*Lazar Bianca-Maria, group 924*

**Yes, to err is human** ~~(so don’t be one)~~

* *My 24 day journey with SQL errors –*
* “There are fewer columns in the INSERT statement than values specified in the VALUES clause. The number of values in the VALUES clause must match the number of columns specified in the INSERT statement.”

SolarSystems[ **SSID**, SolarSystemName, GalaxyID ]

INSERT INTO SolarSystems(SSName,GID) VALUES (**100**,'Solar System',1000)

*(We have too many values on the right) (Either put SSID on the left, or, if IDENTITY was used, do not specify the id in VALUES)*

*Similarly, if we put*

**PlanetID** INT PRIMARY KEY ***IDENTITY***(100,1)

*And then try the Insert:*

INSERT INTO Planets(**PlanetID**,PName,PDensity,PPopulation,SSID) VALUES (**2**,'Mars',5243,0,104)

* “Cannot insert explicit value for identity column in table 'Planets' when IDENTITY\_INSERT is set to OFF”

*(We can’t specify the ID explicitly if we have the ID set to* ***IDENTITY****)(Do not specify the ID)*

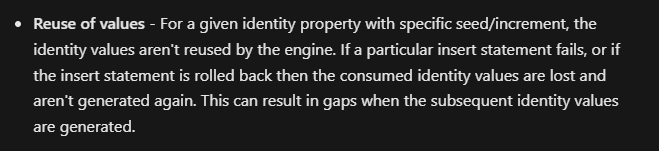
* IMPORTANT WHEN USING IDENTITY:

*PRIMARY KEY IDENTITY(100,1)*

*DOES* ***NOT*** *START OFF WITH SEED AT 100. EVERY EXECUTE, IT WILL GIVE THE NEXT x NUMBERS, STARTING FROM THE PREVIOUS LAST GENERATED ID.*

“***seed*** *=*The value that is used for the very first row loaded into the table.

***Increment*** *=* The incremental value that is added to the identity value of the previous row that was loaded.”



*(src :* [*https://learn.microsoft.com/en-us/sql/t-sql/statements/create-table-transact-sql-identity-property?view=sql-server-ver16*](https://learn.microsoft.com/en-us/sql/t-sql/statements/create-table-transact-sql-identity-property?view=sql-server-ver16) *)*

*FIX:*

*DBCC CHECKIDENT ('Galaxies', RESEED, 999);*

*-- OBS! for the first run, we start from 999; for the second execute, we start with 1000*

* “Column 'Researchers.ResearcherID' is invalid in the select list because it is not contained in either an aggregate function or the GROUP BY clause”

SELECT R.ResearcherID, R.RName, MAX(R.RDoB)

FROM Researchers R INNER JOIN Observe Obs ON R.ResearcherID = Obs.RID

INNER JOIN Planets P ON OBS.PID = P.PlanetID

GROUP BY OBS.PID

* “Only one expression can be specified in the select list when the subquery is not introduced with EXISTS.”

-- (find youngest researcher per observed planet)

SELECT R.ResearcherID, R.RName

FROM Researchers R

WHERE R.RDoB = (

SELECT O.PID, MAX(R.RDoB)

FROM Researchers R INNER JOIN Observe O ON R.ResearcherID = O.RID

INNER JOIN Planets P ON O.PID = P.PlanetID

GROUP BY O.PID

)

* A STUDY IN OVERCOMPLICATING THINGS:

*The query:*

SELECT O.RID, t.PID

FROM Observe O INNER JOIN(

SELECT O.PID

FROM Researchers R INNER JOIN Observe O ON R.ResearcherID=O.RID

INNER JOIN Planets P ON O.PID=P.PlanetID

GROUP BY O.PID) t

ON O.PID = t.PID

*Is equivalent to:*

SELECT O.RID, O.PID

FROM Observe O

* “Operand data type date is invalid for subtract operator.”

SELECT \*, **Age=CAST( GETDATE() AS Date )-R.RDoB**

FROM Researchers R

*FIX:*

SELECT \*, **DATEDIFF(YEAR, R.RDoB, GETDATE()) AS Age**

FROM Researchers R

* “All queries combined using a UNION, INTERSECT or EXCEPT operator must have an equal number of expressions in their target lists.”

SELECT **R.ResearcherID, R.RName**

FROM Researchers R

WHERE R.RDepartment='chemistry'

UNION

SELECT **O.RID**

FROM Observe O

WHERE O.Obs = 'inconclusive'

* “Invalid object name 'Researchers' “ *(because of table not connected to DB)*
* “Conversion failed when converting the varchar value 'r1' to data type int.”

SELECT **R.RName**

FROM Researchers R

WHERE R.RDepartment='chemistry'

UNION

SELECT **O.RID**

FROM Observe O

WHERE O.Obs = 'inconclusive'

*( INT and VARCHAR are incompatible data types! )*

* *WHEN RUNNING JUST THE SUBQUERY:* “The multi-part identifier "S.PID" could not be bound.”

SELECT \*

FROM Satellites S

WHERE EXISTS (

SELECT \*FROM Planets P

WHERE P.PPopulation>0 AND S.PID=P.PlanetID )

*///// ATTEMPTING TO: Return the data of the planet with the biggest density*

* Try 1: “Column 'Planets.PlanetID' is invalid in the select list because it is not contained in either an aggregate function or the GROUP BY clause”

SELECT \*,MAX(P.PDensity)

FROM Planets P

* *FIX: using subqueries*

SELECT \*

FROM Planets P

WHERE P.PDensity = ANY(

SELECT MAX(P2.PDensity)

FROM Planets P2)

* **IMPORTANT WHEN USING USER SPECIFIED PROCEDURES:**

*When running the main procedure, before executing the code that creates the procedure it calls:*

* “The module 'uspProcMain' depends on the missing object 'uspCreateTableAsteroids'. The module will still be created; however, it cannot run successfully until the object exists.”
* “Procedure or function 'uspSelectVersion' expects parameter '@NewVersion', which was not supplied.”

EXEC uspSelectVersion @Version=1

*(The parameter we supplied has a different name than the parameter from the procedure)*

* Must declare the scalar variable "@NrPlanets"

DECLARE @LocalVar INT

EXEC uspGetPirates **@LocalVar=@NrPlanets**

PRINT @LocalVar

(*The order of the parameters should be @NrPlanets=@LocalVar)*

* Must declare the scalar variable "@LocalVar".

DECLARE @LocalVar INT

GO

EXEC uspGetPirates @NrPlanets=@LocalVar

PRINT @LocalVar

*(The “GO” batch separator is not placed correctly)*