```
CREATE DATABASE library;
SHOW DATABASES;
USE library;

-- Author Table
CREATE TABLE Author (
AuthorID INT PRIMARY KEY,
AuthorFirstName VARCHAR(30),
AuthorLastName VARCHAR(30),
AuthorNationality VARCHAR(50),
);

SELECT *
FROM Author;
--database is created

--Book Table
CREATE TABLE Book (
BookID INT PRIMARY KEY,
BookTitle VARCHAR(100),
AuthorID INT,
Genre VARCHAR(50),
FOREIGN KEY (AuthorID) REFERENCES Author(AuthorID)
);

SELECT *
FROM Book;
--database is created

--Client Table
CREATE TABLE Client (
ClientID INT PRIMARY KEY,
ClientFirstName VARCHAR(50),
ClientLastName VARCHAR(50),
ClientDOB DATE,
Occupation VARCHAR(50)
);

SELECT *
FROM Client;
--database is created

--Borrower table
CREATE TABLE Borrower (
BorrowID INT PRIMARY KEY,
ClientID INT,
BookID INT,
BorrowDate DATE,
FOREIGN KEY (ClientID) REFERENCES Client(ClientID),
FOREIGN KEY (BookID) REFERENCES Book(BookID)
);

Select *
From Borrower;
--database is created

-- Begin to input data into the Author Table
INSERT INTO Author (AuthorID, AuthorFirstName, AuthorLastName,
AuthorNationality)
VALUES
(1, 'Sofia', 'Smith', 'Canada'),
(2, 'Maria', 'Brown', 'Brazil'),
(3, 'Elena', 'Martin', 'Mexico'),
(4, 'Zoe', 'Roy', 'France'),
(5, 'Sebastian', 'Lavoie', 'Canada'),
(6, 'Dylan', 'Garcia', 'Spain'),
(7, 'Ian', 'Cruz', 'Mexico'),
(8, 'Lucas', 'Smith', 'USA'),
(9, 'Fabian', 'Wilson', 'USA'),
(10, 'Liam', 'Taylor', 'Canada'),
(11, 'William', 'Thomas', 'Great Britain'),
(12, 'Logan', 'Moore', 'Canada'),
(13, 'Oliver', 'Martin', 'France'),
(14, 'Alysha', 'Thompson', 'Canada'),
(15, 'Isabelle', 'Lee', 'Canada'),
(16, 'Emily', 'Clark', 'USA'),
(17, 'John', 'Young', 'China'),
(18, 'David', 'Wright', 'Canada'),
(19, 'Thomas', 'Scott', 'Canada'),
(20, 'Helena', 'Adams', 'Canada'),
(21, 'Sofia', 'Carter', 'USA'),
(22, 'Liam', 'Parker', 'Canada'),
(23, 'Emily', 'Murphy', 'USA');

Select *
From Author;
--data successfully intergrated

-- Begin to input data into the Book Table
INSERT INTO Book (BookID, BookTitle, AuthorID, Genre)
VALUES
(1, 'Build your database system', 1, 'Science'),
(2, 'The red wall', 2, 'Fiction'),
(3, 'The perfect match', 3, 'Fiction'),
(4, 'Digital Logic', 4, 'Science'),
(5, 'How to be a great lawyer', 5, 'Law'),
(6, 'Manage successful negotiations', 6, 'Society'),
(7, 'Pollution today', 7, 'Science'),
(8, 'A gray park', 2, 'Fiction'),
(9, 'How to be rich in one year', 8, 'Humor'),
(10, 'Their bright fate', 9, 'Fiction'),
(11, 'Black lines', 10, 'Fiction'),
(12, 'History of theater', 11, 'Literature'),
(13, 'Electrical transformers', 12, 'Science'),
(14, 'Build your big data system', 1, 'Science'),
(15, 'Right and left', 13, 'Children'),
(16, 'Programming using Python', 1, 'Science'),
(17, 'Computer networks', 14, 'Science'),
(18, 'Performance evaluation', 15, 'Science'),
(19, 'Daily exercise', 16, 'Well being'),
(20, 'The silver uniform', 17, 'Fiction'),
