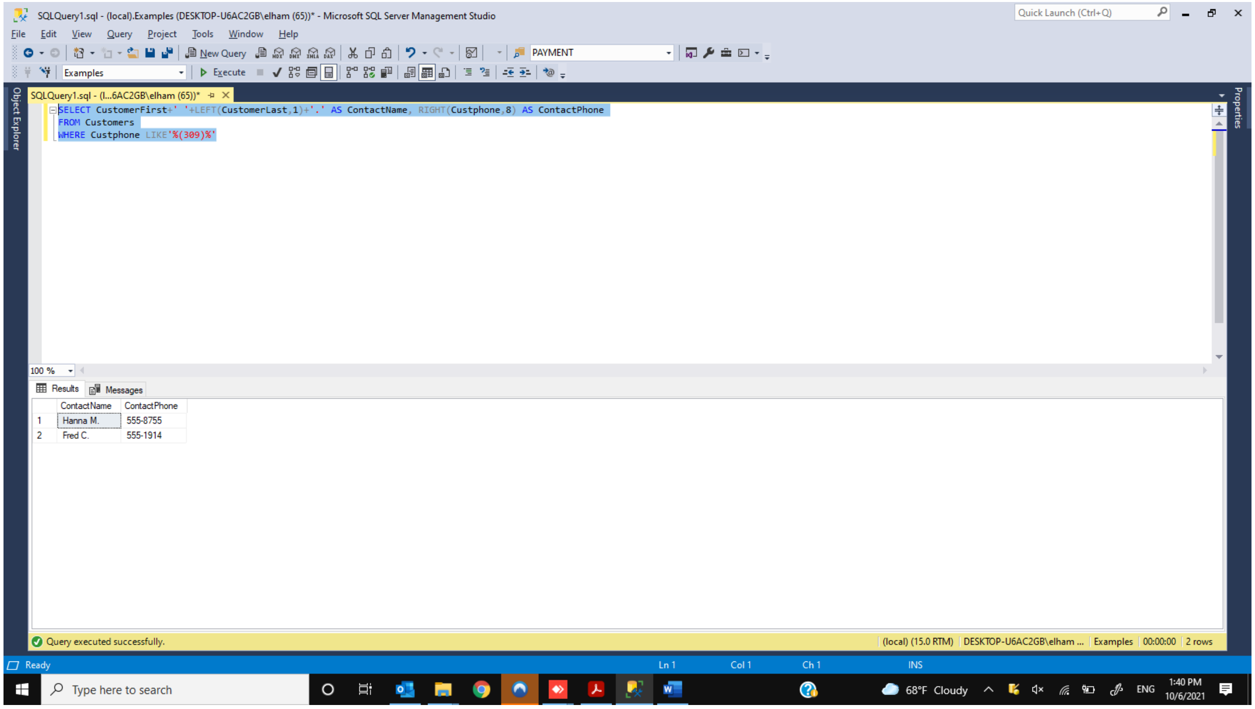
1. Write a SELECT statement that returns two columns based on the Customers table. The first column, ContactName, is the customer's name in this format: Customers first name (i.e. CustomerFirst column) followed by first letter of Customers last name (i.e. CustomerLast column) followed by a dot (for example, the format must look like, "Maria A."). The second column, ContactPhone, is the CustPhone column without the area code. Only return rows for those customers in the '309' area code. Sort the results set by Customers first name in ascending order. Use Examples database.

```sql
SELECT CustomerFirst+' '+LEFT(CustomerLast,1)+'.' AS ContactName, RIGHT(Custphone,8) AS
ContactPhone
FROM Customers
WHERE Custphone LIKE'%(309)%';
```


SQLQuery1.sql - (local).Examples (DESKTOP-U6AC2GB\elham (65))* - Microsoft SQL Server Management Studio
Quick Launch (Ctrl+Q)
File
Edit
View
Query
Project
Tools
Window
Help
New Query
PAYMENT
Examples
Execute
Object Explorer
Properties
SQLQuery1.sql - (l...6AC2GB\elham (65))*

```sql
SELECT CustomerFirst+' '+LEFT(CustomerLast,1)+'.' AS ContactName, RIGHT(Custphone,8) AS ContactPhone
FROM Customers
WHERE Custphone LIKE'%(309)%'
```

