

CS331 004 – Group11;

Mark Youssef

Tyler Ambrozy

Brian Doherty

CREATE TABLE BRANCH

(BLocation VARCHAR(100) NOT NULL,

BSize INT,

BCapacity INT,

PRIMARY KEY (BLocation));

CREATE TABLE EMPLOYEE

(EName VARCHAR(15) NOT NULL,

Ssn CHAR(9) NOT NULL,

date_hired DATE,

Address VARCHAR(30),

Emp_id CHAR(10),

Salary DECIMAL(10,2),

Assigned_f VARCHAR(100),

Dno INT NOT NULL,

PRIMARY KEY (Ssn),

FOREIGN KEY (Assigned_f) REFERENCES BRANCH(BLocation));

CREATE TABLE LIBRARIAN (

Emp_id CHAR(10),

School VARCHAR(100),

Degree_Date DATE,

PRIMARY KEY (Emp_id),

FOREIGN KEY (Emp_id) REFERENCES EMPLOYEE(Emp_id)

);

```
CREATE TABLE LIBRARIAN_SPECIALTY
( Emp_id CHAR(10) NOT NULL,
  Specialty VARCHAR(100) NOT NULL,
  PRIMARY KEY (Emp_id, Specialty),
  FOREIGN KEY (Emp_id) REFERENCES LIBRARIAN(Emp_id) );
```

```
CREATE TABLE PATRON
( PatronID INT NOT NULL,
  Name VARCHAR(100),
  Address VARCHAR(255),
  DateIssued DATE,
  FineBalance DECIMAL(10,2),
  ParentID INT,
  PRIMARY KEY (PatronID),
  FOREIGN KEY (ParentID) REFERENCES PATRON(PatronID) );
```

```
CREATE TABLE BOOK
( Cong_d VARCHAR(50) NOT NULL,
  Title VARCHAR(255),
  Authors VARCHAR(255),
  Publisher VARCHAR(255),
  PubDate DATE,
  LendablePeriod INT,
  PRIMARY KEY (Cong_d) );
```

```
CREATE TABLE BOOK_COPY
( CopyID INT NOT NULL,
  Cong_d VARCHAR(50) NOT NULL,
```

Format VARCHAR(20),
BLocation VARCHAR(100),
Status VARCHAR(50),
DueDate DATE,
PRIMARY KEY (CopyID),
FOREIGN KEY (Cong_d) REFERENCES BOOK(Cong_d),
FOREIGN KEY (BLocation) REFERENCES BRANCH(BLocation));

CREATE TABLE CHECKOUT

(CopyID INT NOT NULL,
PatronID INT NOT NULL,
CheckoutDate DATE NOT NULL,
DueDate DATE,
ReturnDate DATE,
PRIMARY KEY (CopyID, PatronID, CheckoutDate),
FOREIGN KEY (CopyID) REFERENCES BOOK_COPY(CopyID),
FOREIGN KEY (PatronID) REFERENCES PATRON(PatronID));

CREATE TABLE HOLD_REQUEST

(RequestID INT NOT NULL,
CopyID INT NOT NULL,
PatronID INT NOT NULL,
RequestDate DATE,
PRIMARY KEY (RequestID),
FOREIGN KEY (CopyID) REFERENCES BOOK_COPY(CopyID),
FOREIGN KEY (PatronID) REFERENCES PATRON(PatronID));

CREATE TABLE AUTHOR (

Author_id CHAR(10),

```
Name VARCHAR(100),  
Cong_d VARCHAR(50),  
PRIMARY KEY (Author_id),  
FOREIGN KEY (Cong_d) REFERENCES BOOK(Cong_d)  
);
```

BRANCH

```
INSERT INTO BRANCH (BLocation, BSize, BCapacity)  
VALUES ('Main Branch', 10000, 50000);
```

```
INSERT INTO BRANCH (BLocation, BSize, BCapacity)  
VALUES ('North Newark Branch', 5000, 20000);
```

```
INSERT INTO BRANCH (BLocation, BSize, BCapacity)  
VALUES ('South Ward Branch', 4000, 15000);
```

```
INSERT INTO BRANCH (BLocation, BSize, BCapacity)  
VALUES ('West Side Branch', 4500, 18000);
```

```
INSERT INTO BRANCH (BLocation, BSize, BCapacity)  
VALUES ('Science and Technology Branch', 6000, 25000);
```

