Errors	case accure
Cannot find the object "flight_led" because it does not exist or you do not have permissions.	try to add foreign key in flight_leg table by usling alter
Foreign key 'seat' references invalid column 'airport_code' in referencing table 'seat'.	try to add foreign key in saet table by usling alter
Foreign key 'leg_instance' references invalid column 'leg_no' in referencing table 'leg_instance'.	try to add foreign key in saet table by usling alter , in this case both of table are week entity
Column names in each table must be unique. Column name 'date' in table 'flight_leg' is specified more than once.	add the date column to the flight_leg table
There are no primary or candidate keys in the referenced table 'flight_leg' that match the referencing column list in the foreign key 'FK__leg_Instance__6754599E'.	trying to create a foreign key from leg_Instance to flight_leg, but the column(s) you're referencing in flight_leg are not part of a PRIMARY KEY or UNIQUE constraint, which is required.

Could not drop object 'leg_Instance' because it is referenced by a FOREIGN KEY constraint.	delete table already other table connect with it using foreign key
ALTER TABLE ALTER COLUMN failed because column 'Mname' does not exist in table 'Employee'.	try to change the data type of column
Column 'Departments.Dname' is invalid in the select list because it is not contained in either an aggregate function or the GROUP BY clause.	The Departments.Dname column is used in a SELECT statement, but it is not in a GROUP BY clause and is not wrapped with an aggregate function (such as MAX, MIN, AVG, etc.).
Add nue column to exist table and inseart different value to this column based on the number of exist rows, notes that all rows get same last value	use "update" function to update the inseart the value s to the new column with out where condition
An expression of non-boolean type specified in a context where a condition is expected, near ')'.	use aggregate function with having condition to get Most Watched Movie [select sum(WH. WatchDuration) as Most_Watched_Movie , Movies.Title from WatchHistory WH join Movies on Movies.MovieID = WH.MovieID group by Movies.Title having max(WH.WatchDuration)]
Column name or number of supplied values does not match table definition.	values that add to the table ad new data do not match with the number of columns exist in the table , which has constrains that should be not null
String or binary data would be truncated in table 'hotel.dbo.branches', column 'name'. Truncated value: 'SalalahSta'.	inseart values to table

Arithmetic overflow error converting numeric to data type numeric.	try to inseart data to the table which has column with decimal data type and put the decimal range is (4,2) and inseart values of salary with more than 4 bit
	Stored proc
Msg 208, Level 16, State 3, Line 17 Invalid object name 'UpdateBookStatus'.	try to show data of UpdateBookStatus procedure
Msg 102, Level 15, State 1, Procedure sp_UpdateBookStatus, Line 4 [Batch Start Line 164] Incorrect syntax near '@NewStatus'. Msg 137, Level 15, State 2, Procedure sp_UpdateBookStatus, Line 9 [Batch Start Line 164] Must declare the scalar variable "@NewStatus".	try to create stored procedure
Invalid column name 'A123'.	Incorrect Data Type Assignment
	Trigge
Msg 102, Level 15, State 1, Procedure trg_PreventBookDeletion, Line 21 [Batch Start Line 3] Incorrect syntax near 'BEGIN'.	Can not delete book dose not exist in the table
BEFORE' is not a recognized trigger.	use before update trigger , while SQL Server, which does not support BEFORE triggers
Cannot create INSTEAD OF DELETE or INSTEAD OF UPDATE TRIGGER 'trg_BeforeUpdateTrigger' on table 'books'. This is because the table has a FOREIGN KEY with cascading DELETE or UPDATE.	"I want to create a BEFORE UPDATE trigger on the parent table to ensure that any updates to it are automatically reflected in the corresponding child tables. However, an error occurs due to the existing foreign key constraint on the child tables, which includes the ON UPDATE CASCADE option.
	Library Management Syste

Column 'books.title' is invalid in the select list because it is not contained in either an aggregate function or the GROUP BY clause.	try to select each book title with its average rating using join
An explicit value for the identity column in table 'member' can only be specified when a column list is used and IDENTITY_INSERT is ON.	add new recored to the table with put the value of primary key in that table
not problem: how extract the name of constrain from sql server	try to delete specific constrain to be able to delete the parent table
Cannot create more than one clustered index on table 'libraries'. Drop the existing clustered index 'PK__librarie__7A2F73CA3F4B18AD' before creating another.	create cluster index in the library table
Create View or Function failed because no column name was specified for column 2.	create veiw with aggregation function
Msg 156, Level 15, State 1, Procedure GetNextAvailableBook, Line 2 [Batch Start Line 22] Incorrect syntax near the keyword 'return'. Msg 102, Level 15, State 1, Procedure GetNextAvailableBook, Line 6 [Batch Start Line 22] Incorrect syntax near '@NextAvailableBook'. Msg 178, Level 15, State 1, Procedure GetNextAvailableBook, Line 11 [Batch Start Line 22] A RETURN statement with a return value cannot be used in this context.	create function return the next available book from book table : casues of error : RETURN in SQL Server is not used this way with RETURN INT if you want to return a value from a query. SELECT TOP @NextAvailableBook = 1 is syntactically incorrect. TOP does not work this way. RETURN ISNULL(..., 'No book available') is false because the function returns an INT, not a VARCHAR. The correct word is RETURNS, not return.

