Create a table authors in MySQL database with the following schema and constraints: (4 Marks)

authors(author_id, author_name, nationality, birth_year)

author_id (identifier for an author) is an integer attribute.

author name (name of the author) is a variable character attribute of length 25. nationality (nationality of the author) is a variable character attribute of length 15. birth_year (birth year of the author) is an integer attribute

Constraints to set for the table authors:

1 author id is the primary key

2. author name should not be empty

3. birth year must be between 1700 and 2000 and set its default value as 1900

Ans:

```
CREATE TABLE authors (
    author_id INT PRIMARY KEY,
    author_name VARCHAR(25) NOT NULL,
    nationality VARCHAR(15),
    birth_year INT DEFAULT 1900 CHECK (birth_year BETWEEN 1700 AND
2000)
);
```

Create a table books in MySQL database with the following schema and constraints: (4 Marks)

books(book_id, author_id, title, genre, price)

book_id (identifier for a book) is an integer attribure author_id (identifier for an author) is an integer attribute title (title of the book) is a variable character attribure of length 100 genre (genre of the book) is a variable character attribure of length 50. price (price of the book) is a a floating point attribute with 3 decimal points and 2 fractional points.

Constraints to set for the table books

1. book_id is the primary key

2. author id is the foreign key attribute that referes to

author_id attribute of authors table

3. title should not be empty and must be unique

4. price must be between 0 and 1000

Ans:

```
CREATE TABLE books (
    book_id INT PRIMARY KEY,
    author_id INT,
    title VARCHAR(100) NOT NULL UNIQUE,
    genre VARCHAR(50),
    price DECIMAL(5,2),
    CONSTRAINT fk_author
        FOREIGN KEY (author_id) REFERENCES authors(author_id),
    CONSTRAINT chk_price
        CHECK (price >= 0 AND price <= 1000)
);
```

Insert the following records to the table authors: (2 Marks)

(1, F. Scott Fitzgerald, American, 1896)

(2, Harper Lee, American, 1926)

(3, J.K. Rowling, British, 1965)

(4. J.R.R. Tolkien, British, 1892)

(5. Dan Brown, American, 1964)

(6, Stieg Larsson, Swedish, 1954)

(7, Suzanne Collins, American, 1962)

Ans:

```
INSERT INTO authors (id, name, nationality, birth_year) VALUES
(1, 'F. Scott Fitzgerald', 'American', 1896),
(2, 'Harper Lee', 'American', 1926),
(3, 'J.K. Rowling', 'British', 1965),
(4, 'J.R.R. Tolkien', 'British', 1892),
(5, 'Dan Brown', 'American', 1964),
(6, 'Stieg Larsson', 'Swedish', 1954),
(7, 'Suzanne Collins', 'American', 1962);
```

Insert the following records to the table books: (2 Marks)

(1, 1, The Great Gatsby, Fiction, 150.25)

(2, 2, To Kill a Mockingbird, Fiction, 230.70)

(3, 3, Harry Potter, Fantasy, 600.00)

(4, 4, The Hobbit, Fantasy, 720.00)

(5, 5, The Da Vinci Code, Mystery, 800.00)

(6, 6, The Girl with the Sword, Mystery, 950.00)

(7. 7, The Hunger Games, Sci-Fi, 120.00)

```
(8, 6, Pride and Prejudice, Fiction, 240.30)
Ans4:
INSERT INTO books (book_id, author_id, title, genre, price) VALUES
(1, 1, 'The Great Gatsby', 'Fiction', 150.25),
(2, 2, 'To Kill a Mockingbird', 'Fiction', 230.70),
(3, 3, 'Harry Potter', 'Fantasy', 600.00),
(4, 4, 'The Hobbit', 'Fantasy', 720.00),
(5, 5, 'The Da Vinci Code', 'Mystery', 800.00),
(6, 6, 'The Girl with the Sword', 'Mystery', 950.00),
(7, 7, 'The Hunger Games', 'Sci-Fi', 120.00),
(8, 6, 'Pride and Prejudice', 'Fiction', 240.30);

Question:5
books(book_id, title, price, author_id, genre_id)
authors(author_id, author_name, birth_year, nationality)
genres(genre_id, genre)
```

1. Retrieve all books and their corresponding authors (include book id, author name, title)

```
SELECT b.book_id, a.author_name, b.title
FROM books b
JOIN authors a ON b.author_id = a.author_id;
```

2. Display all authors whose name ends with 'n' (include only author name)

```
SELECT author_name
FROM authors
WHERE author_name LIKE '%n';
```

3. Get the details of the book with the highest price (include book id, title, price)

```
SELECT book_id, title, price
FROM books
ORDER BY price DESC
LIMIT 1;
```

4. Get the titles of all books written by American authors. (include book_id, title, author_name, nationality)

```
SELECT b.book_id, b.title, a.author_name, a.nationality
FROM books b
JOIN authors a ON b.author_id = a.author_id
WHERE a.nationality = 'American';
```

5. List the authors and their birth year ordered by the birth year in ascending order (include author name, birth year)

```
SELECT author_name, birth_year
FROM authors
ORDER BY birth_year ASC;
```

6. Get the details of the genre with the most books (include genre, total books)

```
SELECT g.genre, COUNT(b.book_id) AS total_books
FROM books b
JOIN genres g ON b.genre_id = g.genre_id
GROUP BY g.genre
ORDER BY total_books DESC
LIMIT 1;
```

7. Display genre and the total price (give 'total_price' for this column) of each genre with total price exceeds 1000

```
SELECT g.genre, SUM(b.price) AS total_price
FROM books b
JOIN genres g ON b.genre_id = g.genre_id
GROUP BY g.genre
HAVING SUM(b.price) > 1000;
```

8. Find the number of books written by each author (include author name, book count)

```
SELECT a.author_name, COUNT(b.book_id) AS book_count
FROM authors a
LEFT JOIN books b ON a.author_id = b.author_id
GROUP BY a.author_name;
```