**SQL Statements**

**Credentials:**

// read $Username, $Password

**EXISTS** ( **SELECT** \*

**FROM** USER **AS** U

**WHERE** U.Username = $Username **AND** U.Password = $Password);

**Create Account:**

// read $Username, $Password

**INSERT INTO** USER (Username, Password)

**VALUES** ($Username, $Password);

**Create Profile:**

// read $Username, $Name, $DOB, $Email, $IsDebarred, $Gender, $Address

// assume $IsFaculty, $Penalty, $Dept are managed by application

// assume dropdowns of “Gender” and “Associated Department” are populated by application

**INSERT INTO** STUDENT\_FACULTY (Username, Name, DOB, Email, IsDebarred, Gender, Address, IsFaculty, Penalty, Dept)

**VALUES** ($Username, $Name, $DOB, $Email, $IsDebarred, $Gender, $Address, $IsFaculty, $Penalty, $Dept);

**Search:**

// read $ISBN, $Title, $Author, $Publisher, $Edition

**SELECT** B.ISBN, B.Title, B.Edition, B.IsReserved **COUNT** (C.CopyID)

**FROM** BOOK **AS** B **INNER JOIN** BOOKCOPY **AS** C **ON** B.ISBN = C.ISBN

**WHERE** (B.ISBN = $ISBN **AND** B.Title = $Title **AND** B.Author = $Author **AND** B.Publisher = $Publisher **AND** B.Edition = $Edition) **AND** (IsChecked = FALSE **AND** IsHold = FALSE **AND** IsDamaged = FALSE)

**GROUP BY** C.ISBN;

**Locate and Return:**

// read $ISBN

**SELECT** S.FloorID, B.ShelfID, S.AisleID, B.SubName

**FROM** BOOK **AS** B **INNER JOIN** SHELF **AS** S **ON** B.ShelfID = S.ShelfID

**WHERE** B.ISBN = $ISBN;

**Confirm**:

// assume $ISBN , $CopyID and $Username are read from scanner

// assume IsDebarred is managed by application

// after pressing “confirm”

**UPDATE** BOOKCOPY

**SET** IsChecked = TRUE, IsHold = FALSE

**WHERE** BOOKCOPY.ISBN = $ISBN **AND** BOOKCOPY.CopyID = $CopyID;

// assume $ReturnDate is either equal to ($CheckoutDate + 14) or $LastAllowedDate

// assume $CheckoutDate is auto-populated as the current date

// assume $LastAllowedDate is managed by application based on the maximum number of days allowed to him and the maximum number of extensions allowed to him

**UPDATE** ISSUE

**SET** ReturnDate = $ReturnDate, IssueDate = $CheckoutDate

**WHERE** ISSUE.Username = $Username **AND** ISSUE.ISBN = $ISBN **AND** ISSUE.CopyID = $CopyID;

**Return:**

// assume $ISBN , $CopyID and $Username are read from scanner

// assume dropdowns of “Return in Damaged Condition” are populated by application

// read $IsDamaged, and convert to boolean

**UPDATE** BOOKCOPY

**SET** IsChecked = FALSE, IsDamaged = $IsDamaged

**WHERE** BOOKCOPY.ISBN = $ISBN **AND** BOOKCOPY.CopyID = $CopyID;

**Submit:**

// assume $ISBN and $CopyID are managed by staff

//after pressing “Look for the last user”