(21, 'Industrial revolution', 18, 'History'),
(22, 'Green nature', 19, 'Well being'),
(23, 'Perfect football', 20, 'Well being'),
(24, 'The chocolate love', 21, 'Humor'),
(25, 'Director and leader', 22, 'Society'),
(26, 'Play football every week', 20, 'Well being'),
(27, 'Maya the bee', 13, 'Children'),
(28, 'Perfect rugby', 20, 'Well being'),
(29, 'The end', 23, 'Fiction'),
(30, 'Computer security', 1, 'Science'),
(31, 'Participate', 22, 'Society'),
(32, 'Positive figures', 3, 'Fiction');

Select *
FROM BOOK;
--data successfully intergrated
```

```
-- Client table was created with ClientDOB date, which requires YYYY/MM/DD
-- So all Clients now share my birthday throughout various years
INSERT INTO Client (ClientID, ClientFirstName, ClientLastName, ClientDOB,
Occupation)
VALUES
(1, 'Kaiden', 'Hill', '2006-07-23', 'Student'),
(2, 'Alina', 'Morton', '2010-07-23', 'Student'),
(3, 'Fania', 'Brooks', '1983-07-23', 'Food Scientist'),
(4, 'Courtney', 'Jensen', '2006-07-23', 'Student'),
(5, 'Brittany', 'Hill', '1983-07-23', 'Firefighter'),
(6, 'Max', 'Rogers', '2005-07-23', 'Student'),
(7, 'Margaret', 'McCarthy', '1981-07-23', 'School Psychologist'),
(8, 'Julie', 'McCarthy', '1973-07-23', 'Professor'),
(9, 'Ken', 'McCarthy', '1974-07-23', 'Securities Clerk'),
(10, 'Britany', 'O''Quinn', '1984-07-23', 'Violinist'),
(11, 'Conner', 'Gardner', '1998-07-23', 'Licensed Massage Therapist'),
(12, 'Mya', 'Austin', '1960-07-23', 'Parquet Floor Layer'),
(13, 'Thierry', 'Rogers', '2004-07-23', 'Student'),
(14, 'Eloise', 'Rogers', '1984-07-23', 'Computer Security Manager'),
(15, 'Gerard', 'Jackson', '1979-07-23', 'Oil Exploration Engineer'),
(16, 'Randy', 'Day', '1986-07-23', 'Aircraft Electrician'),
(17, 'Jodie', 'Page', '1990-07-23', 'Manufacturing Director'),
(18, 'Coral', 'Rice', '1996-07-23', 'Window Washer'),
(19, 'Ayman', 'Austin', '2002-07-23', 'Student'),
(20, 'Jaxson', 'Austin', '1999-07-23', 'Repair Worker'),
(21, 'Joel', 'Austin', '1973-07-23', 'Police Officer'),
(22, 'Alina', 'Austin', '2010-07-23', 'Student'),
(23, 'Elin', 'Austin', '1962-07-23', 'Payroll Clerk'),
(24, 'Ophelia', 'Wolf', '2004-07-23', 'Student'),
(25, 'Eliot', 'McGuire', '1967-07-23', 'Dentist'),
(26, 'Peter', 'McKinney', '1968-07-23', 'Professor'),
(27, 'Annabella', 'Henry', '1974-07-23', 'Nurse'),
(28, 'Anastasia', 'Baker', '2001-07-23', 'Student'),
(29, 'Tyler', 'Baker', '1984-07-23', 'Police Officer'),
(30, 'Lilian', 'Ross', '1983-07-23', 'Insurance Agent'),
(31, 'Thierry', 'Arnold', '1975-07-23', 'Bus Driver'),
(32, 'Angelina', 'Rowe', '1979-07-23', 'Firefighter'),
(33, 'Marcia', 'Rowe', '1974-07-23', 'Health Educator'),
(34, 'Martin', 'Rowe', '1976-07-23', 'Ship Engineer'),
(35, 'Adeline', 'Rowe', '2005-07-23', 'Student'),
(36, 'Colette', 'Rowe', '1963-07-23', 'Professor'),
(37, 'Diane', 'Clark', '1975-07-23', 'Payroll Clerk'),
