

Della's Design Solution

CIS 3365

Transition Technologies

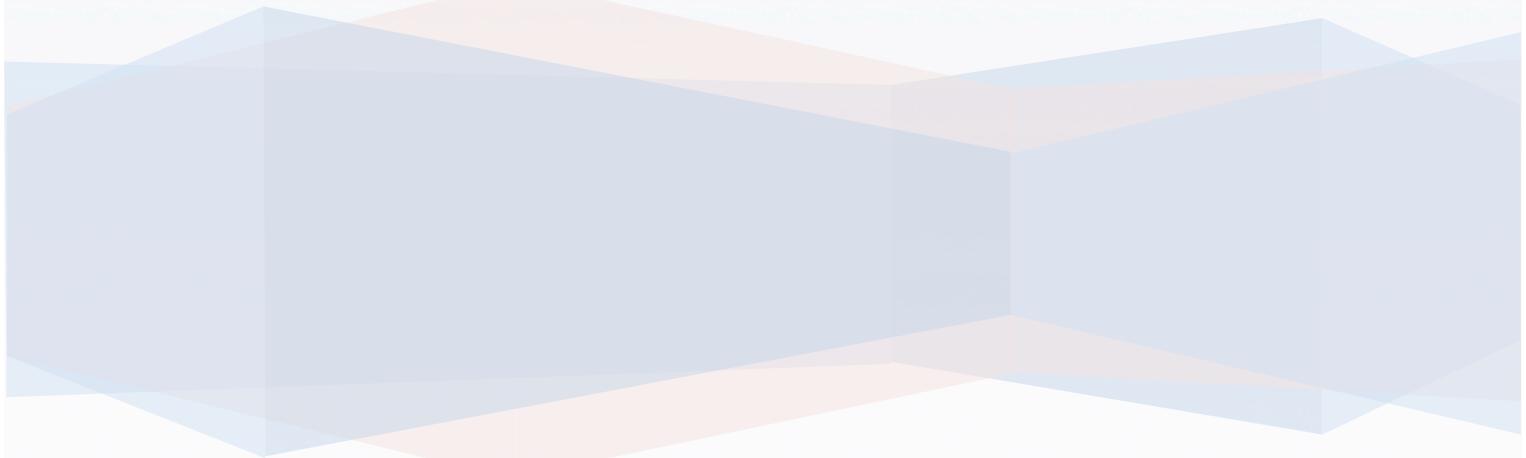


Table of Contents

<u>EXECUTIVE SUMMARY</u>	<u>3</u>
<u>CLIENT INFORMATION</u>	<u>4</u>
<u>BENEFITS AND COST</u>	<u>5</u>
<u>PROJECT APPROACH</u>	<u>6</u>
<u>SOLUTION</u>	<u>7</u>
 DESCRIPTION OF REQUIRED FORMS AND REPORTS	8
 KEY DECISIONS	47
<u>TESTING PROCESS</u>	<u>49</u>
<u>PROJECT IMPROVEMENTS</u>	<u>53</u>
<u>PROJECT DATABASE MAINTENANCE ISSUES</u>	<u>55</u>
<u>LESSONS LEARNED</u>	<u>57</u>
<u>PROJECT SUMMARY</u>	<u>59</u>
<u>REFERENCES</u>	<u>60</u>
<u>APPENDICES</u>	<u>60</u>

Executive Summary

Della Schapiro founded Della's Design Solution on August 2015. She designs websites and graphic designs for clients based on client requests. The service costs are dependent on whether the client needs a website design or a graphic design.

Della Schapiro currently keeps track of her clients and project prices through tables on Microsoft Word and doesn't have an application to input client information along with project and price details that can be adjusted accordingly across different tables. The current system used by Della's Design Solution solely operates through manual creation and calculation and it has separate files and folders for clients' information such as invoices and proposals.

The problems posed by the current system used by Della include inefficient quote process, inefficient data management process and inefficient calculating process, to mention but a few, hence the need for an automated system that can compile all the design business information into one application – a system that has the ability to track clients, projects, and rates. Furthermore, the application will be completely accessible by Della to make any changes to prices, projects, and clients.