BOOK

```
INSERT INTO BOOK (CONG_D, TITLE, AUTHORS, PUBLISHER, PUBDATE, LENDABLEPERIOD)  
VALUES ('QA76.73.P98', 'Python Crash Course', 'Eric Matthes', 'No Starch Press', TO_DATE('2019-05-03', 'YYYY-MM-DD'), 21);
```

```
INSERT INTO BOOK (CONG_D, TITLE, AUTHORS, PUBLISHER, PUBDATE, LENDABLEPERIOD)  
VALUES ('QA76.9.D3', 'Data Structures and Algorithms', 'Mark Allen Weiss', 'Pearson',  
TO_DATE('2021-08-20', 'YYYY-MM-DD'), 28);
```

```
INSERT INTO BOOK (CONG_D, TITLE, AUTHORS, PUBLISHER, PUBDATE, LENDABLEPERIOD)
VALUES ('QA76.73.J38', 'Effective Java', 'Joshua Bloch', 'Addison-Wesley', TO_DATE('2018-01-11',
'YYYY-MM-DD'), 14);
```

```
INSERT INTO BOOK (CONG_D, TITLE, AUTHORS, PUBLISHER, PUBDATE, LENDABLEPERIOD)
VALUES ('QA76.73.S67', 'C Programming: A Modern Approach', 'K. N. King', 'Norton X',
TO_DATE('2008-09-15', 'YYYY-MM-DD'), 21);
```

```
INSERT INTO BOOK (CONG_D, TITLE, AUTHORS, PUBLISHER, PUBDATE, LENDABLEPERIOD)
VALUES ('QA76.76.H94', 'Clean Code', 'Robert C. Martin', 'Prentice Hall', TO_DATE('2008-08-11',
'YYYY-MM-DD'), 30);
```

BOOK_COPY

```
INSERT INTO BOOK_COPY (COPYID, CONG_D, FORMAT, BLOCATION, STATUS, DUEDATE)
VALUES (2001, 'QA76.73.P98', 'Paperback', 'Main Branch', 'On Shelf', NULL);
```

```
INSERT INTO BOOK_COPY (COPYID, CONG_D, FORMAT, BLOCATION, STATUS, DUEDATE)
VALUES (2002, 'QA76.9.D3', 'Hardcover', 'Main Branch', 'Checked Out', TO_DATE('2025-04-22', 'YYYY-
MM-DD'));
```

```
INSERT INTO BOOK_COPY (COPYID, CONG_D, FORMAT, BLOCATION, STATUS, DUEDATE)
VALUES (2003, 'QA76.73.J38', 'Hardcover', 'Science Branch', 'On Shelf', NULL);
```

```
INSERT INTO BOOK_COPY (COPYID, CONG_D, FORMAT, BLOCATION, STATUS, DUEDATE)
VALUES (2004, 'QA76.73.S67', 'Paperback', 'Downtown Branch', 'Needs Maintenance', NULL);
```

```
INSERT INTO BOOK_COPY (COPYID, CONG_D, FORMAT, BLOCATION, STATUS, DUEDATE)
VALUES (2005, 'QA76.76.H94', 'Hardcover', 'Main Branch', 'Checked Out', TO_DATE('2025-04-25',
'YYYY-MM-DD'));
```

EMPLOYEE

```
INSERT INTO EMPLOYEE (ENAME, SSN, DATE_HIRED, ADDRESS, EMP_ID, SALARY, ASSIGNED_F, DNO)
```

```
VALUES ('John Doe', '123456789', TO_DATE('2015-03-10', 'YYYY-MM-DD'), '123 Main St, Newark, NJ', 'E1001', 55000.00, 'Main Branch', 1);
```

```
INSERT INTO EMPLOYEE (ENAME, SSN, DATE_HIRED, ADDRESS, EMP_ID, SALARY, ASSIGNED_F, DNO)
```

```
VALUES ('Jane Smith', '234567890', TO_DATE('2018-06-15', 'YYYY-MM-DD'), '456 North Ave, Newark, NJ', 'E1002', 40000.00, 'North Newark Branch', 2);
```

```
INSERT INTO EMPLOYEE (ENAME, SSN, DATE_HIRED, ADDRESS, EMP_ID, SALARY, ASSIGNED_F, DNO)
```

```
VALUES ('Carlos Gomez', '345678901', TO_DATE('2020-01-05', 'YYYY-MM-DD'), '789 South Rd, Newark, NJ', 'E1003', 35000.00, 'South Ward Branch', 3);
```

```
INSERT INTO EMPLOYEE (ENAME, SSN, DATE_HIRED, ADDRESS, EMP_ID, SALARY, ASSIGNED_F, DNO)
```

```
VALUES ('Emily Johnson', '456789012', TO_DATE('2017-09-01', 'YYYY-MM-DD'), '101 Science St, Newark, NJ', 'E1004', 60000.00, 'Science and Technology Branch', 1);
```

```
INSERT INTO EMPLOYEE (ENAME, SSN, DATE_HIRED, ADDRESS, EMP_ID, SALARY, ASSIGNED_F, DNO)
```

```
VALUES ('Robert Lee', '567890123', TO_DATE('2022-04-10', 'YYYY-MM-DD'), '202 West St, Newark, NJ', 'E1005', 32000.00, 'West Side Branch', 4);
```

