

# Code Girl T-SQL Problem Set 2- Solution

Most of the time you not need all of the rows returned from a table. You will frequently need to filter the rows with a WHERE clause. The WHERE clause contains one or more expressions that compares two things. When the comparison returns TRUE, the row is returned.

Before working on this problem set, be sure to watch these videos on the Aunt Kathi Coder Girl page:

[Filter your results](#)

[WHERE Clause deep dive](#)

## Exercise 1

1.1 Write a query that returns the SalesOrderID, CustomerID, TotalDue, and OrderDate from the Sales.SalesOrderHeader table. Only return the rows that are for CustomerID 29825.

---

```
SELECT SalesOrderID, CustomerID, TotalDue,
       OrderDate
FROM Sales.SalesOrderHeader
WHERE CustomerID = 29825;
```

---

1.2 Change the query from 1.1 so that it returns the orders place on 2013-10-04 for any customers.

---

```
SELECT SalesOrderID, CustomerID, TotalDue,
       OrderDate
FROM Sales.SalesOrderHeader
WHERE OrderDate = '2013-10-04';

--Actually better. Required if times were filled in
SELECT SalesOrderID, CustomerID, TotalDue,
       OrderDate
FROM Sales.SalesOrderHeader
WHERE OrderDate >= '2013-10-04' AND OrderDate < '2013-10-05';
```

---

1.3 Change the query from 1.1 so that it returns the orders where the TotalDue is equal to \$40.432.

---

```
SELECT SalesOrderID, CustomerID, TotalDue,
       OrderDate
FROM Sales.SalesOrderHeader
```

```
WHERE TotalDue = 40.432;
```

---

1.4 Write a query that returns the BusinessEntityID, FirstName, and LastName from the Person.Person table. Only return the rows that have a last name of Cook.

---

```
SELECT BusinessEntityID,  
       FirstName, LastName  
FROM Person.Person  
WHERE LastName = 'Cook';
```

---

1.5 Try finding the table that has a list of departments. Return the DepartmentID and Name of all departments in the Sales and Marketing group.

---

```
SELECT DepartmentID, Name  
FROM HumanResources.Department  
WHERE GroupName = 'Sales and Marketing';
```

---

## Exercise 2

2.1 Write a query that returns the SalesOrderID, CustomerID, TotalDue, and OrderDate from the Sales.SalesOrderHeader table. Only return the rows for all customers except for customer 29825.

---

```
SELECT SalesOrderID, CustomerID, TotalDue,  
       OrderDate  
FROM Sales.SalesOrderHeader  
WHERE CustomerID <> 29825;
```

--This is also correct

```
SELECT SalesOrderID, CustomerID, TotalDue,  
       OrderDate  
FROM Sales.SalesOrderHeader  
WHERE CustomerID != 29825;
```

---

2.2 Write a query that returns the SalesOrderID, CustomerID, TotalDue, and OrderDate from the Sales.SalesOrderHeader table. Only return the rows that have a TotalDue over \$1000.

---

```
SELECT SalesOrderID, CustomerID, TotalDue,  
       OrderDate  
FROM Sales.SalesOrderHeader
```

WHERE TotalDue > 1000;

---

2.3 Write a query that returns a list of the products, including the name, ProductID, and color where the color is red, blue, or yellow.

---

```
SELECT ProductID, Name, Color
FROM Production.Product
WHERE Color IN ('Red','Blue','Yellow');
```

```
SELECT ProductID, Name, Color
FROM Production.Product
WHERE Color = 'Red'
       OR Color = 'Blue'
       OR Color = 'Yellow';
```

---

2.4 Write a query that returns the SalesOrderID, CustomerID, TotalDue, and OrderDate from the Sales.SalesOrderHeader for all sales in 2014 and later.

---

```
SELECT SalesOrderID, CustomerID, TotalDue,
       OrderDate
FROM Sales.SalesOrderHeader
WHERE OrderDate >= '2014-01-01';
```

---

2.5 Write a query that returns the SalesOrderID, CustomerID, TotalDue, and OrderDate from the Sales.SalesOrderHeader for the year 2012.

---

```
SELECT SalesOrderID, CustomerID, TotalDue,
       OrderDate
FROM Sales.SalesOrderHeader
WHERE OrderDate >= '2012-01-01'
       AND OrderDate < '2013-01-01';
```

---

2.6 Write a query that returns the ProductID, Name, Color, and ListPrice for all products with list prices between \$100 and \$300.

---

```
SELECT ProductID, Name, Color, ListPrice
FROM Production.Product
```

```
WHERE ListPrice BETWEEN 100 AND 300;
```

```
SELECT ProductID, Name, Color, ListPrice  
FROM Production.Product  
WHERE ListPrice >= 100 AND ListPrice <= 300;
```

---

### Exercise 3

3.1 Write a query that returns the ProductID, Name, Color, and ListPrice for all products with a color name that contains the letter B.

---

```
SELECT ProductID, Name, Color, ListPrice  
FROM Production.Product  
WHERE Color LIKE '%B%';
```

---

3.2 Write a query that returns the ProductID, Name, Color, and ListPrice for all products with a name containing bike or bicycle.

---

```
SELECT ProductID, Name, Color, ListPrice  
FROM Production.Product  
WHERE Name LIKE '%bike%'  
       OR NAME LIKE '%bicycle%';
```

---

3.3 Write a query that returns the ProductID, Name, Color, and ListPrice for all products with a name starting with Mountain or Road.

---

```
SELECT ProductID, Name, Color, ListPrice  
FROM Production.Product  
WHERE Name LIKE 'Mountain%'  
       OR NAME LIKE 'Road%';
```

---

3.4 Write a query that returns the list of names where the first name or last name starts with Z.

---

```
SELECT FirstName, LastName  
FROM Person.Person  
WHERE LastName LIKE 'Z%'  
       OR FirstName LIKE 'Z%';
```

---

3.5 Write a query that returns the list of name where the first name and last name does not start with Z.

---

```
SELECT FirstName, LastName
FROM Person.Person
WHERE NOT (LastName LIKE 'Z%'
          OR FirstName LIKE 'Z%');
```

```
SELECT FirstName, LastName
FROM Person.Person
WHERE LastName NOT LIKE 'Z%'
      AND FirstName NOT Like 'Z%';
```

---

3.6 Write a query that returns a list of the products that starts with the word mountain and has a list price less than \$20.

---

```
SELECT ProductID, Name, ListPrice
FROM Production.Product
WHERE Name LIKE 'Mountain%'
      AND ListPrice < 20;
```

---

3.7 Write a query that returns a list of the products that contain the word socks in the name. The products must also either be the color white or have a list price less than \$20.

---

```
SELECT ProductID, Name, Color, ListPrice
FROM Production.Product
WHERE Name LIKE '%Socks%'
      AND (Color='White' OR ListPrice < 20);
```

---

## Exercise 4

4.1 When adding a WHERE clause, remember to use tick marks or single quotes around literal strings and dates.

4.2 WHERE clauses contain expressions, called predicates, that can be evaluated to TRUE, FALSE or UNKNOWN.

4.3 When using a WHERE clause, you can use many operators, not just the equal sign.