As part of the data gathering aspect of this project, the team engaged Della in a live interview. The interview was centered on data storage and the processes clients will go through to purchase a web/graphic design service. The data gathered during the interview include documents, emails, past invoices, facts, history, and purpose of the business that Transition Technologies was building the application for.

A new system has been proposed to solve the problems Della's Design Solutions is currently facing. This system, after series of successful prototype testing, will take the fields Della has in her current system and integrate them with new data fields and tables for quoting, storage and invoicing. In the context of web designing to meet Della's required strategic business needs, Transition Technologies has proposed a new system that can input client information and their project scope to help generate an invoice, break down the different options to a project to determine the service prices, keep track of hours, and calculate the deposit for projects. This new application will store all client information, the type of project requested, work type, additional services, hours, and be able to calculate various prices.

Two options were proposed for the system operation. The first option is to run the application off a Windows operating system and use Microsoft Access as a new system for Della's Design Solutions. The major downside to this option is the additional cost (to the client - Della) of Microsoft Access and the fact that it could only be accessed locally. The second option is the proposal to develop and host the database for the application on Microsoft Azure. The developed application would be able to run on Microsoft Azure's Cloud servers once complete and available as long as the service is available. This second option offers no additional cost to the client since Della's Design Solutions has access to

Microsoft Azure. This option will also allow for the full implementation of the entire proposed system from any computer hence is Transition Technologies' preferred option.

Transition Technologies' plans to complete the prototype's final system test phase on schedule. The application, upon successful tests on multiple machines, will be delivered to Della's Design Solutions together with the consumer user guide.

Client Information

The client is Della Schapiro of Della's Design Solutions and she keeps records of clients by adding and updating information to a Microsoft word document manually. She needs a database to store and search for projects and clients that are required. Our group Transition Technologies' decided to create a new application for Della's Design Solutions with features that will allow the input of client information and the project options to help generate invoices keep track of project prices and hours and break down the total price of the project with additional costs calculated. With the all the data, the deposit required on a project can be generated to help create a final invoice for the Della's Design Solutions and the client.

The current system used by Della Schapiro consists of receiving email requests for projects then manually calculates options and related prices for data such as total prices, hours, and down payments onto the Microsoft word document. The options provided are the size of websites, the website type, graphic design size, and additional functions. The total price and deposit for each project is dependent on what options need to be created. Then a final copy is sent back to the consumer via email for project approval and initialization.

Through this new application, Della Schapiro needs to be able to break down services, track prices, calculate deposits, calculate markups, and generate forms. This application will be accessible through the internet and the application will run on SQL with a Java GUI front end. It should be able to login, input, output, and calculate the costs and requirements of each project according to what is requested. The client would like to be able to protect sensitive data with a login feature, the ability to browse through the project and make any changes or additions. The system should also be able to calculate the project cost and the resulting deposit.

Our sponsor Della Schapiro of Della's Design Solutions has agreed to sponsor our group in creating this database for her business in order to make data input and output more efficient to both the client and user. Our group works closely with Della Schapiro in order to adjust the database to what she requires for her business.

Benefits and Cost

The costs and benefits of switching Della's Design Solutions from the current Microsoft word document system to a SQL database system would increase the productivity of both our client and users because of the reduced effort to track each project and client details. The total cost of ownership for Della Schapiro to keep the application running will include time and resources such as hardware, software, and additional setup and maintenance costs. Between the proposed system options for the database, it was decided that Microsoft Azure would be the most feasible platform.

The costs and time to create and install the application had been estimated and compared with other options to see the overall benefits and feasibility. Comparatively it was decided that creating the database on Microsoft Azure is estimated to be more expensive than creating the database on Microsoft Access, but overall offers more functions and is more suitable for possible future updates. The estimate for installation is approximately \$640 with about 16 hours to set up for completion, the analysis and design will be about \$2400 with 60 hours to complete, the GUI design will be about \$1200 for 30 hours to complete, \$400 to test the GUI for 10 hours, the build and design of the application will be about \$1600 with about 40 hours to complete, the testing and debugging will be about \$800 with 20 hour completion, the training of operators will be about \$200 for 5 hours, \$6750 for the transfer of existing data for 15 hours, and \$400 to test and update the database for 10 hours. The total cost of ownership for Microsoft Azure is a total of about \$14,390 and 208 hours to complete the implementation. Afterwards there will be a recurring fee to keep the system functional and online. These are all estimates and the costs and time of each objective can vary depending on the situation of the systems.