PATERON

```
INSERT INTO PATRON (PATRONID, NAME, ADDRESS, DATEISSUED, FINEBALANCE, PARENTID)
```

```
VALUES (1, 'John Smith', '123 Elm St, Newark, NJ', TO_DATE('2020-06-15', 'YYYY-MM-DD'), 5.00, NULL);
```

```
INSERT INTO PATRON (PATRONID, NAME, ADDRESS, DATEISSUED, FINEBALANCE, PARENTID)
VALUES (2, 'Sarah Johnson', '456 Oak Ave, Newark, NJ', TO_DATE('2021-03-10', 'YYYY-MM-DD'), 0.00,
NULL);
```

```
INSERT INTO PATRON (PATRONID, NAME, ADDRESS, DATEISSUED, FINEBALANCE, PARENTID)
VALUES (3, 'Emily Smith', '123 Elm St, Newark, NJ', TO_DATE('2022-07-20', 'YYYY-MM-DD'), 2.50, 1);
```

```
INSERT INTO PATRON (PATRONID, NAME, ADDRESS, DATEISSUED, FINEBALANCE, PARENTID)
VALUES (4, 'Michael Brown', '789 Pine Rd, Newark, NJ', TO_DATE('2023-01-05', 'YYYY-MM-DD'), 0.00,
NULL);
```

```
INSERT INTO PATRON (PATRONID, NAME, ADDRESS, DATEISSUED, FINEBALANCE, PARENTID)
VALUES (5, 'Jake Brown', '789 Pine Rd, Newark, NJ', TO_DATE('2023-08-11', 'YYYY-MM-DD'), 1.25, 4);
```

```
INSERT INTO PATRON (PATRONID, NAME, ADDRESS, DATEISSUED, FINEBALANCE, PARENTID)
VALUES (6, 'Olivia Johnson', '456 Oak Ave, Newark, NJ', TO_DATE('2023-09-30', 'YYYY-MM-DD'), 0.00,
2);
```

```
HOLD_REQUEST
INSERT INTO HOLD_REQUEST (REQUESTID, COPYID, PATRONID, REQUESTDATE)
VALUES (101, 2001, 1, TO_DATE('2024-12-01', 'YYYY-MM-DD'));
```

```
INSERT INTO HOLD_REQUEST (REQUESTID, COPYID, PATRONID, REQUESTDATE)
VALUES (102, 2003, 3, TO_DATE('2025-01-15', 'YYYY-MM-DD'));
```

```
INSERT INTO HOLD_REQUEST (REQUESTID, COPYID, PATRONID, REQUESTDATE)
VALUES (103, 2002, 2, TO_DATE('2025-04-10', 'YYYY-MM-DD'));
```

```
INSERT INTO HOLD_REQUEST (REQUESTID, COPYID, PATRONID, REQUESTDATE)
VALUES (104, 2005, 5, TO_DATE('2025-04-12', 'YYYY-MM-DD'));
```

```
INSERT INTO HOLD_REQUEST (REQUESTID, COPYID, PATRONID, REQUESTDATE)
VALUES (105, 2004, 4, TO_DATE('2025-03-20', 'YYYY-MM-DD'));
```

```
LIBRARIAN
INSERT INTO LIBRARIAN (EMP_ID, SCHOOL, DEGREE_DATE)
VALUES ('E1001', 'Rutgers University', TO_DATE('2014-05-15', 'YYYY-MM-DD'));
```

```
INSERT INTO LIBRARIAN (EMP_ID, SCHOOL, DEGREE_DATE)
VALUES ('E1002', 'New Jersey Institute of Technology', TO_DATE('2016-12-20', 'YYYY-MM-DD'));
```

```
INSERT INTO LIBRARIAN (EMP_ID, SCHOOL, DEGREE_DATE)
VALUES ('E1003', 'Seton Hall University', TO_DATE('2019-06-01', 'YYYY-MM-DD'));
```

```
INSERT INTO LIBRARIAN (EMP_ID, SCHOOL, DEGREE_DATE)
VALUES ('E1004', 'Princeton University', TO_DATE('2017-05-25', 'YYYY-MM-DD'));
```

```
INSERT INTO LIBRARIAN (EMP_ID, SCHOOL, DEGREE_DATE)
VALUES ('E1005', 'Montclair State University', TO_DATE('2021-11-10', 'YYYY-MM-DD'));
```

```
LIBRARIAN_SPECIALITY
INSERT INTO LIBRARIAN_SPECIALITY (EMP_ID, SPECIALTY)
VALUES ('E1001', 'Children's Literature');
```

```
INSERT INTO LIBRARIAN_SPECIALITY (EMP_ID, SPECIALTY)
VALUES ('E1002', 'Community Outreach');
```

```
INSERT INTO LIBRARIAN_SPECIALITY (EMP_ID, SPECIALTY)
VALUES ('E1003', 'Digital Archives');
```



```
INSERT INTO LIBRARIAN_SPECIALITY (EMP_ID, SPECIALTY)
VALUES ('E1004', 'STEM Educational Programs');
```

```
INSERT INTO LIBRARIAN_SPECIALITY (EMP_ID, SPECIALTY)
VALUES ('E1005', 'Library Technology Systems');
```

```
AUTHOR
INSERT INTO AUTHOR (AUTHOR_ID, NAME, CONG_D)
VALUES ('A1001', 'Robert C. Martin', 'QA76.76.H94');
```

```
INSERT INTO AUTHOR (AUTHOR_ID, NAME, CONG_D)
VALUES ('A1002', 'K. N. King', 'QA76.73.S67');
```

```
INSERT INTO AUTHOR (AUTHOR_ID, NAME, CONG_D)
VALUES ('A1003', 'Eric Matthes', 'QA76.73.P98');
```

```
INSERT INTO AUTHOR (AUTHOR_ID, NAME, CONG_D)
VALUES ('A1004', 'Mark Allen Weiss', 'QA76.9.D3');
```

```
INSERT INTO AUTHOR (AUTHOR_ID, NAME, CONG_D)
VALUES ('A1005', 'Joshua Bloch', 'QA76.73.J38');
```