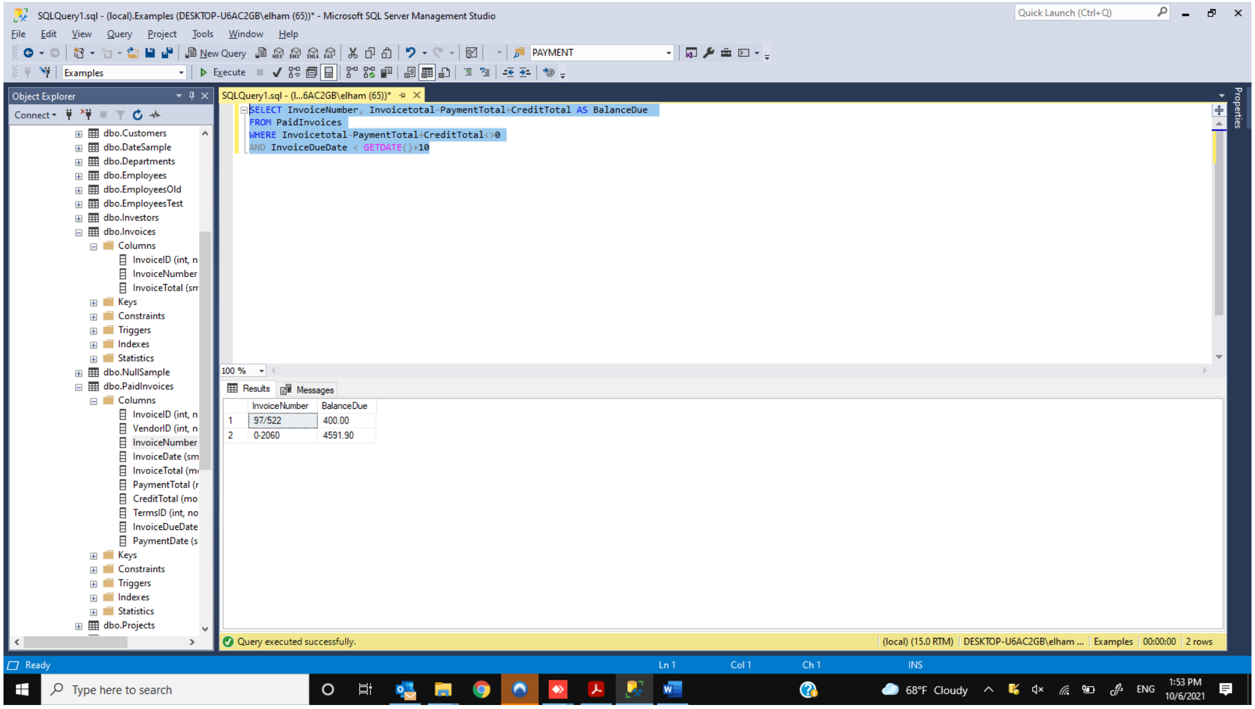
	ContactName	ContactPhone
1	Hanna M.	555-8755
2	Fred C.	555-1914

Query executed successfully.

(local) (15.0 RTM) | DESKTOP-U6AC2GB\elham ... | Examples | 00:00:00 | 2 rows

2. Write a SELECT statement that returns the InvoiceNumber and balance due for every invoice with a non-zero balance and an InvoiceDueDate that's less than 10 days from today (i.e. InvoiceDueDate < today's date + 10).

```sql
SELECT InvoiceNumber, Invoicetotal-PaymentTotal + CreditTotal AS BalanceDue
FROM PaidInvoices
WHERE Invoicetotal - PaymentTotal + CreditTotal <> 0
AND InvoiceDueDate < GETDATE() + 10;
```