(38, 'Caroline', 'Clark', '1960-07-23', 'Dentist'),
(39, 'Dalton', 'Clayton', '1982-07-23', 'Police Officer'),
(40, 'Steve', 'Clayton', '1990-07-23', 'Bus Driver'),
(41, 'Melanie', 'Clayton', '1987-07-23', 'Computer Engineer'),
(42, 'Alana', 'Wilson', '2007-07-23', 'Student'),
(43, 'Carson', 'Byrne', '1995-07-23', 'Food Scientist'),
(44, 'Conrad', 'Byrne', '2007-07-23', 'Student'),
(45, 'Ryan', 'Porter', '2008-07-23', 'Student'),
(46, 'Elin', 'Porter', '1978-07-23', 'Computer Programmer'),
(47, 'Tyler', 'Harvey', '2007-07-23', 'Student'),
(48, 'Arya', 'Harvey', '2008-07-23', 'Student'),
(49, 'Serena', 'Harvey', '1978-07-23', 'School Teacher'),
(50, 'Lilly', 'Franklin', '1976-07-23', 'Doctor'),
(51, 'Mai', 'Franklin', '1994-07-23', 'Dentist'),
(52, 'John', 'Franklin', '1999-07-23', 'Firefighter'),
(53, 'Judy', 'Franklin', '1995-07-23', 'Firefighter'),
(54, 'Katy', 'Lloyd', '1992-07-23', 'School Teacher'),
(55, 'Tamara', 'Allen', '1963-07-23', 'Ship Engineer'),
(56, 'Maxim', 'Lyons', '1985-07-23', 'Police Officer'),
(57, 'Allan', 'Lyons', '1983-07-23', 'Computer Engineer'),
(58, 'Marc', 'Harris', '1980-07-23', 'School Teacher'),
(59, 'Elin', 'Young', '2009-07-23', 'Student'),
(60, 'Diana', 'Young', '2008-07-23', 'Student'),
(61, 'Diane', 'Young', '2006-07-23', 'Student'),
(62, 'Alana', 'Bird', '2003-07-23', 'Student'),
(63, 'Anna', 'Becker', '1979-07-23', 'Security Agent'),
(64, 'Katie', 'Grant', '1977-07-23', 'Manager'),
(65, 'Joan', 'Grant', '2010-07-23', 'Student'),
(66, 'Bryan', 'Bell', '2001-07-23', 'Student'),
(67, 'Belle', 'Miller', '1970-07-23', 'Professor'),
(68, 'Peggy', 'Stevens', '1990-07-23', 'Bus Driver'),
(69, 'Steve', 'Williamson', '1975-07-23', 'HR Clerk'),
(70, 'Tyler', 'Williamson', '1999-07-23', 'Doctor'),
(71, 'Izabelle', 'Williamson', '1990-07-23', 'Systems Analyst'),
(72, 'Annabel', 'Williamson', '1960-07-23', 'Cashier'),
(73, 'Mohamed', 'Waters', '1966-07-23', 'Insurance Agent'),
(74, 'Marion', 'Newman', '1970-07-23', 'Computer Programmer'),
(75, 'Ada', 'Williams', '1986-07-23', 'Computer Programmer'),
(76, 'Sean', 'Scott', '1983-07-23', 'Bus Driver'),
(77, 'Farrah', 'Scott', '1974-07-23', 'Ship Engineer'),
(78, 'Christine', 'Lambert', '1973-07-23', 'School Teacher'),
(79, 'Alysha', 'Lambert', '2007-07-23', 'Student'),
(80, 'Maia', 'Grant', '1984-01-01', 'School Teacher');

SELECT *
Client;
--data successfully intergrated

INSERT INTO Borrower (BorrowID, ClientID, BookID, BorrowDate)
VALUES
(1, 35, 17, '2016-07-20'),
(2, 1, 3, '2017-04-19'),
(3, 42, 8, '2016-10-03'),
(4, 62, 16, '2016-04-05'),
(5, 53, 13, '2017-01-17'),
(6, 33, 15, '2015-11-26'),
(7, 40, 14, '2015-01-21'),
(8, 64, 2, '2017-09-10'),
(9, 56, 30, '2017-08-02'),
(10, 23, 2, '2018-06-28'),
(11, 46, 19, '2015-11-18'),
(12, 61, 20, '2015-11-24'),
(13, 58, 7, '2017-06-17'),
(14, 46, 16, '2017-02-12'),
(15, 80, 21, '2018-03-18'),
(16, 51, 23, '2015-09-01'),
