Video 5.4

The template below will help you structure your video using the PowerPoint slides and content flow given. You need to use this as a reference to create your video. Please **DO NOT FILL** the section below, it is for your reference only.

|  |
| --- |
| Plan your narration using the template below |

|  |  |
| --- | --- |
| **Introduction** | |
| **Action on Screen (Code /imagery to support the narration)** | **Narration (The corresponding explanation to the Action on Screen)** |
| [Mandatory Slide] This is the introduction slide of the section where the name of the section would be displayed. | *Hi and welcome to Microsoft Access – Building Meaningful Relationships (RDBMS)*  *section 5.4*  *Understanding the Relationship* |
| [ | *In this video, I’ll be covering the following:*   1. *Where to set a Table or Query Join* 2. *Understand the scope of a Table Join* 3. *Showing you how to Join Tables in a Query* 4. *Creating the ‘Find Unmatched’ Query using the buit in Wizard and* 5. *Explaining what the ‘Is Null’ expression is.*   *Let’s get started…* |
|  | *Setting Access Joins can be set in one two ways…*   * *The Query Window or the Relationship Window and have different uses and reasons…* * *Both areas can handle the setting of tables and queries but…* * *In a Query Window, it’s said to be a local join compared to…* * *The Relationship Window being a global one. The scope is different between the two.* * *In a Query Window, a join are temporary and disconnects when the query closes whereas…* * *In a Relationship Window, it’s always connected until you actually break the link yourself.* * *The Query Window join is basic and sufficient for reports but…* * *The Relationship Window has more functionality to ensure data integrity is maintained.* * *The Query Window join is better suited for general reporting in your database though it can also be…* * *In the Relationship Window too but here it’s also for more the data inputting aspects to the database and better controls data validations and their joins between tables.* * *So, it’s down to the scope…LOCAL versus…* * *GOBAL. But… be clear…* * *A LOCAL will supersede the GLOBAL while the query is running if you have the same join between tables in both area and will revert to the GLOBAL setting when the query closes.*   *[Back in Access…]* |
|  | *Let’s go and set up a join in the Query Window…*  *In a new query, I add the tables I wish to include in a relationship based on my report requirements.*  *For example, if I want to report on all my UK Customers and their Orders for the year 2015 and sort them by the Order ID, I will need the following tables…*  *In a new Query, (CREATE tab and click on the Query Design icon)…*  *I need to add three tables from the Show Table dialog box that appears…*  *I select the Countries, Customers and orders tables using the CTRL key held down as single clicking each item…clicking the Add button to append them to the query design…*  *And then click the Close button for the dialog box…*  *Next, quickly re-arranging and resizing the three tables so it becomes a little easier to work with…*  *I can now identify and join the first two tables…using the CountryID field in Countries linked to the Country field in Customers…which are both indexed (as a primary and a secondary key respectively)…*  *Selecting the field in the Countries table and then holding down the mouse, I drag it over the corresponding field in the Customers table like so…and release the mouse button.*  *You can now see a join… it’s an inner join meaning only show me records where there’s a match between the two tables.*  *If you make a mistake and connect the wrong field, it can be deleted by first selecting anywhere on the join and either press the DELETE key or right mouse click and choose delete…*  *Next, I will connect the Customers to the Orders table by identifier the common field between the two being CustomerID…*  *Moving any table around will keep the join in tact…* |
|  | *If I now add the following fields:*  *Customer ID, Company Name, Country, Order ID, Order Date and Order Amount…*  *I quickly run the query and check my results…*  *Where I can see there are 1078 records – this is important to note as it can be a check to see if the joins are working as they should – knowing there are only 1078 orders from the orders table.*  *If this number were different, it would need to be confirmed that it was correct and intentional…*  *Both joins are inner joins meaning only reveal a matching record across the three tables and suppresses the rest…so in effect, a filter has been applied even before we add criteria.*  *Back in design, I now add the criteria…*  *UK for the Country field…run the table and confirm the record count again…303.*  *Add the date range to the order date field as ‘between 1/1/15 and 31/12/15 (apply the sorting too) and run again…confirm the record count being…99.* |