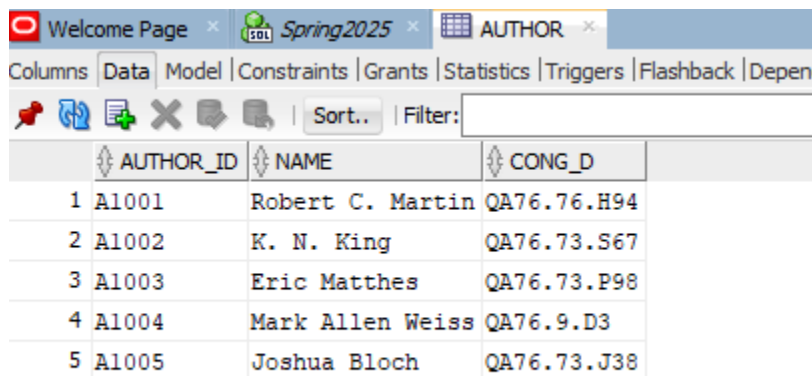
```
CHECKOUT
INSERT INTO CHECKOUT (COPYID, PATRONID, CHECKOUTDATE, DUE DATE, RETURN DATE)
VALUES (2001, 1, TO_DATE('2025-03-05', 'YYYY-MM-DD'), TO_DATE('2025-03-19', 'YYYY-MM-DD'),
TO_DATE('2025-03-15', 'YYYY-MM-DD'));
```

```
INSERT INTO CHECKOUT (COPYID, PATRONID, CHECKOUTDATE, DUE DATE, RETURN DATE)
VALUES (2002, 2, TO_DATE('2025-04-02', 'YYYY-MM-DD'), TO_DATE('2025-04-16', 'YYYY-MM-DD'),
NULL);
```

