-18', 'SA\_REP', '10500.00', '0.25', '147', '80'),**

**('163', 'Danielle', 'Greene', 'DGREENE', '011.44.1346.229268', '1987-08-19', 'SA\_REP', '9500.00', '0.15', '147', '80'),**

**('164', 'Mattea', 'Marvins', 'MMARVINS', '011.44.1346.329268', '1987-08-20', 'SA\_REP', '7200.00', '0.10', '147', '80'),**

**('165', 'David', 'Lee', 'DLEE', '011.44.1346.529268', '1987-08-21', 'SA\_REP', '6800.00', '0.10', '147', '80'),**

**('166', 'Sundar', 'Ande', 'SANDE', '011.44.1346.629268', '1987-08-22', 'SA\_REP', '6400.00', '0.10', '147', '80'),**

**('167', 'Amit', 'Banda', 'ABANDA', '011.44.1346.729268', '1987-08-23', 'SA\_REP', '6200.00', '0.10', '147', '80'),**

**('168', 'Lisa', 'Ozer', 'LOZER', '011.44.1343.929268', '1987-08-24', 'SA\_REP', '11500.00', '0.25', '148', '80'),**

**('169', 'Harrison', 'Bloom', 'HBLOOM', '011.44.1343.829268', '1987-08-25', 'SA\_REP', '10000.00', '0.20', '148', '80'),**

**('170', 'Tayler', 'Fox', 'TFOX', '011.44.1343.729268', '1987-08-26', 'SA\_REP', '9600.00', '0.20', '148', '80'),**

**('171', 'William', 'Smith', 'WSMITH', '011.44.1343.629268', '1987-08-27', 'SA\_REP', '7400.00', '0.15', '148', '80'),**

**('172', 'Elizabeth', 'Bates', 'EBATES', '011.44.1343.529268', '1987-08-28', 'SA\_REP', '7300.00', '0.15', '148', '80'),**

**('173', 'Sundita', 'Kumar', 'SKUMAR', '011.44.1343.329268', '1987-08-29', 'SA\_REP', '6100.00', '0.10', '148', '80'),**

**('174', 'Ellen', 'Abel', 'EABEL', '011.44.1644.429267', '1987-08-30', 'SA\_REP', '11000.00', '0.30', '149', '80'),**

**('175', 'Alyssa', 'Hutton', 'AHUTTON', '011.44.1644.429266', '1987-08-31', 'SA\_REP', '8800.00', '0.25', '149', '80'),**

**('176', 'Jonathon', 'Taylor', 'JTAYLOR', '011.44.1644.429265', '1987-09-01', 'SA\_REP', '8600.00', '0.20', '149', '80'),**

**('177', 'Jack', 'Livingston', 'JLIVINGS', '011.44.1644.429264', '1987-09-02', 'SA\_REP', '8400.00', '0.20', '149', '80'),**

**('178', 'Kimberely', 'Grant', 'KGRANT', '011.44.1644.429263', '1987-09-03', 'SA\_REP', '7000.00', '0.15', '149', '0'),**

**('179', 'Charles', 'Johnson', 'CJOHNSON', '011.44.1644.429262', '1987-09-04', 'SA\_REP', '6200.00', '0.10', '149', '80'),**

**('180', 'Winston', 'Taylor', 'WTAYLOR', '650.507.9876', '1987-09-05', 'SH\_CLERK', '3200.00', '0.00', '120', '50'),**

**('181', 'Jean', 'Fleaur', 'JFLEAUR', '650.507.9877', '1987-09-06', 'SH\_CLERK', '3100.00', '0.00', '120', '50'),**

**('182', 'Martha', 'Sullivan', 'MSULLIVA', '650.507.9878', '1987-09-07', 'SH\_CLERK', '2500.00', '0.00', '120', '50'),**

**('183', 'Girard', 'Geoni', 'GGEONI', '650.507.9879', '1987-09-08', 'SH\_CLERK', '2800.00', '0.00', '120', '50'),**

**('184', 'Nandita', 'Sarchand', 'NSARCHAN', '650.509.1876', '1987-09-09', 'SH\_CLERK', '4200.00', '0.00', '121', '50'),**

**('185', 'Alexis', 'Bull', 'ABULL', '650.509.2876', '1987-09-10', 'SH\_CLERK', '4100.00', '0.00', '121', '50'),**

**('186', 'Julia', 'Dellinger', 'JDELLING', '650.509.3876', '1987-09-11', 'SH\_CLERK', '3400.00', '0.00', '121', '50'),**