(17, 49, 18, '2015-07-28'),
(18, 43, 18, '2015-11-04'),
(19, 30, 2, '2018-08-10'),
(20, 48, 24, '2015-05-13'),
(21, 71, 5, '2016-09-05'),
(22, 35, 3, '2016-07-03'),
(23, 57, 1, '2015-03-17'),
(24, 23, 25, '2017-08-16'),
(25, 20, 12, '2018-07-24'),
(26, 25, 7, '2015-01-31'),
(27, 72, 29, '2016-04-10'),
(28, 74, 20, '2017-07-31'),
(29, 53, 14, '2016-02-20'),
(30, 32, 10, '2017-07-24'),
(31, 12, 15, '2018-04-25'),
(32, 77, 13, '2017-06-09'),
(33, 30, 4, '2017-10-24'),
(34, 37, 24, '2016-01-14'),
(35, 27, 26, '2017-06-05'),
(36, 1, 16, '2018-05-06'),
(37, 21, 9, '2016-03-19'),
(38, 69, 28, '2017-03-29'),
(39, 17, 19, '2017-03-14'),
(40, 8, 9, '2016-04-22'),
(41, 63, 18, '2015-01-25'),
(42, 65, 20, '2016-10-10'),
(43, 51, 19, '2015-07-28'),
(44, 23, 12, '2017-01-25'),
(45, 17, 4, '2017-04-18'),
(46, 68, 5, '2016-09-06'),
(47, 46, 13, '2017-09-30'),
(48, 15, 13, '2017-07-05'),
(49, 11, 19, '2017-12-14'),
(50, 78, 15, '2017-01-26'),
(51, 47, 9, '2015-03-03'),
(52, 68, 7, '2016-05-26'),
(53, 37, 26, '2017-02-06'),
(54, 48, 27, '2015-12-30'),
(55, 9, 21, '2017-10-21'),
(56, 29, 8, '2018-04-01'),
(57, 64, 18, '2017-08-29'),
(58, 61, 26, '2018-02-21'),
(59, 39, 28, '2016-07-26'),
(60, 73, 18, '2018-08-22'),
(61, 11, 13, '2018-01-17'),
(62, 45, 6, '2016-07-20'),
(63, 33, 13, '2018-03-18'),
(64, 10, 17, '2016-06-06'),
(65, 28, 18, '2017-02-17'),
(66, 51, 3, '2016-12-09'),
(67, 29, 2, '2015-09-18'),
(68, 28, 30, '2017-09-14'),
(69, 74, 20, '2015-12-12'),
(70, 15, 22, '2015-01-14'),
(71, 57, 8, '2017-08-20'),
(72, 2, 5, '2015-01-18'),
(73, 74, 12, '2018-04-14'),
(74, 51, 10, '2016-02-25'),
(75, 25, 17, '2015-02-24'),
(76, 45, 21, '2017-02-10'),
(77, 27, 25, '2016-08-03'),
(78, 32, 28, '2016-06-15'),
(79, 71, 21, '2017-05-21'),
(80, 75, 26, '2016-05-03'),
(81, 56, 32, '2015-12-23'),
(82, 26, 32, '2015-05-16'),
(83, 66, 32, '2015-05-30'),
(84, 57, 18, '2017-09-15'),
(85, 40, 15, '2016-09-02'),
(86, 65, 4, '2017-08-17'),
(87, 54, 7, '2015-12-19'),
(88, 29, 4, '2017-07-22'),
(89, 44, 9, '2017-12-31'),
(90, 56, 31, '2015-06-13'),
(91, 17, 4, '2015-04-01'),
(92, 35, 16, '2018-07-19'),
(93, 22, 18, '2017-06-22'),
(94, 39, 24, '2015-05-29'),
(95, 63, 14, '2018-01-20'),
(96, 53, 21, '2016-07-31'),
(97, 40, 9, '2016-07-10'),
(98, 52, 4, '2017-04-05'),
(99, 27, 20, '2016-09-04'),
(100, 72, 29, '2015-12-06'),
(101, 49, 16, '2017-12-19'),
(102, 6, 12, '2016-12-04'),
(103, 74, 31, '2016-07-27'),
(104, 48, 32, '2016-06-29'),
(105, 69, 2, '2016-12-27'),
(106, 60, 32, '2017-10-29'),
(107, 45, 22, '2017-06-12'),
(108, 42, 15, '2017-05-14'),
(109, 79, 8, '2016-10-13'),
(110, 70, 18, '2016-12-04'),
(111, 34, 8, '2016-03-06'),
(112, 43, 8, '2015-12-19'),
(113, 42, 32, '2016-04-20'),
(114, 67, 5, '2017-03-06'),
(115, 80, 25, '2015-06-23'),
(116, 54, 11, '2017-05-03'),
(117, 34, 28, '2017-08-30'),
(118, 65, 20, '2017-08-26'),
(119, 61, 19, '2018-01-05'),