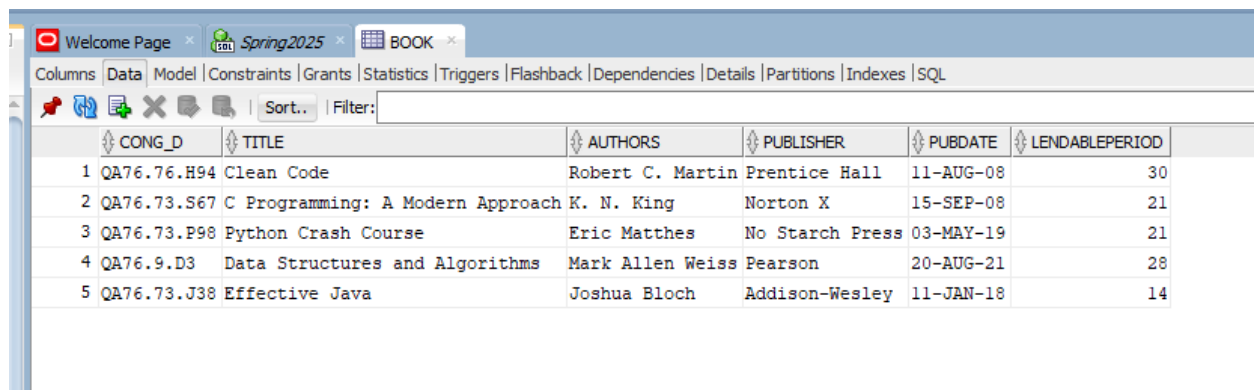
```
INSERT INTO CHECKOUT (COPYID, PATRONID, CHECKOUTDATE, DUE DATE, RETURN DATE)
VALUES (2003, 3, TO_DATE('2025-03-22', 'YYYY-MM-DD'), TO_DATE('2025-04-05', 'YYYY-MM-DD'),
TO_DATE('2025-03-30', 'YYYY-MM-DD'));
```

```
INSERT INTO CHECKOUT (COPYID, PATRONID, CHECKOUTDATE, DUE DATE, RETURN DATE)
VALUES (2004, 4, TO_DATE('2025-02-18', 'YYYY-MM-DD'), TO_DATE('2025-03-04', 'YYYY-MM-DD'),
TO_DATE('2025-03-02', 'YYYY-MM-DD'));
```

```
INSERT INTO CHECKOUT (COPYID, PATRONID, CHECKOUTDATE, DUE DATE, RETURN DATE)
VALUES (2005, 5, TO_DATE('2025-03-29', 'YYYY-MM-DD'), TO_DATE('2025-04-12', 'YYYY-MM-DD'),
NULL);
```



	AUTHOR_ID	NAME	CONG_D
1	A1001	Robert C. Martin	QA76.76.H94
2	A1002	K. N. King	QA76.73.S67
3	A1003	Eric Matthes	QA76.73.P98
4	A1004	Mark Allen Weiss	QA76.9.D3
5	A1005	Joshua Bloch	QA76.73.J38



	CONG_D	TITLE	AUTHORS	PUBLISHER	PUBDATE	LENDABLEPERIOD
1	QA76.76.H94	Clean Code	Robert C. Martin	Prentice Hall	11-AUG-08	30
2	QA76.73.S67	C Programming: A Modern Approach	K. N. King	Norton X	15-SEP-08	21
3	QA76.73.P98	Python Crash Course	Eric Matthes	No Starch Press	03-MAY-19	21
4	QA76.9.D3	Data Structures and Algorithms	Mark Allen Weiss	Pearson	20-AUG-21	28
5	QA76.73.J38	Effective Java	Joshua Bloch	Addison-Wesley	11-JAN-18	14







Welcome Page × Spring2025 × BOOK_COPY ×						
Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions Indexes SQL						
Sort.. Filter:						
	COPYID	CONG_D	FORMAT	BLOCATION	STATUS	DUEDATE
1	2001	QA76.73.P98	Paperback	Main Branch	On Shelf	(null)
2	2002	QA76.9.D3	Hardcover	Main Branch	Checked Out	22-APR-25
3	2005	QA76.76.H94	Hardcover	Main Branch	Checked Out	25-APR-25
4	2003	QA76.73.J38	Hardcover	Science and Technology Branch	On Shelf	(null)
5	2004	QA76.73.S67	Paperback	North Newark Branch	Needs Maintenance	(null)

Welcome Page × Spring2025 × BRANCH ×			
Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions Indexes SQL			
Sort.. Filter:			
	BLOCATION	BSIZE	BCAPACITY
1	Main Branch	10000	50000
2	North Newark Branch	5000	20000
3	South Ward Branch	4000	15000
4	West Side Branch	4500	18000
5	Science and Technology Branch	6000	25000

Welcome Page × Spring2025 × CHECKOUT ×					
Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions Indexes SQL					
Sort.. Filter:					
	COPYID	PATRONID	CHECKOUTDATE	DUEDATE	RETURNDATE
1	2001	1	05-MAR-25	19-MAR-25	15-MAR-25
2	2002	2	02-APR-25	16-APR-25	(null)
3	2003	3	22-MAR-25	05-APR-25	30-MAR-25
4	2004	4	18-FEB-25	04-MAR-25	02-MAR-25
5	2005	5	29-MAR-25	12-APR-25	(null)

Welcome Page ×Spring2025 ×EMPLOYEE ×

Columns | Data | Model | Constraints | Grants | Statistics | Triggers | Flashback | Dependencies | Details | Partitions | Indexes | SQL



Sort.. | Filter:

	ENAME	SSN	DATE_HIRED	ADDRESS	EMP_ID	SALARY	ASSIGNED_F	DNO
1	John Doe	123456789	10-MAR-15	123 Main St, Newark, NJ	E1001	55000	Main Branch	1
2	Jane Smith	234567890	15-JUN-18	456 North Ave, Newark, NJ	E1002	40000	North Newark Branch	2
3	Carlos Gomez	345678901	05-JAN-20	789 South Rd, Newark, NJ	E1003	35000	South Ward Branch	3
4	Emily Johnson	456789012	01-SEP-17	101 Science St, Newark, NJ	E1004	60000	Science and Technology Branch	1
5	Robert Lee	567890123	10-APR-22	202 West St, Newark, NJ	E1005	32000	West Side Branch	4