**('187', 'Anthony', 'Cabrio', 'ACABRIO', '650.509.4876', '1987-09-12', 'SH\_CLERK', '3000.00', '0.00', '121', '50'),**

**('188', 'Kelly', 'Chung', 'KCHUNG', '650.505.1876', '1987-09-13', 'SH\_CLERK', '3800.00', '0.00', '122', '50'),**

**('189', 'Jennifer', 'Dilly', 'JDILLY', '650.505.2876', '1987-09-14', 'SH\_CLERK', '3600.00', '0.00', '122', '50'),**

**('190', 'Timothy', 'Gates', 'TGATES', '650.505.3876', '1987-09-15', 'SH\_CLERK', '2900.00', '0.00', '122', '50'),**

**('191', 'Randall', 'Perkins', 'RPERKINS', '650.505.4876', '1987-09-16', 'SH\_CLERK', '2500.00', '0.00', '122', '50'),**

**('192', 'Sarah', 'Bell', 'SBELL', '650.501.1876', '1987-09-17', 'SH\_CLERK', '4000.00', '0.00', '123', '50'),**

**('193', 'Britney', 'Everett', 'BEVERETT', '650.501.2876', '1987-09-18', 'SH\_CLERK', '3900.00', '0.00', '123', '50'),**

**('194', 'Samuel', 'McCain', 'SMCCAIN', '650.501.3876', '1987-09-19', 'SH\_CLERK', '3200.00', '0.00', '123', '50'),**

**('195', 'Vance', 'Jones', 'VJONES', '650.501.4876', '1987-09-20', 'SH\_CLERK', '2800.00', '0.00', '123', '50'),**

**('196', 'Alana', 'Walsh', 'AWALSH', '650.507.9811', '1987-09-21', 'SH\_CLERK', '3100.00', '0.00', '124', '50'),**

**('197', 'Kevin', 'Feeney', 'KFEENEY', '650.507.9822', '1987-09-22', 'SH\_CLERK', '3000.00', '0.00', '124', '50'),**

**('198', 'Donald', 'OConnell', 'DOCONNEL', '650.507.9833', '1987-09-23', 'SH\_CLERK', '2600.00', '0.00', '124', '50'),**

**('199', 'Douglas', 'Grant', 'DGRANT', '650.507.9844', '1987-09-24', 'SH\_CLERK', '2600.00', '0.00', '124', '50'),**

**('200', 'Jennifer', 'Whalen', 'JWHALEN', '515.123.4444', '1987-09-25', 'AD\_ASST', '4400.00', '0.00', '101', '10'),**

**('201', 'Michael', 'Hartstein', 'MHARTSTE', '515.123.5555', '1987-09-26', 'MK\_MAN', '13000.00', '0.00', '100', '20'),**

**('202', 'Pat', 'Fay', 'PFAY', '603.123.6666', '1987-09-27', 'MK\_REP', '6000.00', '0.00', '201', '20'),**

**('203', 'Susan', 'Mavris', 'SMAVRIS', '515.123.7777', '1987-09-28', 'HR\_REP', '6500.00', '0.00', '101', '40'),**

**('204', 'Hermann', 'Baer', 'HBAER', '515.123.8888', '1987-09-29', 'PR\_REP', '10000.00', '0.00', '101', '70'),**

**('205', 'Shelley', 'Higgins', 'SHIGGINS', '515.123.8080', '1987-09-30', 'AC\_MGR', '12000.00', '0.00', '101', '110'),**