(120, 38, 12, '2018-01-17'),
(121, 51, 4, '2016-05-13'),
(122, 7, 16, '2016-03-17'),
(123, 46, 16, '2016-11-25'),
(124, 75, 30, '2018-08-12'),
(125, 72, 32, '2015-03-12'),
(126, 44, 17, '2015-06-15'),
(127, 68, 15, '2016-02-21'),
(128, 21, 1, '2016-06-19'),
(129, 14, 25, '2016-10-10'),
(130, 68, 21, '2016-05-27'),
(131, 35, 20, '2015-03-19'),
(132, 16, 27, '2016-08-08'),
(133, 79, 31, '2018-03-07'),
(134, 14, 17, '2018-04-28'),
(135, 29, 28, '2018-03-11'),
(136, 41, 4, '2018-08-08'),
(137, 42, 3, '2016-02-23'),
(138, 45, 3, '2017-07-10'),
(139, 36, 16, '2018-07-19'),
(140, 36, 30, '2015-08-07'),
(141, 54, 32, '2018-03-14'),
(142, 61, 15, '2017-03-28'),
(143, 1, 13, '2018-05-17'),
(144, 43, 1, '2015-05-14'),
(145, 37, 14, '2015-07-30'),
(146, 62, 17, '2015-09-19'),
(147, 50, 22, '2016-12-02'),
(148, 45, 1, '2016-07-24'),
(149, 32, 17, '2018-03-10'),
(150, 13, 28, '2016-02-14'),
(151, 15, 9, '2018-08-11'),
(152, 10, 19, '2018-08-29'),
(153, 66, 3, '2016-11-27'),
(154, 68, 29, '2017-07-12'),
(155, 21, 14, '2018-06-27'),
(156, 35, 9, '2016-01-22'),
(157, 17, 24, '2016-08-25'),
(158, 40, 21, '2015-07-09'),
(159, 1, 24, '2016-03-28'),
(160, 70, 27, '2015-07-10'),
(161, 80, 26, '2016-04-24'),
(162, 29, 5, '2015-10-18'),
(163, 76, 12, '2018-04-25'),
(164, 22, 4, '2016-12-24'),
(165, 2, 2, '2017-10-26'),
(166, 35, 13, '2016-02-28'),
(167, 40, 8, '2017-10-02'),
(168, 68, 9, '2016-01-03'),
(169, 32, 5, '2016-11-13'),
(170, 34, 17, '2016-09-15'),
(171, 34, 16, '2018-04-13'),
(172, 80, 30, '2016-10-13'),
(173, 20, 32, '2015-11-17'),
(174, 36, 10, '2017-09-01'),
(175, 78, 12, '2018-06-27'),
(176, 57, 8, '2016-03-22'),
(177, 75, 11, '2017-06-27'),
(178, 71, 10, '2015-08-01'),
(179, 48, 22, '2015-09-29'),
(180, 19, 16, '2016-02-21'),
(181, 79, 30, '2018-08-20'),
(182, 70, 13, '2016-09-16'),
(183, 30, 6, '2017-02-10'),
(184, 45, 12, '2017-10-12'),
(185, 30, 27, '2016-11-23'),
(186, 26, 3, '2016-08-13'),
(187, 66, 6, '2017-01-14'),
(188, 47, 15, '2016-02-10'),
(189, 53, 30, '2018-08-08'),
(190, 80, 16, '2016-03-31'),
(191, 70, 13, '2018-02-03'),
(192, 14, 25, '2016-03-27'),
(193, 46, 22, '2016-01-13'),
(194, 30, 32, '2015-08-06'),
(195, 60, 14, '2016-11-27'),
(196, 14, 13, '2018-05-23'),
(197, 71, 15, '2016-06-22'),
(198, 38, 21, '2015-12-27'),
(199, 69, 30, '2017-04-29'),
(200, 49, 31, '2018-06-03'),
(201, 28, 28, '2015-05-29'),
(202, 49, 3, '2016-08-30'),
(203, 75, 1, '2015-10-29'),
(204, 78, 3, '2017-05-12'),
(205, 43, 18, '2015-03-25'),
(206, 27, 21, '2016-02-22'),
(207, 64, 22, '2015-04-03'),
(208, 21, 11, '2017-12-09'),
(209, 66, 29, '2016-12-20'),
(210, 45, 13, '2017-04-15'),
(211, 48, 30, '2015-01-31'),
(212, 20, 25, '2017-12-20'),
(213, 41, 20, '2018-01-29'),
(214, 51, 12, '2015-07-05'),
(215, 5, 1, '2015-04-12'),
(216, 40, 3, '2018-02-24'),
(217, 79, 4, '2018-06-27'),
(218, 15, 10, '2016-11-01'),
(219, 42, 22, '2016-12-28'),