|  | *Back in design…Let’s see the impact if I delete a join now…*  *First, I’m going to add another sorting to the Customer ID field and quickly run the query again…*  *As you can see the first three records are for the same customer ‘Around The Horn’ who has placed three different orders…back in design.*  *If I now delete a join between the Countries and Customers table, let’s see the impact here…*  *The number of records has increased from 99 to 328 – but why?*  *Take a look at the recordset carefully and note the three orders for the customer ‘Around The Horn’ has moved down and appears after five other records suddenly appearing…*  *In fact, the two other customer ID’s are not actually UK based customers but have appeared in the query and seem to be so…but this is incorrect!*  *If I quickly open the Customers table and take a look, you can see there are USA based customers…*  *In effect, having a stand-alone table in a query causes a compounding effect and multiplies the records set out ignoring a country match altogether – which is incorrect.*  *This is why you need to keep an eye on the record count in a query and check your results elsewhere.*  *It also, confirms that all tables 9and queries if used) must be joined somehow in a query to avoid this cartesian compounding effect it will cause.*  *If I remove the remaining join again, it will now compound across all three tables or simple multiply each record by each record and apply a criteria as set … causing this to be an irrelevant report…*  *Look at the number of records now? Nearly 30,000!* |
|  | *Joining back the three tables, if I try and connect a mismatch field…*  *For example, I join the ‘Country’ field instead of CountryID to Country in the Customers table and connect the Customers to orders as before…*  *When I attempt to run the query now…*  *It will show an error and a type mismatch message will appear.*  *This supports the rules about joining fields and must be of the same data type.* |
|  | *Let’s now take a look at creating a different type of query join…*  *I’m going to use the ‘Query Wizard’ feature and create what is known as a ‘Find Unmatched Query’…*  *Under the CREATE tab, click the ‘Query Wizard’ icon…*  *There are a few query options to choose, I’m interested in the last option…*  *I want to create an unmatched query between customers and orders and if we have any customers yet to place their first order…*  *I choose customers being the table to interrogate…*  *I match it to the related table – its orders (as prompted in the caption)…*  *I then confirm the common field to join…Customer ID…and click the join button to establish it…*  *Pick and chose the fields to view…Customer ID and Company Name will do…*  *Give it a name (or leave as it) and click the Finish button to create the query…*  *We now can see just the one record – PARIS which suggest this customer has yet to place their first order…*  *Let’s look at the design view and see the link…*  *You can a see a left join here meaning only show me all customers and related orders…*  *However, it has also suppressed its matching orders too using the ‘Is Null’ expression in the criteria for the Customer ID field (of the orders table) in the lower half of the query grid…*  *You can see the left join settings by right mouse clicking on the join and choose Properties…*  *Just remember, a join in a query will disconnect when the query is closed.* |
|  | *Finally, The Is Null expression needs a little clarifying when used in Microsoft Access…as it’s not an empty value!*   * *‘Null’ really means ‘No Data’ (and hasn’t been specified)* * *The ‘Null’ is not the same as ‘Empty’* * *The ‘Null’ is not the same as Zero* * *Can be used in a Table (Required property)* * *Can be used in a Query/SQL (as criteria)* * *Can be used in Macros and VBA code* * *Date/Time Data Types are ‘Null’ until it has a value* * *‘IsNull’ is also a Function which means it returns either a TRUE or FALSE value* |
|  |  |
|  |  |
|  |  |

|  |  |  |
| --- | --- | --- |
| **Steps or Tasks** | | |
| **Action on Screen (Code /imagery to support the narration)** | | **Narration (The corresponding explanation to the Action on Screen)** |
| **Summary** | | |
|  | *To summarise:*   1. *You know where to set a Table or Query* 2. *Understand the scope of a Table Join* 3. *Shown you how to Join Tables in a Query* 4. *Created the ‘Find Unmatched’ Query using the built in Wizard and* 5. *Explained what the ‘Is Null’ expression is all about.* | |
| [Mandatory slide] – Next Video | *In the next video, I will taking a closer look the Access Relationship Window and cover its features.* | |