**('206', 'William', 'Gietz', 'WGIETZ', '515.123.8181', '1987-10-01', 'AC\_ACCOUNT', '8300.00', '0.00', '205', '110');**

**-- --------------------------------------------------------**

**--**

**-- Table structure for table `job\_history`**

**--**

**CREATE TABLE IF NOT EXISTS `job\_history` (**

**`EMPLOYEE\_ID` decimal(6,0) NOT NULL,**

**`START\_DATE` date NOT NULL,**

**`END\_DATE` date NOT NULL,**

**`JOB\_ID` varchar(10) NOT NULL,**

**`DEPARTMENT\_ID` decimal(4,0) DEFAULT NULL,**

**PRIMARY KEY (`EMPLOYEE\_ID`,`START\_DATE`),**

**KEY `JHIST\_DEPARTMENT\_IX` (`DEPARTMENT\_ID`),**

**KEY `JHIST\_EMPLOYEE\_IX` (`EMPLOYEE\_ID`),**

**KEY `JHIST\_JOB\_IX` (`JOB\_ID`)**

**) ENGINE=MyISAM DEFAULT CHARSET=latin1;**

**--**

**-- Dumping data for table `job\_history`**

**--**

**INSERT INTO `job\_history` (`EMPLOYEE\_ID`, `START\_DATE`, `END\_DATE`, `JOB\_ID`, `DEPARTMENT\_ID`) VALUES**

**('102', '1993-01-13', '1998-07-24', 'IT\_PROG', '60'),**

**('101', '1989-09-21', '1993-10-27', 'AC\_ACCOUNT', '110'),**

**('101', '1993-10-28', '1997-03-15', 'AC\_MGR', '110'),**

**('201', '1996-02-17', '1999-12-19', 'MK\_REP', '20'),**

**('114', '1998-03-24', '1999-12-31', 'ST\_CLERK', '50'),**

**('122', '1999-01-01', '1999-12-31', 'ST\_CLERK', '50'),**

**('200', '1987-09-17', '1993-06-17', 'AD\_ASST', '90'),**

**('176', '1998-03-24', '1998-12-31', 'SA\_REP', '80'),**

**('176', '1999-01-01', '1999-12-31', 'SA\_MAN', '80'),**

**('200', '1994-07-01', '1998-12-31', 'AC\_ACCOUNT', '90'),**

**('0', '0000-00-00', '0000-00-00', '', '0');**

**-- --------------------------------------------------------**

**--**

**-- Table structure for table `jobs`**

**--**

**CREATE TABLE IF NOT EXISTS `jobs` (**

**`JOB\_ID` varchar(10) NOT NULL DEFAULT '',**

**`JOB\_TITLE` varchar(35) NOT NULL,**

**`MIN\_SALARY` decimal(6,0) DEFAULT NULL,**

**`MAX\_SALARY` decimal(6,0) DEFAULT NULL,**

**PRIMARY KEY (`JOB\_ID`)**

**) ENGINE=MyISAM DEFAULT CHARSET=latin1;**

**--**

**-- Dumping data for table `jobs`**

**--**

**INSERT INTO `jobs` (`JOB\_ID`, `JOB\_TITLE`, `MIN\_SALARY`, `MAX\_SALARY`) VALUES**

**('AD\_PRES', 'President', '20000', '40000'),**

**('AD\_VP', 'Administration Vice President', '15000', '30000'),**

**('AD\_ASST', 'Administration Assistant', '3000', '6000'),**

**('FI\_MGR', 'Finance Manager', '8200', '16000'),**

**('FI\_ACCOUNT', 'Accountant', '4200', '9000'),**

**('AC\_MGR', 'Accounting Manager', '8200', '16000'),**

**('AC\_ACCOUNT', 'Public Accountant', '4200', '9000'),**

**('SA\_MAN', 'Sales Manager', '10000', '20000'),**

**('SA\_REP', 'Sales Representative', '6000', '12000'),**

**('PU\_MAN', 'Purchasing Manager', '8000', '15000'),**

**('PU\_CLERK', 'Purchasing Clerk', '2500', '5500'),**

**('ST\_MAN', 'Stock Manager', '5500', '8500'),**

**('ST\_CLERK', 'Stock Clerk', '2000', '5000'),**

**('SH\_CLERK', 'Shipping Clerk', '2500', '5500'),**

**('IT\_PROG', 'Programmer', '4000', '10000'),**

**('MK\_MAN', 'Marketing Manager', '9000', '15000'),**

**('MK\_REP', 'Marketing Representative', '4000', '9000'),**

**('HR\_REP', 'Human Resources Representative', '4000', '9000'),**

**('PR\_REP', 'Public Relations Representative', '4500', '10500');**

**-- --------------------------------------------------------**

**--**

**-- Table structure for table `locations`**

**--**

**CREATE TABLE IF NOT EXISTS `locations` (**