The decision to design the database on Microsoft Azure instead of Microsoft Access is because Microsoft Azure offers more usability and reliable features such as online access. The use of Microsoft Azure would allow the organization to create and access the application across different machines on the cloud. The downtime of the cloud is rare, so it would be available for most of the time with little maintenance.

There would be minimal training for the management and entry of data because there is no handling of technical side. The options and costs of the services provided will be calculated and an invoice can be generated from information provided by the application based on the additional markups and costs. The benefits of the application would increase the productivity of Della's Design Solutions and the consumer compared to the current Microsoft word document system.

Project Approach

The approach our group took for designing the database are results of the discussions of the activities that the team undertook to solve this client's database requirements. This database design and implementation process that was took place during the semester.

List of database design steps:

1. Identify entities and attributes from business process
2. Establish correct relationships between tables
3. Identified primary and secondary keys
4. Addressed M:M relationships to ensure that there was an associative table
5. Created the data dictionary and identify the metadata

List of database creation steps:

1. Created the tables as it was presented in the ERD
2. Assigned primary keys and secondary keys to the tables
3. Created bulked data to load into the database
4. Loaded bulk data scripts to load the data into the database
5. Created user defined functions and queries
6. Ensured that cascade update and delete was included

List of GUI creation steps:

1. Linked the database to the Java application
2. Created database forms
3. Created login screen
4. Created report functionality for different reports reports

List of testing steps:

1. Tested the data in the SQL database by running queries and the user functions
2. Validated relationships in the ERD and Database design
3. Addressed bugs in our SQL code
4. Verified the connection between the Java application and SQL database
5. Loaded the database multiple time to ensure that the scripts had no bugs
6. Tested all the forms in the GUI to ensure input and output of the data
7. Formatted the bulk data as needed in order to load into the database
8. Compiled the create, alter, bulk insert scripts, and sample data files
9. Verified for referential integrity

List of implementation steps:

Implement system in business
Checked for system compatibility

Installed software and database for the client
Hold training sessions in groups for users and managers
Monitored the performance of the database
Performed updates for the systems to ensure that they were up to date with patches

List of maintenance steps:

Scheduled automated full database backups
Scheduled database clean up
Scheduled incremental backups
Scheduled updates for the database

Solution

The database designed this semester, was simplified to what was originally, needed versus wants. Last semester when talking with our client, we discovered everything was manually done. From back forth emails about contract of what is going to be done, invoices being sent, and generating client interest. Della also specified that she wanted a way to track cost, calculate deposit and markup, generate forms, break down services offered, and track cost. What we did to meet the necessary requirements was create a simplified database from 40 plus tables to fewer than 20 tables in SQL Server 2017 and linked database to a Java application.

On the Java application we first created a login page for our client to access database and get information from database. Whatever query is ran, it will show the result as a table in the application below the text box.

While developing the database, we decided to come up with three tables that would make things easier for Della. First, we decided to create a TempClient entity that would store client info who have inquired about Della's services. If no response is made from them within a week, their information will be deleted automatically. However, if a client decides to further pursue Della services, the information stored in TempClient will be saved for future reference. This will help differentiate interest from customers. Second, we added a Balance table that would record any unpaid balance (deposit) from Della's customers and be removed when paid. When the balance is paid, the information goes to PaidToDate table which records when payments were made to Della. In order for Della to track cost, we implemented a Requirements table that holds information regarding programs used to track cost of those programs as well as added expenses. Lastly, we created a Service table which records description of service, service date, Clientid, WebsiteJobid, and GraphicDesignJobid to track past services. Additionally , the WebmasterLogin associates the Client to a username and password.

