**WORKING WITH VIEWS**

Views are select statements thats result is stored in your database. Let's create a view that contains our main purchase order info.

CREATE VIEW purchase\_order\_overview AS

SELECT sales\_order.purchase\_order\_number, customer.company,

sales\_item.quantity, product.supplier, product.name, item.price,

--Can’t use total if you want this to be updated Fix Below

(sales\_item.quantity \* item.price) AS Total,

--Remove concat if you want this to be updatable

CONCAT(sales\_person.first\_name, ' ', sales\_person.last\_name) AS Salesperson

FROM sales\_order -- Join some tables

JOIN sales\_item

ON sales\_item.sales\_order\_id = sales\_order.id -- Tables go together by joining on sales order id

-- Any time you join tables you need to find foreign and primary keys that match up

JOIN item

ON item.id = sales\_item.item\_id -- Join item as well using matching item id

JOIN customer

ON sales\_order.cust\_id = customer.id // Join customer using customer ids

JOIN product

ON product.id = item.product\_id

JOIN sales\_person

ON sales\_person.id = sales\_order.sales\_person\_id

ORDER BY purchase\_order\_number;

When data in the database is updated so is the view. You can use the view in all the same ways you can a regular table. If you want it to be updatable though it can’t include DISTINCT, UNION, Aggregate Functions, GROUP BY or HAVING.

SELECT \* FROM purchase\_order\_overview;

**Recalculate Total**

If we removed total above so it could be updated we can just calculate with total like this

SELECT \*, (quantity \* price) AS Total

FROM purchase\_order\_overview;

**Drop a View**

DROP VIEW purchase\_order\_overview;