Welcome Page × Spring2025 × HOLD_REQUEST ×				
Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details				
Sort.. Filter:				
	REQUESTID	COPYID	PATRONID	REQUESTDATE
1	101	2001	1	01-DEC-24
2	102	2003	3	15-JAN-25
3	103	2002	2	10-APR-25
4	104	2005	5	12-APR-25
5	105	2004	4	20-MAR-25

Welcome Page × Spring2025 × LIBRARIAN ×			
Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions Indexes S			
Sort.. Filter:			
	EMP_ID	SCHOOL	DEGREE_DATE
1	E1001	Rutgers University	15-MAY-14
2	E1002	New Jersey Institute of Technology	20-DEC-16
3	E1003	Seton Hall University	01-JUN-19
4	E1004	Princeton University	25-MAY-17
5	E1005	Montclair State University	10-NOV-21

Welcome Page x Spring2025 x LIBRARIAN_SPECIALITY x		
Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details		
Sort.. Filter:		
EMP_ID	SPECIALTY	
1 E1001	Children's Literature	
2 E1002	Community Outreach	
3 E1003	Digital Archives	
4 E1004	STEM Educational Programs	
5 E1005	Library Technology Systems	

Welcome Page x Spring2025 x PATRON x						
Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions Indexes SQL						
Sort.. Filter:						
PATRONID	NAME	ADDRESS	DATEISSUED	FINEBALANCE	PARENTID	
1	1 John Smith	123 Elm St, Newark, NJ	15-JUN-20	5	(null)	
2	2 Sarah Johnson	456 Oak Ave, Newark, NJ	10-MAR-21	0	(null)	
3	3 Emily Smith	123 Elm St, Newark, NJ	20-JUL-22	2.5	1	
4	4 Michael Brown	789 Pine Rd, Newark, NJ	05-JAN-23	0	(null)	
5	5 Jake Brown	789 Pine Rd, Newark, NJ	11-AUG-23	1.25	4	

-- Update BSize for the 'Main Branch'

UPDATE BRANCH

SET BSize = 12000

WHERE BLocation = 'Main Branch';

-- Delete the 'South Ward Branch'

DELETE FROM BRANCH

WHERE BLocation = 'West Side Branch';

-- Update the specialty for librarian with Emp_id 'E1003'

UPDATE LIBRARIAN_SPECIALITY

SET Specialty = 'Advanced Programming'

WHERE Emp_id = 'E1003' AND Specialty = 'Digital Archives';

-- Delete the specialty for librarian with Emp_id 'E1004'

DELETE FROM LIBRARIAN_SPECIALITY

WHERE Emp_id = 'E1004' AND Specialty = 'STEM Educational Programs';

-- Update the salary of the employee with Ssn '123456789'

UPDATE EMPLOYEE

SET Salary = 60000

WHERE Ssn = '123456789';

-- Delete the employee with Ssn '234567890'

DELETE FROM EMPLOYEE

WHERE Ssn = '234567890';

-- Update the RequestDate for the hold request with RequestID 101

UPDATE HOLD_REQUEST

SET RequestDate = TO_DATE('2025-04-05', 'YYYY-MM-DD')

WHERE RequestID = 101;

-- Delete the hold request with RequestID 105

DELETE FROM HOLD_REQUEST

WHERE RequestID = 105;

-- Update the ReturnDate for checkout with CopyID 2002 and PatronID 2

UPDATE CHECKOUT

SET ReturnDate = TO_DATE('2025-04-10', 'YYYY-MM-DD')

WHERE CopyID = 2002 AND PatronID = 2;

-- Delete the checkout record for CopyID 2001 and PatronID 1

DELETE FROM CHECKOUT

WHERE CopyID = 2001 AND PatronID = 1;

-- Update the name of the author with Author_id 'A1001'

UPDATE AUTHOR

SET Name = 'Robert C. Martin (Updated)'

WHERE Author_id = 'A1001';

-- Delete the author with Author_id 'A1002'

DELETE FROM AUTHOR

WHERE Author_id = 'A1002';

-- Update the status of the book copy with CopyID 2003

UPDATE BOOK_COPY

SET Status = 'Checked Out'

WHERE CopyID = 2003;

-- Delete the book copy with CopyID 2004

DELETE FROM BOOK_COPY

WHERE CopyID = 2004;

-- Update the publisher for the book with Cong_d 'QA76.73.S67'

UPDATE BOOK

SET Publisher = 'Prentice Hall'