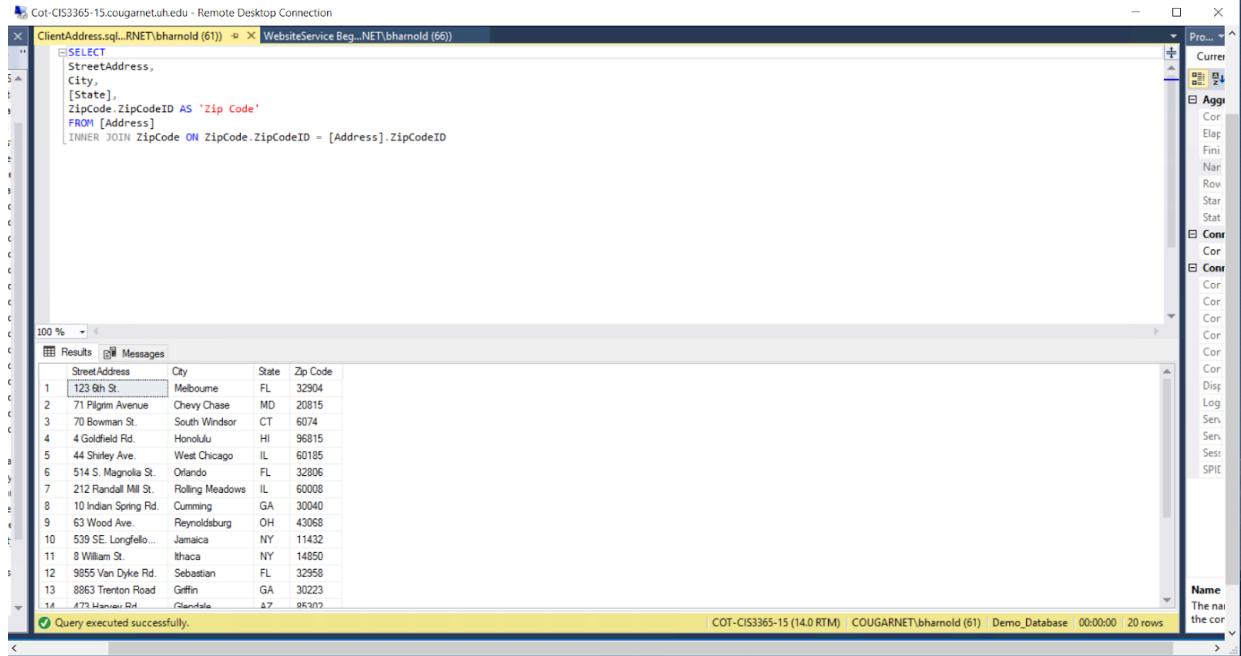
In the early stages of the project we were troubled with 40 plus tables. As some tables, functionality wise, were necessary but weren't mission critical gave us the hardship of trying to figure out how reduce the size of tables to no more than 20. Once we broke the tables down to the core we were than able to determine and bring back certain tables that made sense functionality wise in the database. We also realized that

not all objectives discussed last semester were going to be met due to time constraints this semester.

Description of required forms and reports

ClientAddress

ClientAddress query results all addresses associated with clients. Reason why our client would want to use this is to specify a certain address for a client.



The screenshot shows a SQL Server Management Studio window with two tabs: 'ClientAddress.sql' and 'WebsiteService Beg...NET'. The 'ClientAddress.sql' tab contains the following SQL code:

```
SELECT
    StreetAddress,
    City,
    [State],
    ZipCode.ZipCodeID AS 'Zip Code'
FROM [Address]
INNER JOIN ZipCode ON ZipCode.ZipCodeID = [Address].ZipCodeID
```

The 'Results' tab displays the query results in a table:

	StreetAddress	City	State	Zip Code
1	123 Bhn St.	Melbourne	FL	32904
2	71 Pilgrim Avenue	Chevy Chase	MD	20815
3	70 Bowman St.	South Windsor	CT	6074
4	4 Goldfield Rd.	Honolulu	HI	96815
5	44 Shirley Ave	West Chicago	IL	60185
6	514 S. Magnolia St.	Orlando	FL	32806
7	212 Randall Mill St.	Rolling Meadows	IL	60008
8	10 Indian Spring Rd.	Cumming	GA	30040
9	63 Wood Ave.	Reynoldsburg	OH	43068
10	539 SE. Longfello...	Jamaica	NY	11432
11	8 William St.	Ithaca	NY	14850
12	9655 Van Dyke Rd.	Sebastian	FL	32958
13	8863 Trenton Road	Griffith	GA	30223
14	472 Hanover Rd	Glen Dale	47	85302

At the bottom of the results pane, it says 'Query executed successfully.' and shows the session details: COT-CIS3365-15 (14.0 RTM) | COUGARNET\bharnold (61) | Demo_Database | 00:00:00 | 20 rows.