(220, 17, 9, '2018-01-29'),
(221, 38, 13, '2016-05-09'),
(222, 79, 2, '2017-12-06'),
(223, 74, 3, '2015-12-07'),
(224, 46, 8, '2016-06-05'),
(225, 78, 22, '2018-08-11'),
(226, 45, 2, '2015-04-20'),
(227, 72, 31, '2015-11-11'),
(228, 18, 17, '2015-03-21'),
(229, 29, 3, '2017-08-13'),
(230, 66, 11, '2018-06-05'),
(231, 36, 16, '2016-04-28'),
(232, 26, 2, '2016-10-23'),
(233, 32, 1, '2017-10-31'),
(234, 62, 14, '2017-07-25'),
(235, 12, 4, '2015-07-08'),
(236, 38, 32, '2015-02-24'),
(237, 29, 16, '2016-07-28'),
(238, 36, 25, '2017-05-07'),
(239, 76, 7, '2015-06-13'),
(240, 28, 16, '2016-08-15'),
(241, 60, 13, '2016-08-26'),
(242, 8, 3, '2017-07-28'),
(243, 25, 1, '2016-07-30'),
(244, 62, 29, '2018-08-24'),
(245, 51, 8, '2016-09-01'),
(246, 27, 23, '2015-02-08'),
(247, 69, 12, '2018-06-25'),
(248, 51, 12, '2015-07-04'),
(249, 7, 4, '2015-05-01'),
(250, 31, 15, '2017-10-29'),
(251, 14, 23, '2015-01-15'),
(252, 14, 1, '2018-05-21'),
(253, 39, 25, '2015-12-26'),
(254, 79, 24, '2016-05-31'),
(255, 40, 15, '2016-03-18'),
(256, 51, 13, '2018-04-13'),
(257, 61, 1, '2015-02-11'),
(258, 15, 24, '2018-03-02'),
(259, 10, 22, '2018-01-21'),
(260, 67, 10, '2017-07-08'),
(261, 79, 11, '2016-12-11'),
(262, 19, 32, '2016-05-04'),
(263, 35, 11, '2017-08-01'),
(264, 27, 13, '2017-12-15'),
(265, 30, 22, '2015-12-22'),
(266, 8, 7, '2015-06-26'),
(267, 70, 9, '2016-03-20'),
(268, 56, 18, '2016-01-29'),
(269, 13, 19, '2015-03-06'),
(270, 61, 2, '2016-06-18'),
(271, 47, 13, '2017-09-18'),
(272, 30, 22, '2016-02-19'),
(273, 18, 22, '2016-12-31'),
(274, 34, 29, '2017-10-27'),
(275, 32, 21, '2015-06-03'),
(276, 9, 28, '2016-03-30'),
(277, 62, 24, '2015-03-23'),
(278, 44, 22, '2017-04-29'),
(279, 27, 5, '2015-03-25'),
(280, 61, 28, '2017-07-14'),
(281, 5, 13, '2016-12-04'),
(282, 43, 19, '2018-03-15'),
(283, 34, 19, '2016-06-05'),
(284, 35, 5, '2018-02-19'),
(285, 13, 12, '2016-09-23'),
(286, 74, 18, '2016-12-26'),
(287, 70, 31, '2017-08-15'),
(288, 42, 17, '2016-06-15'),
(289, 51, 24, '2018-07-30'),
(290, 45, 30, '2015-01-15'),
(291, 70, 17, '2017-10-07'),
(292, 77, 7, '2017-01-06'),
(293, 74, 25, '2015-09-25'),
(294, 47, 14, '2018-02-01'),
(295, 10, 2, '2017-04-18'),
(296, 16, 21, '2016-10-03'),
(297, 48, 5, '2016-09-17'),
(298, 72, 3, '2017-02-10'),
(299, 26, 23, '2016-03-01'),
(300, 49, 23, '2016-10-25');

Select *
FROM Borrower;
--data successfully intergrated


-- 1. Display all contents of the Clients table

SELECT * FROM Client; -- Grabs all data in Clients



-- 2. First names, last names, ages, and occupations of all clients

SELECT
ClientFirstName, ClientLastName, -- Gather Clients first and last name
TIMESTAMPDIFF(YEAR, ClientDOB, CURDATE()) AS Age, -- Calculate the age of the
client
based on their date of birth
Occupation
FROM
Client;



--3. First and last names of clients that borrowed books in March 2018