A SELECT statement that assigns a value to a variable must not be combined with data-retrieval operations.
Msg 156, Level 15, State 1, Procedure CalculateLibraryOccupancyRate, Line 13 [Batch Start Line 72]
Incorrect syntax near the keyword 'group'.

```
Operand data type date is invalid for subtract operator. " -- ========================
fn_GetLateReturnDays  ==================================
-- Return the number of late days for a loan (0 if not late)
select * from loan

create function fn_GetLateReturnDays ( @book_id int)
returns int
as
begin

declare @RemaineDays int;
declare @today date;
select @RemaineDays = ( return_date -  @today)
from loan
where book_id = @book_id;
return @RemaineDays;
end;"
```

Problems in Your Original Code:
Invalid SELECT with GROUP BY when assigning @borrowed_books.

You tried to assign a scalar (@borrowed_books) from a grouped result — that doesn’t work.

Ambiguous logic: if avail_status = 1 is meant to indicate borrowed, you don’t need to join with loan table to count them.

Table name inconsistency: libraries vs library.

@today is declared but not assigned a value.

return_date - @today syntax isn't valid SQL for date difference — should use DATEDIFF.

No handling if the book is returned early or on time.

note: if the purpose of function is create table the we have put return table inseate of return int "create function dbo.fn_ListAvailableBooksByLibrary(@libr

```
RETURNS TABLE
as
RETURN
(

    select *
    from books
    where library_id = @library_id  and avail_status = 0


);"
```

when we whant to test function return value use this syntax: exec <funtion_name> , while if we want to test function return table use this syntax : select

CREATE FUNCTION failed because a column name is not specified for column 1. in this code "create

```
function fn_GetTopRatedBooks(@library_id int)
returns table
as
return
(
select avg(rating) , books.title
from books join MemberBookReviews MBR
on books.book_id = MBR.book_id
JOIN reviews rev
on rev.review_number = MBR.review_number
where books.library_id = @library_id
group by  books.title
HAVING avg(rating) >= 4.56

);"
```

try create function by input specidic id value

```
Msg 102, Level 15, State 31, Procedure fn_FormatMemberName, Line 4 [Batch Start Line 189]
Incorrect syntax near 'RETURN'.
Msg 102, Level 15, State 1, Procedure fn_FormatMemberName, Line 13 [Batch Start Line 189]
Incorrect syntax near ')'.
```

in this code "-- =========================== fn_FormatMemberName =====================

```
-- Returns the full name formatted as "LastName, FirstName"
create function fn_FormatMemberName (@member_id int)
returns varchar(20)
as
return
(
declare @FullName varchar(20);

select @FullName = (mem.F_name + ' ' + mem.L_name)
from member mem
where mem.member_id = @member_id;

return @FullName;
);"
```

descripe of project "   Problems in your original code:

You're using DECLARE and SELECT inside a RETURN (...) block — that's not allowed.

Scalar functions in SQL Server require a proper BEGIN ... END block.

You must RETURN a value after assigning it."

```
Subquery returned more than 1 value. This is not permitted when the subquery follows =, !=, <, <= , >, >=
or when the subquery is used as an expression.
```

create function return full name "create

```
function fn_FormatMemberName
(@member_id int)
returns varchar(20)
as
return
(
declare @FullName varchar(20);

select @FullName = (mem.F_name + ' ' + mem.
L_name)
from member mem
where mem.member_id = @member_id;

return @FullName;
);"
```

try make procedure with use if condection with logic operator "Or" to chech the date and update the status,   Why did this error occur?

Because the book (@book_id) may have been checked out more than once, so:

The subquery returns multiple dates (return_date).

But you're using it in a condition with >=, which only accepts one value.

An aggregate may not appear in the set list of an UPDATE statement.

try to update column on table with values get after use aggregation function , update libraries

```
        set libraries.revenue = sum(p.amount)  "
CREATE TRIGGER trg_CalculateLibraryRevenue
on Payments
AFTER INSERT
as
begin
update libraries
set libraries.revenue = sum(p.amount)
from libraries join books
on libraries.library_id = books.library_id
join loan
on books.book_id = loan.book_id
join loan_pay lp
on
loan.loan_id = lp.loan_id and
loan.book_id = lp.book_id and
loan.member_id = lp.member_id
join Payments p
on p.pay_id = lp.pay_id
end;"
```