ClientLogin

ClientLogin results in matching the domain with the specific username and password of the client. This will allow Della to relate her clients to their domain with login credentials.

ClientLogin.sql - ...UGARNET\iakim (57) x

```
SELECT
    ClientDomain,
    Username,
    [Password]
FROM ClientDomain
INNER JOIN
WebmasterLogin ON WebmasterLogin.WebmasterLoginID = ClientDomain.WebmasterLoginID
```

100 % < >

Results Messages

	ClientDomain	Username	Password
1	website.com	tlowny	password
2	myspace.com	athynes	12345
3	dubyastep.com	mkarfs	totallynotpassword
4	howtomm.com	ksterinfeld	baby2018
5	got.com	hsetzer	dragon
6	gotapples.com	abagley	timcook
7	texaslove.com	rjanas	ilovetexas
8	thealamofostho.com	kmandeville	rememberthealamo
9	anyme.com	ealphones	ninja
10	doublepumpshotgun.com	lwidman	fortnite
11	theydontunderstand.com	lphelps	strnger
12	batlink.com	klarock	darkknight7
13	invisibleplane.com	vcoons	wonderwoman
14	urathananain.com	mehirah	jeanreinoue

Query executed successfully.

COT-CIS3365-15 (14.0 RTM) COUGARNET\iakim (57) Demo_Database 00:00:00 20 rows

ClientPhone

Associates area code with phone number. Della can use this query to find area code of given phone number to contact her clients.

The screenshot shows a SQL Server Management Studio (SSMS) window titled "Cot-CIS3365-15.cougar.net.uh.edu - Remote Desktop Connection". There are four tabs at the top: "Complete.Client.sql...barnold (69)", "ClientPhone.sql ...RNET\barnold (68)" (selected), "Function - FindAdd...NET\barnold (67)", and "ClientLogin.sql ...RNET\barnold (64)". The main area displays a query result grid. The query is:

```
SELECT
    AreaCode/AreaCodeID,
    Number FROM PhoneNumber
    INNER JOIN AreaCode ON AreaCode/AreaCodeID = PhoneNumber/AreaCodeID;
```

The results grid shows 20 rows with columns "AreaCodeID" and "Number".

	AreaCodeID	Number
1	361	9999999
2	713	6905413
3	281	3308004
4	209	4531289
5	417	7413658
6	503	1973482
7	202	1234567
8	254	3571598
9	330	4826716
10	401	9102973
11	270	5643212
12	501	4683218
13	242	8920156
14	630	7941254

At the bottom, a message says "Query executed successfully." and the status bar indicates "COT-CIS3365-15 (14.0 RTM) COUGARNET\barnold (68) Demo_Database 00:00:00 20 rows".

Complete Client

Results in all client information being shown. Dicifiers between a temp client and actual client as well list all information associated with that client. She can use this query to see how many clients she has as well as potential clients.

Cot-CIS3365-15.cougartech.uh.edu - Remote Desktop Connection

Complete Client.sql...NET\bharnold (69) ClientPhone.sql...RNET\bharnold (68) Function - FindAdd...NET\bharnold (67) ClientLogin.sql...RNET\bharnold (64)

```

SELECT
    FirstName AS 'First Name',
    LastName AS 'Last Name',
    StreetAddress AS 'Street Address',
    City,
    [State],
    ZipCode.ZipCodeID,
    CONCAT('(', AreaCode.AreaCodeID, ')', ' ', Number) AS 'Phone Number',
    Email, CASE WHEN TempClient = 1 THEN 'Yes' ELSE 'No' END AS 'Temp Client'
    FROM PhoneNumber
    INNER JOIN AreaCode ON AreaCode.AreaCodeID = PhoneNumber.AreaCodeID
    INNER JOIN Client ON Client.ClientID = PhoneNumber.PhoneNumberID
    INNER JOIN [Address] ON [Address].AddressID = Client.AddressID
    INNER JOIN ZipCode ON ZipCode.ZipCodeID = [Address].ZipCodeID