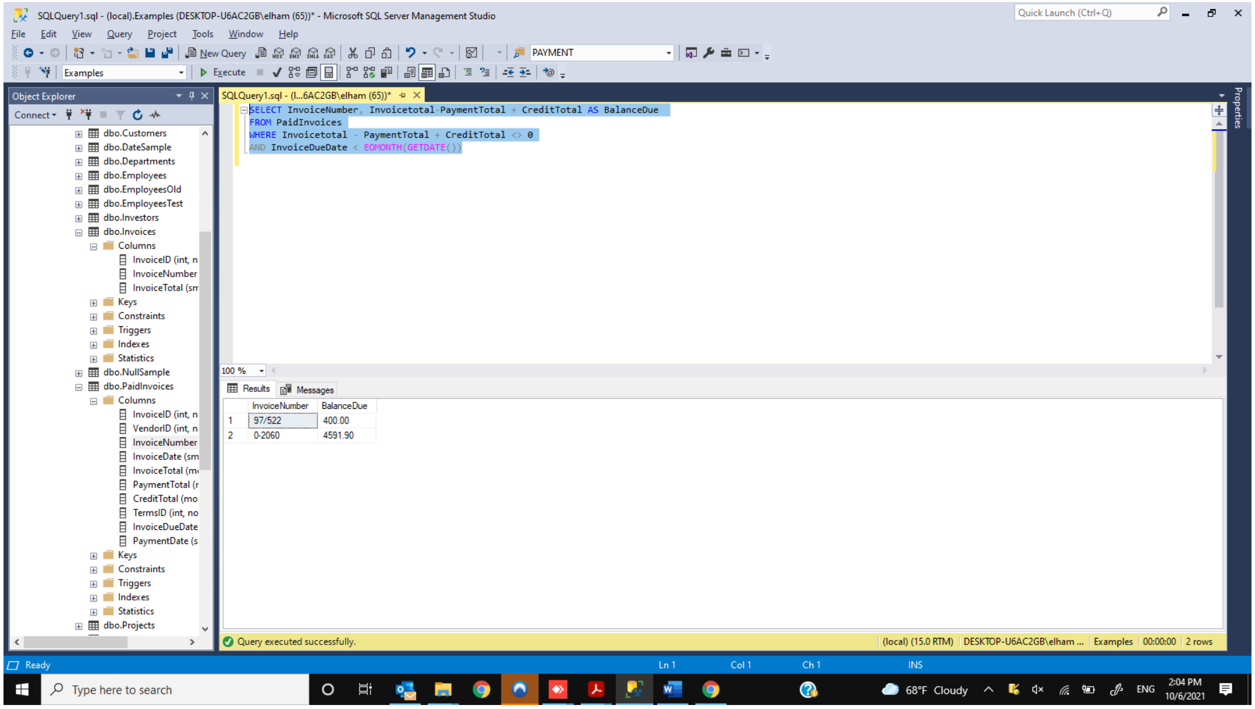
```sql
SELECT InvoiceNumber, Invoicetotal-PaymentTotal+CreditTotal AS BalanceDue
FROM PaidInvoices
WHERE Invoicetotal-PaymentTotal+CreditTotal<>0
AND InvoiceDueDate < GETDATE()+10
```

	InvoiceNumber	BalanceDue
1	97/522	400.00
2	0-2060	4591.90

Query executed successfully. (local) (15.0 RTM) DESKTOP-U6AC2GB\elham ... Examples 00:00:00 2 rows

3. Modify the search expression for InvoiceDueDate from the solution for question 2. Rather than 10 days from today, return invoices due before the last day of the current month.

```sql
SELECT InvoiceNumber, Invoicetotal-PaymentTotal + CreditTotal AS BalanceDue
FROM PaidInvoices
WHERE Invoicetotal - PaymentTotal + CreditTotal <> 0
AND InvoiceDueDate < EOMONTH(GETDATE());
```



SQLQuery1.sql - (local).Examples (DESKTOP-U6AC2GB\elham (65))* - Microsoft SQL Server Management Studio
File Edit View Query Project Tools Window Help
New Query
PAYMENT
Examples
Execute
Object Explorer
Connect
SQLQuery1.sql - (l...6AC2GB\elham (65))*