solve	code (if it is possible)
make sure for the table name	
should add "airport_code" column to seat table	
make sure for the name of column "leg_no" is exist in the flight_leg table"	
change the name of date column as FlightLeg_Date	EXEC sp_rename 'leg_instance.date', 'date0', 'COLUMN';
make soure thate attributes og foreign key has same attributes of primay key of references table	

just rename the name of table no need to delet the column or table	
just make sure of the name of column	
Make sure put the column is not in aggregation funcation in the group by function	select count(*) as NumEmployee, emp.Dno, dep.Dname from Employee emp left join Departments dep ON emp.Dno = dep.Dnum group by emp.Dno , dep.Dname
add where condition to the update script the inseart value to specific id in the datable	UPDATE seat SET Customer_name = 'Fatema' where seat_no = 1;
use top 1 with with group by and order by and no need to use having condition	select Top 1 sum(WH.WatchDuration) as Most_Watched_Movie , Movies.Title from WatchHistory WH join Movies on Movies.MovieID = WH.MovieID group by Movies.Title
make sure new data has values for add columns, or change the constrians of unimportant column	
change the lenght of varvhar for that columm	ALTER TABLE branches ALTER COLUMN name varchar(50);

change the constrains of salary column with different of number of bit of values, which i change from (4,2) to (8,4)	alter table staff alter column salary decimal (8,4)
edured	

Stored Procedure, it is not possible to use SELECT * FROM on stored procedures.	SELECT * FROM Book;
	EXEC UpdateBookStatus @BookID = 1, @NewStatus = 'Issued';
	SELECT * FROM Book

Incorrect table name 'Books' instead of 'Book', so check the name of table . and also check to put semicolon at the end of the UPDATE statement	
make sure of datatype	
er	
, we should make sure the id of book exist	INSTEAD OF INSERT
must use use an INSTEAD OF UPDATE trigger. in sql to achieve the desired behavior of a "before update" trigger in SQL Server	CREATE TRIGGER trg_BeforeUpdateBookPrice ON books INSTEAD OF UPDATE AS .....
if still want to do this case with trigger, we must drop the ON UPDATE CASCADE constrain of foriegn key in child tables which related with the parent table .	* ALTER TABLE ChildTableName DROP CONSTRAINT FK_ConstraintName; * ALTER TABLE ChildTableName ADD CONSTRAINT FK_ConstraintName FOREIGN KEY (book_id) REFERENCES book (book_id)
m – DB Project Part 2	

use group by function for title column

do not manually insert values into the primary column. This column has been configured with an IDENTITY_INSERT constraint, meaning its values are automatically generated by the system. Manual entry will lead to errors.

```
SELECT
  table_name,
  constraint_type,
  constraint_name
FROM information_schema.table_constraints
WHERE table_name = 'loan_pay';
```

remove the exist cluster and create new one. Or do not create new clustore becouse the exist clustor is primary key which other tables dependen on it

name column

count(loan.member_id) as loan_count

We used TOP 1 correctly.

The function returns book_id or -1 if no book was found.

You can change -1 to any value you like, such as "not found."

Assumptions:
A book is considered borrowed if avail_status = 1.

Table libraries contains total_capacity.

Uses GETDATE() to get today's date.

Uses DATEDIFF to calculate the number of late days.

Uses TOP 1 ... ORDER BY loan_date DESC to ensure only the latest loan for the book is used.

ary_id int)

* from <function table>

put aggregation function with name like this "select avg (rev.rating) AS average_rating, books.title"

```
create function dbo.fn_FormatMemberName (@member_id int)
returns varchar(20)
as

begin
	declare @FullName varchar(20);

	select @FullName = (mem.F_name + ' ' + mem.L_name)
	from member mem
	where mem.member_id = @member_id;

	return @FullName;
end;
```

use begin and ends inseated of Parentheses () block

We only need to handle the most recent loan (latest loan_date for this book_id) using TOP 1 and ORDER BY loan_date DESC.   ** We used TOP 1 ... ORDER BY loan_date DESC to ensure only the most recent loan is retrieved.

We bound UPDATE to the same book_id and return_date to ensure only one record is updated.

It's better to use loan_id directly instead of return_date to precisely identify the record.

```
CREATE TRIGGER trg_CalculateLibraryRevenue
on Payments
AFTER INSERT
as
begin
UPDATE lib
   SET revenue = rev.total_amount
   FROM libraries lib
   JOIN (
      SELECT
         b.library_id,
         SUM(p.amount) AS total_amount
      FROM books b
      JOIN loan l ON b.book_id = l.book_id
      JOIN loan_pay lp ON
         l.loan_id = lp.loan_id AND
         l.book_id = lp.book_id AND
         l.member_id = lp.member_id
      JOIN Payments p ON p.pay_id = lp.pay_id
      GROUP BY b.library_id
   ) AS rev
   ON lib.library_id = rev.library_id;
END;
```

The UPDATE statement should be rewritten to update each library (library_id) with only its own fines sum value, using FROM and GROUP BY in the subquery and linking them appropriately.