```

100 %

Results Messages

	First Name	Last Name	Street Address	City	State	ZipCodeID	Phone Number	Email	Temp Client
1	Timothy	Lowry	123 8th St.	Melbourne	FL	32904	(361) 9999999	tlowry@website.com	No
2	Anna	Rhynes	71 Pilgrim Avenue	Chevy Chase	MD	20815	(713) 6905413	arhynes@myspace.com	No
3	Mindi	Karla	70 Bowman St.	South Windsor	CT	6074	(281) 308004	mkarla@dubyastep.com	No
4	Kent	Steinfeldt	4 Goldfield Rd.	Honolulu	HI	96815	(209) 4531289	ksteinfeldt@howtomom.com	Yes
5	Howard	Sitzer	44 Shirley Ave.	West Chicago	IL	60185	(417) 7143658	hsitzer@gvt.com	No
6	Arlene	Bagley	514 S. Magnolia St.	Orlando	FL	32806	(503) 1973482	abagley@gotapples.com	No
7	Norman	Janae	212 Randall Mill St.	Rolling Meadow	IL	60006	(202) 1234567	rjanae@xoxalove.com	No
8	Katlyn	Mandeville	10 Indian Spring Rd.	Cumming	GA	30040	(254) 3571598	kmmandeville@thealarmofosho.com	No
9	Emilio	Alphonse	63 Wood Ave.	Reynoldsburg	OH	43068	(330) 4326716	ealphonese@anyame.com	No
10	Liliana	Widman	539 SE Longfellow Street	Jamaica	NY	11432	(401) 9102973	lwidman@doublepumpshotgun.com	Yes
11	Leena	Phelps	8 William St.	Ithaca	NY	14850	(270) 5643212	lphelps@heydorunderstand.com	No
12	Korey	Larock	9855 Van Dyke Rd.	Sebastian	FL	32958	(501) 4683218	klarock@batlink.com	No
13	Vicki	Coons	8863 Trenton Road	Giffin	GA	30223	(242) 8920156	vcoons@invisibleplane.com	No
14	Masakhi	Shirah	473 Hanover Rd.	Glendale	AZ	85307	(623) 7911254	mehirsh@justiniansain.com	Yes

Query executed successfully.

COT-CIS3365-15 (14.0 RTM) | COUGARNET\bharnold (69) | Demo_Database | 00:00:00 | 20 rows

Function - FindAddon

This query results in filtering services done by description and addon. Shows price of the addon and how many times that addon has been done for the description.

The screenshot shows a Microsoft SQL Server Management Studio (SSMS) window. The title bar indicates the connection is to 'COT-CIS3365-15.cougar.net.uh.edu' via 'Remote Desktop Connection'. There are four tabs open in the tabs bar: 'Complete Client.sq...', 'ClientPhone.sq...', 'Function - FindAdd...', and 'ClientLogin.sq...'. The 'Function - FindAdd...' tab is active, displaying a query result set.

```

SELECT [Services].[ServicesID], [Services].[Services_Description], Addons.Addon, Addons.Price, Service2Addon.Quantity
FROM Addons INNER JOIN ([Services] INNER JOIN Service2Addon ON [Services].ServicesID = Service2Addon.ServiceID)
ON Addons.AddonID = Service2Addon.AddonID
WHERE Addons.AddonID = dbo.FindAddon('SEO')
ORDER BY ServicesID ;

```

The results grid shows the following data:

ServicesID	Services_Description	Addon	Price	Quantity	
1	2	Wedding Planner Website	SEO	50.00	5
2	4	Online Support Forum	SEO	50.00	10
3	6	Antique Store Website	SEO	50.00	1
4	14	Auction Website	SEO	50.00	20
5	16	Arts and Crafts Online Store	SEO	50.00	30
6	20	Beauty Products Online Store	SEO	50.00	25

At the bottom of the results pane, a message states 'Query executed successfully.' and provides execution details: 'COT-CIS3365-15 (14.0 RTM) | COUGARNET\bharnold (67) | Demo_Database | 00:00:00 | 6 rows'.