WHERE Cong_d = 'QA76.73.S67';

-- Delete the book with Cong_d 'QA76.73.P98'

DELETE FROM BOOK

WHERE Cong_d = 'QA76.73.P98';

-- Update the FineBalance for Patron with PatronID 3

UPDATE PATRON

SET FineBalance = 3.00

WHERE PatronID = 3;

-- Delete the patron with PatronID 5

DELETE FROM PATRON

WHERE PatronID = 5;

-- Update the Degree_Date of librarian with Emp_id 'E1001'

UPDATE LIBRARIAN

SET Degree_Date = TO_DATE('2015-06-01', 'YYYY-MM-DD')

WHERE Emp_id = 'E1001';

-- Delete the librarian with Emp_id 'E1002'

DELETE FROM LIBRARIAN

WHERE Emp_id = 'E1002';

Welcome Page x Spring2025 x AUTHOR x			
Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions Indexes SQL			
Sort.. Filter:			
	AUTHOR_ID	NAME	CONG_D
1	A1001	Robert C. Martin (Updated)	QA76.76.H94
2	A1003	Eric Matthes	QA76.73.P98
3	A1004	Mark Allen Weiss	QA76.9.D3
4	A1005	Joshua Bloch	QA76.73.J38

Welcome Page

Spring2025

BOOK

Columns

Data

Model

Constraints

Grants

Statistics

Triggers

Flashback

Dependencies

Details

Partitions

Indexes

SQL

Sort..

Filter:

	CONG_D	TITLE	AUTHORS	PUBLISHER	PUBDATE	LENDABLEPERIOD
1	QA76.76.H94	Clean Code	Robert C. Martin	Prentice Hall	11-AUG-08	30
2	QA76.73.P98	Python Crash Course	Eric Matthes	Prentice Hall	03-MAY-19	21
3	QA76.9.D3	Data Structures and Algorithms	Mark Allen Weiss	Pearson	20-AUG-21	28
4	QA76.73.J38	Effective Java	Joshua Bloch	Addison-Wesley	11-JAN-18	14

Welcome Page xSpring2025 xBOOK_COPY x

Columns | Data | Model | Constraints | Grants | Statistics | Triggers | Flashback | Dependencies | Details | Partitions | Indexes | SQL

Welcome Page × Spring2025 × BRANCH ×			
Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions Indexes SQL			
Sort.. Filter:			
	BLOCATION	BSIZE	BCAPACITY
1	Main Branch	11000	50000
2	North Newark Branch	5000	20000
3	South Ward Branch	4000	15000
4	West Side Branch	4500	18000
5	Science and Technology Branch	6000	25000

Welcome Page × Spring2025 × CHECKOUT ×					
Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions Indexes SQL					
Sort.. Filter:					
	COPYID	PATRONID	CHECKOUTDATE	DUE DATE	RETURNDATE
1	2002	2	02-APR-25	16-APR-25	10-APR-25
2	2003	3	22-MAR-25	05-APR-25	30-MAR-25
3	2001	1	18-FEB-25	04-MAR-25	02-MAR-25
4	2005	5	29-MAR-25	12-APR-25	(null)

Welcome Page × Spring2025 × EMPLOYEE ×								
Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions Indexes SQL								
Sort.. Filter:								
	ENAME	SSN	DATE_HIRED	ADDRESS	EMP_ID	SALARY	ASSIGNED_F	DNO
1	John Doe	123456789	10-MAR-15	123 Main St, Newark, NJ	E1001	60000	Main Branch	1
2	Jane Smith	234567890	15-JUN-18	456 North Ave, Newark, NJ	E1002	40000	North Newark Branch	2
3	Carlos Gomez	345678901	05-JAN-20	789 South Rd, Newark, NJ	E1003	35000	South Ward Branch	3
4	Robert Lee	567890123	10-APR-22	202 West St, Newark, NJ	E1005	32000	Main Branch	4

Welcome Page Spring2025 PATRON						
Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions Indexes SQL						
Sort.. Filter:						
	PATRONID	NAME	ADDRESS	DATEISSUED	FINEBALANCE	PARENTID
1	1	John Smith	123 Elm St, Newark, NJ	15-JUN-20	5	(null)
2	2	Sarah Johnson	456 Oak Ave, Newark, NJ	10-MAR-21	0	(null)
3	3	Emily Smith	123 Elm St, Newark, NJ	20-JUL-22	3	1
4	5	Jake Brown	789 Pine Rd, Newark, NJ	11-AUG-23	1.25	2

```
SELECT BLocation, COUNT(*) AS NumCopies
FROM BOOK_COPY
GROUP BY BLocation;
```