// get the username and return date for this book

**V1: CREATE VIEW** ISSUE1

**AS SELECT** I.Username, I.ReturnDate

**FROM** ISSUE **AS** I

**WHERE** I.ISBN = $ISBN **AND** I.CopyID = $CopyId;

// get the last return date

**V2**: **CREATE VIEW** ISSUE2

**AS SELECT MAX**(S.ReturnDate) **AS** ReturnDate

**FROM** ISSUE1 **AS** S;

// get the last user

**QV1**: **SELECT** U.Username

**FROM** ISSUE1 **AS** U

**WHERE** U.ReturnDate = ISSUE2.ReturnDate;

**V1A**: **DROP** **VIEW** ISSUE1;

**V2A**: **DROP** **VIEW** ISSUE2;

// assume $Penalty is managed by staff

// assume $IsDebarred is managed by application

**UPDATE** STUDENT\_FACULTY

**SET** Penalty = Penalty + $Penalty, IsDebarred = $IsDebarred

**WHERE** STUDENT\_FACULTY.Username = $Username;

**Generate Report:**

// **Damaged Books Report**

//read $Month, $SubName1, $SubName2, $SubName3

**V1: CREATE VIEW** ISSUE1

**AS SELECT** I.ISBN, I.CopyID, **MONTH**(I.IssueDate) **AS** Month

**FROM** ISSUE **AS** I;

// select books based on $Month

**V2: CREATE VIEW** ISSUE2

**AS SELECT** S.ISBN, S.CopyID, S.IssueDate

**FROM** ISSUE1 **AS** S

**WHERE** S. Month = $Month;

//select bookcopies based on $SubName

**V3: CREATE VIEW** ISSUE3

**AS SELECT** U.ISBN, U.CopyID, U. Month, B.SubName

**FROM** ISSUE2 **AS** U **INNER JOIN** BOOK **AS** B **ON** U.ISBN = B.ISBN

**WHERE** B.SubName = $SubName1 **OR** B.SubName = $SubName2 **OR** B.SubName = $SubName3;

// select damaged books

**V4: CREATE VIEW** ISSUE4

**AS SELECT** E.ISBN, E.CopyID, E. Month, E.SubName, C.IsDamaged

**FROM** ISSUE3 **AS** E **INNER JOIN** BOOKCOPY **AS** C **ON** (E.ISBN = C.ISBN **AND** E.CopyID = C.CopyID)

**WHERE** C.IsDamaged = TRUE;

//count damaged books

**QV4**: **SELECT** A.Month, A.SubName, COUNT(\*) **AS** #damaged\_books

**FROM** ISSUE4 **AS** A

**GROUP BY** A.SubName;

**V1A**: **DROP** **VIEW** ISSUE1;

**V2A**: **DROP** **VIEW** ISSUE2;

**V3A**: **DROP** **VIEW** ISSUE3;

**V4A**: **DROP** **VIEW** ISSUE4;

//**Popular Books Report**

**V1: CREATE VIEW** ISSUE1

**AS SELECT** I.ISBN, I.IssueDate, **MONTH**(I.IssueDate) **AS** Month

**FROM** ISSUE **AS** I;

**QV1: SELECT** U. Month, B.Title, COUNT(\*) **AS** #checkouts

**FROM** ISSUE1 **AS** U **INNER JOIN** BOOK **AS** B **ON** U.ISBN = B.ISBN

**WHERE** U.Month = 1 **OR** U.Month = 2

**GROUP BY** U.Month, B.Title

**ORDER BY** #checkouts **DESC**

**LIMIT** 3;

**V1A**: **DROP** **VIEW** ISSUE1;

//**Frequent Users Report**

**V1: CREATE VIEW** ISSUE1

**AS SELECT** I.ISBN, I.IssueDate, I.Username, **MONTH**(I.IssueDate) **AS** Month

**FROM** ISSUE **AS** I;

**QV1: SELECT** U. Month, U.Username, COUNT(\*) **AS** #checkouts

**FROM** ISSUE1 **AS** U

**WHERE** U.Month = 1 **OR** U.Month = 2

**GROUP BY** U.Month, U.Username

**HAVING** COUNT(\*) > 10

**ORDER BY** #checkouts **DESC**

**LIMIT** 5;

**V1A**: **DROP** **VIEW** ISSUE1;

//**Popular Subject Report**

**V1: CREATE VIEW** ISSUE1

**AS SELECT** I.ISBN, I.IssueDate, **MONTH**(I.IssueDate) **AS** Month

**FROM** ISSUE **AS** I;

**QV1: SELECT** U. Month, B.SubName **AS** Top\_Subject, COUNT(\*) **AS** #checkouts

**FROM** ISSUE1 **AS** U **INNER JOIN** BOOK **AS** B **ON** U.ISBN = B.ISBN

**WHERE** U.Month = 1 **OR** U.Month = 2

**GROUP BY** U.Month, Top\_Subject

**ORDER BY** #checkouts **DESC**

**LIMIT** 3;

**V1A**: **DROP** **VIEW** ISSUE1;