Function - FindDomain

Filters domain with login credentials based on domain of choice. Della can use this to find a specific domain for a certain client.

```
Function - FindDo...GARNET\jakim (57)  ×
SELECT
    ClientDomain,
    Username,
    [Password]
FROM ClientDomain
INNER JOIN
    WebmasterLogin ON WebmasterLogin.WebmasterLoginID = ClientDomain.WebmasterLoginID
WHERE ClientDomainID = dbo.FindDomain('myspace.com')

100 %  ( 100 %  )
Results Messages
ClientDomain Username Password
1 myspace.com afhyne 12345

Query executed successfully. COT-CI53365-15 (14.0 RTM) | COUGARNET\jakim (57) | Demo_Database | 00:00:00 | 1 rows
```

GraphicDesignService Beginning

This query results association between client information, services, balance, and paid. Based off of client information, it will result in the description of the service, job type, balance, paid, and when the service took place. Della can use this query to see past services she has done.

```

SELECT FirstName, LastName, sub.Services_Description, sub.GraphicDesignJobID, sub.BalanceID, sub.PaidID, sub.Service_Date 
FROM Client
INNER JOIN (
    SELECT ServicesID, [Services_Description], ClientID, GraphicDesignJobID, BalanceID, PaidID, Service_Date
    FROM [Services] WHERE [Services].GraphicDesignJobID IS NOT NULL
) sub ON sub.ClientID = Client.ClientID

```

	FirstName	LastName	Services_Description	GraphicDesignJobID	BalanceID	PaidID	Service_Date
1	Timothy	Lowry	Spring Sale Advertisment	1	1	1	2018-04-18
2	Howard	Setzer	New Company logo	5	5	5	2017-10-31
3	Norman	Janas	Website Banner	7	7	7	2017-06-20
4	Katlyn	Mandeville	Weekly Advertisment	12	8	8	2017-10-15
5	Emilio	Alphonse	Custom Website Image	2	9	9	2017-09-27
6	Liliana	Widman	Custom Design Seasonal Template	10	10	10	2017-10-23
7	Leena	Phelps	Revised Company logo	3	11	11	2016-12-15
8	Korey	Larock	Newspaper Advertisment	20	12	12	2017-09-12
9	Vicki	Coons	Alternate Company logo	11	13	13	2017-06-12
10	Cara	Pickford	Custom Design Template	17	17	17	2017-01-29
11	Max	Waters	Revised Website image	9	19	19	2017-06-26

GraphicDesignService DONE

This query results in all client information associated with service provided, job description, balance, deposit, paid, and whether they are a temp client or not. Using this query will allow Della to explicitly see how much is due from that client, the amount of deposit, and how has been paid. Because it is linked to services she will also be able to see when the service started and whether or not they are a potential client.

The screenshot shows a Microsoft SQL Server Management Studio window with the following details:

- Title Bar:** GraphicDesignService.DONE.sql - COT-CIS3365-15.Demo_Database (COUGARNET\akim (61)) - Microsoft SQL Server Management Studio
- Toolbar:** File, Edit, View, Query, Project, Debug, Tools, Window, Help
- Query Editor:** Contains a complex multi-table query involving multiple joins and subqueries across tables like ClientDomain, Client, ZipCode, Service, and Job.
- Results Grid:** Displays 11 rows of data with columns: Name, Address, Phone Number, Client Domain, email, Service, Job Description, Balance, Deposit, Paid, PaidToDate, Service Date, Temp Client.
- Status Bar:** Shows "Query executed successfully." and other session details.