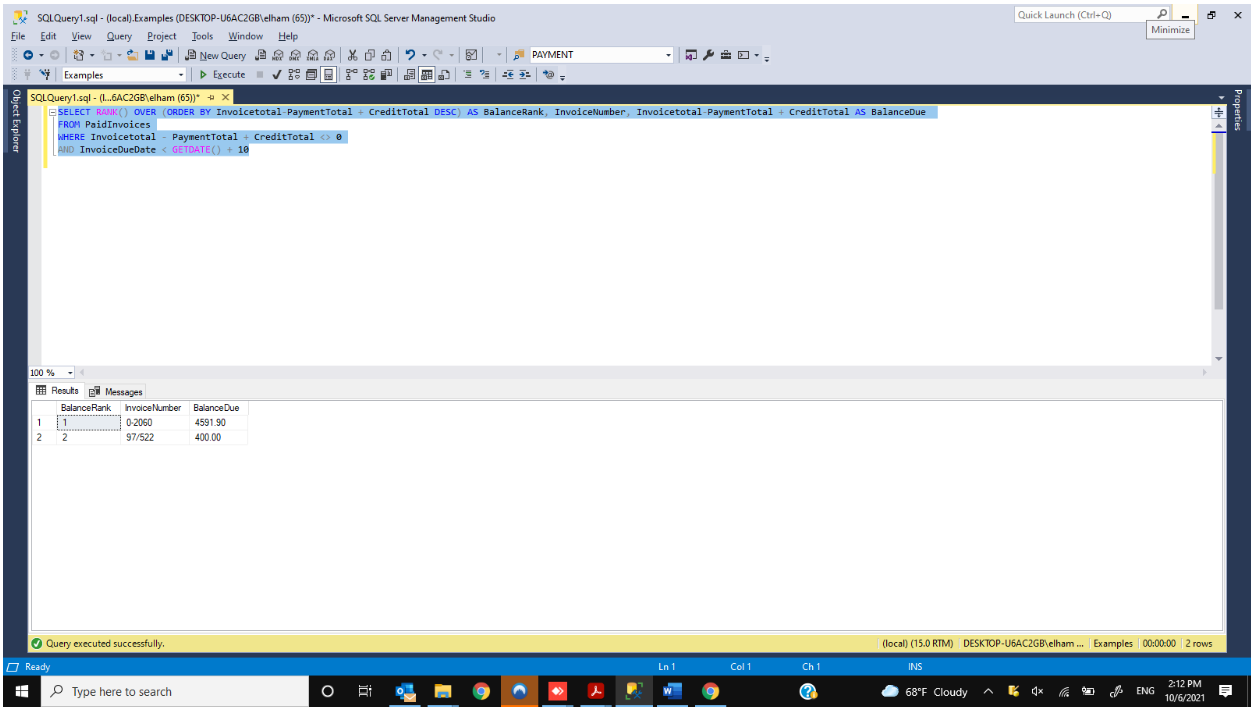
```sql
SELECT InvoiceNumber, Invoicetotal-PaymentTotal + CreditTotal AS BalanceDue
FROM PaidInvoices
WHERE Invoicetotal - PaymentTotal + CreditTotal <> 0
AND InvoiceDueDate < EOMONTH(GETDATE())
```

	InvoiceNumber	BalanceDue
1	97/522	400.00
2	0-2060	4591.90

Query executed successfully.

4. Add a column to the query described in question 2 that uses the RANK() function to return a column named BalanceRank that ranks the balance due in descending order.

```
SELECT InvoiceNumber, Invoicetotal-PaymentTotal + CreditTotal AS BalanceDue
FROM PaidInvoices
WHERE Invoicetotal - PaymentTotal + CreditTotal <> 0
AND InvoiceDueDate < GETDATE() + 10;
```



```
SELECT RANK() OVER (ORDER BY Invoicetotal-PaymentTotal + CreditTotal DESC) AS BalanceRank, InvoiceNumber, Invoicetotal-PaymentTotal + CreditTotal AS BalanceDue
FROM PaidInvoices
WHERE Invoicetotal - PaymentTotal + CreditTotal <> 0
AND InvoiceDueDate < GETDATE() + 10
```

	BalanceRank	InvoiceNumber	BalanceDue
1	1	0-2060	4591.90
2	2	97/522	400.00

Query executed successfully.

Thank you for your time.

Sincerely,

Seyed Alireza Zarrin Mehr