SELECT ClientFirstName, ClientLastName -- Gather Clients first and last name
FROM Client
INNER JOIN Borrower -- Join the Client and Borrower Tables
ON Client.ClientId = Borrower.ClientId
WHERE MONTH(BorrowDate) = 3 AND YEAR(BorrowDate) = 2018
GROUP BY Client.ClientId;



-- 4. First and last names of the top 5 authors clients borrowed in 2017

SELECT AuthorFirstName, AuthorLastName -- Gather Clients first and last name
FROM Author
INNER JOIN Book
ON Book.BookAuthor = Author.AuthorId -- Interjoin Book table and Author Table
INNER JOIN Borrower
ON Book.BookId = Borrower.BookId -- Interjoin Book table and Borrower Table
WHERE YEAR(BorrowDate) = 2017
GROUP BY Borrower.BookId
ORDER BY COUNT(Borrower.BookId) DESC LIMIT 5; -- Desc and limit the count to 5



-- 5. Nationalities of the least 5 authors that clients borrowed during the
years 2015-2017

SELECT Author.AuthorNationality
FROM Author
```

```
JOIN Book ON Author.AuthorID = Book.AuthorID -- Join the Author table and Book table
JOIN Borrower ON Book.BookID = Borrower.BookID -- Join the Book table and Borrower tables
WHERE YEAR(Borrower.BorrowDate) BETWEEN 2015 AND 2017 -- Used WHERE and BETWEEN to give dates between 2015 through 2017
GROUP BY Author.AuthorNationality
ORDER BY COUNT(Borrower.BookID) ASC LIMIT 5; -- Asc and limit the count to 5
```

```
-- 6. The book that was most borrowed during the years 2015-2017

SELECT Book.BookTitle
FROM Book
JOIN Borrower ON Book.BookID = Borrower.BookID -- Join the Book table and Borrower table
WHERE YEAR(Borrower.BorrowDate) BETWEEN 2015 AND 2017 -- Used WHERE and BETWEEN to give dates between 2015 through 2017
GROUP BY Book.BookID -- Group the results by the book ID
ORDER BY COUNT(Borrower.BookID) DESC LIMIT 1; -- Dec and limit count to 1 row, which will give the number 1 book
```

7. Top borrowed genres for clients born in years 1970-1980

```
SELECT Book.Genre, -- Select the genre of the book
COUNT(*) AS BorrowCount -- Count the number of borrowings for each genre
FROM Book -- Specify the table name "Book"
JOIN Borrower
ON Book.BookID = Borrower.BookID -- Join the Book table and Borrower table
JOIN Client
ON Borrower.ClientID = Client.ClientID -- Join the Borrower table and Client table
WHERE YEAR(Client.ClientDOB) BETWEEN 1970 AND 1980 -- Used WHERE and BETWEEN to give dates between 2015 through 2017
GROUP BY Book.Genre --
ORDER BY BorrowCount DESC; -- DECS order based on Borrow count
```

8. Top 5 occupations that borrowed the most in 2016

```
SELECT Client.Occupation
COUNT(*) AS BorrowCount -- Count the number of borrowings for each occupation
FROM Client
JOIN Borrower
ON Client.ClientID = Borrower.ClientID -- Join the Client table and Borrower table
JOIN Book
ON Borrower.BookID = Book.BookID -- Join the Borrower table and Book table
WHERE YEAR(Borrower.BorrowDate) = 2016 -- Used WHERE to filter to dates in 2016
GROUP BY Client.Occupation