Script Output Query Result	
Task completed in 0.027 seconds	
More Details : https://docs.oracle.com/error-help/db/ora-02292/	
1 row updated.	
1 row deleted.	
BLOCATION	NUMCOPIES
Main Branch	3
Science and Technology Branch	1

```
SELECT BLocation, COUNT(*) AS NumCopies
FROM BOOK_COPY
GROUP BY BLocation
HAVING COUNT(*) > 1;
```

Script Output x Query Result x	
Task completed in 0.043 seconds	
1 row deleted.	
BLOCATION	NUMCOPIES
Main Branch	3
Science and Technology Branch	1
BLOCATION	NUMCOPIES
Main Branch	3

SELECT Name, FineBalance

FROM PATRON P1

WHERE FineBalance > ALL (

SELECT FineBalance

FROM PATRON P2

WHERE P1.Address = P2.Address AND P1.PatronID <> P2.PatronID

);

Script Output x Query Result x	
Task completed in 0.052 seconds	
Science and Technology Branch	1
BLOCATION	NUMCOPIES
Main Branch	3
NAME	FINEBALANCE
John Smith	5
Sarah Johnson	0
Jake Brown	1.25

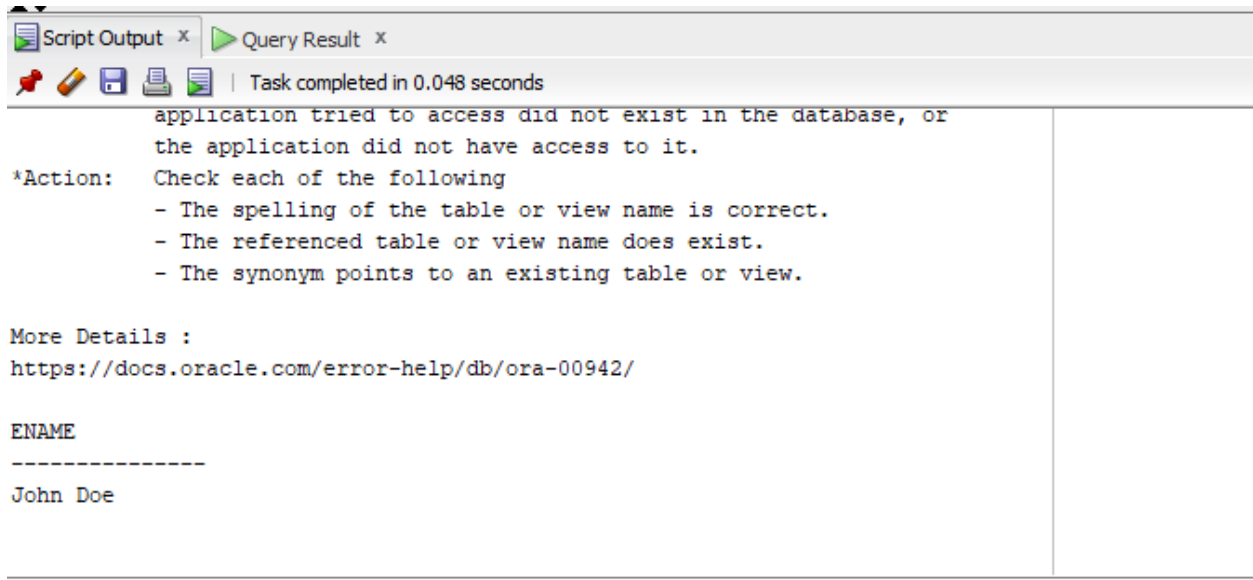
SELECT EName

FROM EMPLOYEE

WHERE Emp_id IN (

SELECT Emp_id

```
FROM LIBRARIAN_SPECIALITY
WHERE Specialty IN (
    SELECT Specialty
    FROM LIBRARIAN_SPECIALITY
    WHERE Emp_id = 'E1001'
)
);
```



The screenshot shows a database client window with two tabs: "Script Output" and "Query Result". The "Script Output" tab is active, displaying an error message. The message states that an application tried to access a table or view that does not exist in the database, or that the application did not have access to it. It provides a list of actions to check: the spelling of the table or view name, the existence of the referenced table or view, and whether the synonym points to an existing table or view. Below the error message, it says "More Details :" and provides a URL: <https://docs.oracle.com/error-help/db/ora-00942/>. At the bottom, it shows the username "ENAME" and a password prompt "-----" followed by the name "John Doe".

Script Output x Query Result x

Task completed in 0.048 seconds

application tried to access did not exist in the database, or
the application did not have access to it.

*Action: Check each of the following

- The spelling of the table or view name is correct.
- The referenced table or view name does exist.
- The synonym points to an existing table or view.

More Details :
<https://docs.oracle.com/error-help/db/ora-00942/>

ENAME

John Doe