**`LOCATION\_ID` decimal(4,0) NOT NULL DEFAULT '0',**

**`STREET\_ADDRESS` varchar(40) DEFAULT NULL,**

**`POSTAL\_CODE` varchar(12) DEFAULT NULL,**

**`CITY` varchar(30) NOT NULL,**

**`STATE\_PROVINCE` varchar(25) DEFAULT NULL,**

**`COUNTRY\_ID` varchar(2) DEFAULT NULL,**

**PRIMARY KEY (`LOCATION\_ID`),**

**KEY `LOC\_CITY\_IX` (`CITY`),**

**KEY `LOC\_COUNTRY\_IX` (`COUNTRY\_ID`),**

**KEY `LOC\_STATE\_PROVINCE\_IX` (`STATE\_PROVINCE`)**

**) ENGINE=MyISAM DEFAULT CHARSET=latin1;**

**--**

**-- Dumping data for table `locations`**

**--**

**INSERT INTO `locations` (`LOCATION\_ID`, `STREET\_ADDRESS`, `POSTAL\_CODE`, `CITY`, `STATE\_PROVINCE`, `COUNTRY\_ID`) VALUES**

**('1000', '1297 Via Cola di Rie', '989', 'Roma', '', 'IT'),**

**('1100', '93091 Calle della Testa', '10934', 'Venice', '', 'IT'),**

**('1200', '2017 Shinjuku-ku', '1689', 'Tokyo', 'Tokyo Prefecture', 'JP'),**

**('1300', '9450 Kamiya-cho', '6823', 'Hiroshima', '', 'JP'),**

**('1400', '2014 Jabberwocky Rd', '26192', 'Southlake', 'Texas', 'US'),**

**('1500', '2011 Interiors Blvd', '99236', 'South San Francisco', 'California', 'US'),**

**('1600', '2007 Zagora St', '50090', 'South Brunswick', 'New Jersey', 'US'),**

**('1700', '2004 Charade Rd', '98199', 'Seattle', 'Washington', 'US'),**

**('1800', '147 Spadina Ave', 'M5V 2L7', 'Toronto', 'Ontario', 'CA'),**

**('1900', '6092 Boxwood St', 'YSW 9T2', 'Whitehorse', 'Yukon', 'CA'),**

**('2000', '40-5-12 Laogianggen', '190518', 'Beijing', '', 'CN'),**

**('2100', '1298 Vileparle (E)', '490231', 'Bombay', 'Maharashtra', 'IN'),**

**('2200', '12-98 Victoria Street', '2901', 'Sydney', 'New South Wales', 'AU'),**

**('2300', '198 Clementi North', '540198', 'Singapore', '', 'SG'),**

**('2400', '8204 Arthur St', '', 'London', '', 'UK'),**

**('2500', '"Magdalen Centre', ' The Oxford ', 'OX9 9ZB', 'Oxford', 'Ox'),**

**('2600', '9702 Chester Road', '9629850293', 'Stretford', 'Manchester', 'UK'),**

**('2700', 'Schwanthalerstr. 7031', '80925', 'Munich', 'Bavaria', 'DE'),**

**('2800', 'Rua Frei Caneca 1360', '01307-002', 'Sao Paulo', 'Sao Paulo', 'BR'),**

**('2900', '20 Rue des Corps-Saints', '1730', 'Geneva', 'Geneve', 'CH'),**

**('3000', 'Murtenstrasse 921', '3095', 'Bern', 'BE', 'CH'),**

**('3100', 'Pieter Breughelstraat 837', '3029SK', 'Utrecht', 'Utrecht', 'NL'),**

**('3200', 'Mariano Escobedo 9991', '11932', 'Mexico City', '"Distrito Federal', '"');**

**-- --------------------------------------------------------**

**--**

**-- Table structure for table `regions`**

**--**

**CREATE TABLE IF NOT EXISTS `regions` (**

**`REGION\_ID` decimal(5,0) NOT NULL,**

**`REGION\_NAME` varchar(25) DEFAULT NULL,**

**PRIMARY KEY (`REGION\_ID`),**

**UNIQUE KEY `sss` (`REGION\_NAME`)**

**) ENGINE=MyISAM DEFAULT CHARSET=latin1;**

**--**

**-- Dumping data for table `regions`**

**--**

**INSERT INTO `regions` (`REGION\_ID`, `REGION\_NAME`) VALUES**

**('1', 'Europe\r'),**

**('2', 'Americas\r'),**

**('3', 'Asia\r'),**

**('4', 'Middle East and Africa\r');**

**(For your reference the above commands has been given)**

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

**A.**

**create table countries(**

**country\_id int,**

**country\_name varchar(15),**

**region\_id int**

**);**

**2.** Write a SQL statement to create a simple table countries including columns country\_id,country\_name and region\_id which is already exists.