ORDER BY BorrowCount DESC LIMIT 5; -- Decs and limit count to 5, which should be the Top 5 occupations
```

9. Average number of borrowed books by job title

```
SELECT Client.Occupation, ROUND(COUNT(Client.Occupation)/ COUNT(DISTINCT Borrower.ClientId))AS Average_Borrowed
```

```
-- Take occupations and count how many have that occupation, the round by
dividing how many books were borrowed by that occupation
FROM Client
INNER JOIN Borrower
ON Borrower.ClientId = Client.ClientId --  Innerjoin Borrower table and Client
table
GROUP BY Client.Occupation;
```

10. Create a VIEW and display the titles that were borrowed by at least 20% of clients

```
CREATE VIEW 'Books borrowed by Twenty Percent of Clients' AS
SELECT Book.BookTitle,
COUNT(DISTINCT Borrower.ClientID) AS NumClients -- Count the number of distinct
clients
who borrowed each book
FROM Book
JOIN Borrower
ON Book.BookID = Borrower.BookID -- Join the Book table and Borrower table
GROUP BY Book.BookID
HAVING COUNT(DISTINCT Borrower.ClientID) >= 0.2 * (SELECT COUNT(DISTINCT
ClientID) FROM Borrower);
 -- Removes titles that were borrowed by less than 20% of clients

SELECT * FROM 'Books borrowed by Twenty percent of Clients';
```

11. The top month of borrows in 2017

```
SELECT MONTH(BorrowDate) AS Top_Month, COUNT(MONTH(BorrowDate)) AS
Amount_of_books_borrowed
-- Selecting month with the Month() function, combine with the count function to
access the top months
FROM Borrower
WHERE YEAR(BorrowDate) = 2017 -- Narrows data to 2017
GROUP BY MONTH(BorrowDate)
ORDER BY Amount_of_books_borrowed DESC LIMIT 3; -- DESC and limit count to 3, 3
was picked due to a three way tie
```

12. Average number of borrows by age

```
SELECT DISTINCT (YEAR(NOW())-Client.ClientDoB) AS AGE,
ROUND(COUNT(Client.ClientId)/ COUNT(DISTINCT Borrower.ClientId))AS
Average_Borrowed
-- First function is the current year (with the YEAR() function) minus their
year of birth to caluclate age followed by the round and count funtion to find
an average books borrowed
FROM Client
INNER JOIN Borrower
ON Borrower.ClientId = Client.ClientId -- Innerjoin Borrower table and Client
table
GROUP BY AGE;
```

13. The oldest and the youngest clients of the library

```
SELECT MAX(YEAR(NOW())-Client.ClientDoB) AS Oldest,MIN(YEAR(NOW())-
Client.ClientDoB)AS Youngest
```

```
-- Similar to the function above, with included MAX() for largest results and
MIN() for smallest
FROM Client
INNER JOIN Borrower
ON Borrower.ClientId = Client.ClientId;
```

14. First and last names of authors that wrote books in more than one genre

```
SELECT Author.AuthorFirstName Author.AuthorLastName
FROM Author
JOIN Book
ON Author.AuthorID = Book.AuthorID -- Join the Author table and Book tables
GROUP BY Author.AuthorID
HAVING COUNT(DISTINCT Book.Genre) > 1; -- Creates a group of authors who have
books in more then one genre
```