A. **create table countries(**

**Country\_id int,**

**Country\_name varchar(10),**

**Region\_id varchar(5)**

**);**

**-- It will show error because we have already table like countries name already exists**

**3.** Write a SQL statement to create the structure of a table dup\_countries similar to countries.

**A. create table dup\_countries select \* from countries where country\_id=23445;**

**4.** Write a SQL statement to create a duplicate copy of countries table including structure and data by name dup\_countries.

**A. create table dup\_countries select \* from countries;**

**5.**Write a SQL statement to create a table countries set a constraint NOT NULL for all the columns.

**A. create table countries(**

**country\_id int not null,**

**country\_name varchar(20) not null,**

**region\_id int not null**

**);**

**6.** 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.

A. **create table jobs(**

**Job\_id int,**

**Job\_title varchar(20),**

**Min\_salary int,**

**Max\_salary int,**

**Check(max\_salary<=25000)**

**);**

**7.**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

**A. create table countries(**

**Country\_id int,**

**Country\_name varchar(20),**

**Region\_id varchar(4),**

**Check(country\_name IN(‘italy’,’china’,’india’))**

**);**

**/\*suppose when they ask we don’t want expect that countries in that time**

**We used like check(country\_name NOT IN (‘italy’,’china’,’india’);**

**8.**Write a SQL statement to create a table named job\_histry including columns employee\_id, start\_date, end\_date, job\_id and department\_id and make sure that the value against column end\_date will be entered at the time of insertion to the format like '--/--/----'.

A. **create table job\_history(**

**Employee\_id int not null,**

**Start\_date date not null,**

**End\_date date not null,**

**Job\_id varchar(5) not null,**

**Department\_id int,**

**Check(end\_date LIKE(‘----/--/--‘))**

**);**

**9.**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.

A. **create table countries(**

**Country\_id int,**

**Country\_name varchar(15),**

**Region\_id varchar(4),**

**Unique(country\_id)**

**);**

**10.** 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.

A. **create table jobs3(**

**job\_id varchar(5) null,**

**job\_title varchar(20) default ‘ ’,**

**min\_salary decimal(6,0) default '8000',**

**max\_salary decimal(6,0) default null**

**);**

**/\* WHEN WE WANT TO KNOW ITS WORKING OR NOT DEFAULT VALUES YOU NEED TO SKIP THE COULUMS WHILE INSERTION**

**11.**Write a SQL statement to create a table named countries including columns country\_id, country\_name and region\_id and make sure that the country\_id column will be a key field which will not contain any duplicate data at the time of insertion.

A. **create table countries6(**

**country\_id varchar(5) null unique,**

**country\_name varchar(20) not null,**

**region\_id varchar(4)**

**);**

**12.**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.

**A. create table countries9(**

**country\_id int null unique auto\_increment,**

**country\_name varchar(20) not null,**

**region\_id varchar(4)**

**);**

**Normally auto increment start from integer 0 to nth number but when user need integer to start we use alter condition before inserting the values**

**-- alter table countries9 auto increment=431;**

**13.**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.

A. **create table countries(**

**Country\_id varchar(5) unique,**

**Country\_name varchar(20),**

**Region\_id varchar(4) unique**

**);**

**14.**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 | |

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

A.**create table job\_history4(**

**employee\_id decimal(6,0) unique,**

**start\_date date not null,**

**end\_date date not null,**

**job\_id varchar(10) not null,**

**department\_id decimal(5,0) not null,**

**foreign key(job\_id) references jobs(job\_id)**

**);**

**15.**Write a SQL statement to create a table employees including columns employee\_id, first\_name, last\_name, email, phone\_number hire\_date, job\_id, salary, commission, manager\_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 columns combined by department\_id and manager\_id columns contain only those unique combination values, which combinations are exists in the departments table.

Assume the structure of departments table below.

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

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

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

| DEPARTMENT\_ID | decimal(4,0) | NO | PRI | 0 | |

| DEPARTMENT\_NAME | varchar(30) | NO | | NULL | |

| MANAGER\_ID | decimal(6,0) | NO | PRI | 0 | |

| LOCATION\_ID | decimal(4,0) | YES | | NULL | |

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

A. **create table employees2(**

**employee\_id varchar(10) not null primary key,**

**first\_name varchar(20) not null,**

**last\_name varchar(20) not null,**

**email varchar(30) not null unique,**

**phone\_number int not null unique,**

**hire\_date date not null,**

**job\_id varchar(10) not null,**

**salary decimal(6,0) not null,**

**commission decimal(5,0) not null,**

**manager\_id decimal(6,0) not null unique,**

**department\_id decimal(4,0)not null unique,**

**foreign key(department\_id,manager\_id) references**

**departments(department\_id,manager\_id)**

**);**

**16.**Write a SQL statement to create a table employees including columns employee\_id, first\_name, last\_name, email, phone\_number hire\_date, job\_id, salary, commission, manager\_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 department\_id, reference by the column department\_id of departments table, can contain only those values which are exists in the departments table and another 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.

Assume that the structure of two tables departments and jobs.

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

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

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

| DEPARTMENT\_ID | decimal(4,0) | NO | PRI | 0 | |

| DEPARTMENT\_NAME | varchar(30) | NO | | NULL | |

| MANAGER\_ID | decimal(6,0) | YES | | NULL | |

| LOCATION\_ID | decimal(4,0) | YES | | NULL | |

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

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

| 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 | |

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

**A. create table employees9(**

**employee\_id varchar(10) not null unique,**

**first\_name varchar(15) not null,**

**last\_name varchar(25) not null,**

**phone\_number int not null unique,**

**email varchar(20) unique,**

**hire\_date date not null,**

**job\_id varchar(10) not null,**

**salary decimal(6,0) not null,**

**commission decimal(6,0) not null,**

**manager\_id decimal(6,0) null,**

**department\_id decimal(4,0) not null,**

**foreign key (department\_id) references departments(department\_id),**

**foreign key (job\_id) references jobs(job\_id)**

**);**

**17.**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 UPDATE CASCADE action allows you to perform cross-table update and ON DELETE RESTRICT action reject the deletion. The default action is ON DELETE RESTRICT.

Assume that the structure of the 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 | |

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

**A.** **create table employees10(**

**employee\_id varchar(15) not null unique,**

**first\_name varchar(25) not null unique,**

**last\_name varchar(20) not null unique,**

**job\_id varchar(10) not null,**

**salary decimal(6,0),**

**primary key(job\_id,employee\_id),**

**foreign key(job\_id) references jobs(job\_id) on update cascade on**

**delete restrict**

**);**

**/\* when we use ‘on update cascade’ clause, suppose when we update in parent table it will automatically update in child table and also**

**‘on delete restrict’ means when we want to remove a row in parent table it will display we could not delete like that error type /\***

**18.**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 CASCADE that lets you allow to delete records in the employees(child) table that refer to a record in the jobs(parent) table when the record in the parent table is deleted and the ON UPDATE RESTRICT actions reject any updates.

Assume that the structure of the 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 | |

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

**A. create table employees11(**

**employee\_id varchar(15) not null unique,**

**first\_name varchar(25) not null unique,**

**last\_name varchar(20) not null unique,**

**job\_id varchar(10) not null,**

**salary decimal(6,0),**

**primary key(job\_id,employee\_id),**

**foreign key(job\_id) references jobs(job\_id) on delete cascade on**

**update restrict**

**);**

**/\* here when we use ‘on delete cascade’ when remove parent table it remove child table records also and ‘on update restrict’ means when we update records in parent table it will display we cannot update records like that/\***

**19.**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 | |

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

**A.** **create table employees12(**

**employee\_id varchar(15) not null unique,**

**first\_name varchar(25) not null unique,**

**last\_name varchar(20) not null unique,**

**job\_id varchar(10) null,**

**salary decimal(6,0),**

**primary key(employee\_id),**

**foreign key(job\_id) references jobs(job\_id) on delete set null on**

**update set null**

**);**

**/\*in this ‘on delete set null’ means when we remove record in parent table automatically remove in child table but it won’t remove entire record in child table it remove id name instead of that it place ‘null’ same as like**

**‘on update set null’.**

**20.**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 | |

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

A. **create table employees13(**

**employee\_id varchar(15) not null unique,**

**first\_name varchar(25) not null unique,**

**last\_name varchar(20) not null unique,**

**job\_id varchar(10) null,**

**salary decimal(6,0),**

**primary key(employee\_id),**

**foreign key(job\_id) references jobs(job\_id) on delete no action on update no action**

**);**

**/\* in this we on delete no action and on update no action means when trying to delete/update a record in parent table it won’t update/delete because we use that clause/\***