Name	Address	Phone Number	Client Domain	email	Service	Job Description	Balance	Deposit	Paid	PaidToDate	Service Date	Temp Client
1 Timothy Lowry	123 6th St., Melbourne, FL 32904	(361) 9999999	website.com	tlowry@website.com	Spring Sale Advertisement	Ad	600.00	120.00	100.00	2018-04-09	2018-04-18	No
2 Howard Setzer	44 Shirley Ave., West Chicago, IL 60185	(417) 7413658	got.com	hsetzer@got.com	New Company logo	Logo	300.00	60.00	120.00	2018-05-04	2017-10-31	No
3 Norman Janas	212 Randall Mill St., Rolling Meadows, IL 6008	(202) 1234567	texaslove.com	njanas@texaslove.com	Website Banner	Banner	480.00	96.00	350.00	2018-04-14	2017-06-20	No
4 Kelyn Mandeville	10 Indian Spring Rd., Cumming, GA 30040	(254) 3571598	thealamofosho.com	kmandeville@thealamofosho.com	Weekly Advertisement	Ad	330.00	66.00	200.00	2018-04-10	2017-10-15	No
5 Emilio Aphorne	63 Wood Ave., Reynoldsburg, OH 43068	(330) 4826716	anyme.com	eaphorner@nyyme.com	Custom Website Image	Image	450.00	90.00	0.00	NULL	2017-09-27	No
6 Liliana Widman	539 SE Longfellow Street, Jamaica, NY 11432	(401) 9102973	doubleupmpshotgun.com	lwidman@doubleupmpshotgun.com	Custom Design Seasonal Template	Design	360.00	72.00	25.00	2018-04-18	2017-10-23	Yes
7 Leena Phelps	8 Willam St., Ithaca, NY 14850	(270) 5643212	theydontunderstand.com	lphelps@theydontunderstand.com	Revised Company logo	Logo	450.00	90.00	400.00	2018-04-02	2016-12-15	No
8 Korey Larock	9855 Van Dyke Rd., Sebastian, FL 32958	(501) 4683218	batlink.com	klarock@batlink.com	Newspaper Advertisement	Ad	630.00	126.00	2000.00	2018-03-01	2017-09-12	No
9 Vicki Coons	8863 Trenton Road, Griffin, GA 30223	(242) 8920156	invisibleplane.com	vcoons@invisibleplane.com	Alternate Company logo	Logo	420.00	84.00	30.00	2018-03-10	2017-06-12	No
10 Cara Pickford	7142 Mechanic Street, Goose Creek, SC 29445	(351) 5023070	default.com	cpickford@default.com	Custom Design Template	Design	630.00	126.00	0.00	NULL	2017-01-29	Yes
11 Max Waters	227 Broad Street, Valrico, FL 33594	(479) 8885892	supersad.com	mwaters@supersad.com	Revised Website image	Image	150.00	30.00	550.00	2018-04-20	2017-06-26	No

GraphicDesignService with BALANCE AND PAID

This query shows when the client made a payment towards a balance and when the service began. Della can use this to determine which clients have paid either deposit or full amount towards balance in association with service date to paid date.

```

Cot-CIS3365-T5.cougarinet.uh.edu - Remote Desktop Connection
Database Execute Debug Database Tools Help
GraphicDesignServi...NET\bharnold (73)) > X GraphicDesignServi...NET\bharnold (72) GraphicDesignServi...NET\bharnold (71) Function - FindDo...NET\bharnold (70)
SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.GraphicDesignJobID, sub2.Balance, sub2.Deposit, Paid, PaidToDate, sub2.Service_Date FROM Paid
INNER JOIN (
    SELECT sub.ServicesID, sub.[Services_Description], sub.ClientID, sub.GraphicDesignJobID, Balance, Deposit, PaidID, Service_Date FROM Balance
    INNER JOIN (
        SELECT ServicesID, [Services_Description], ClientID, GraphicDesignJobID, BalanceID, PaidID, Service_Date
        FROM [Services] WHERE [Services].GraphicDesignJobID IS NOT NULL ) sub ON sub.BalanceID = [Balance].BalanceID ) sub2
ON sub2.PaidID = Paid.PaidID

```

Results

ServicesID	Services_Description	ClientID	GraphicDesignJobID	Balance	Deposit	Paid	PaidToDate	Service_Date
1	Spring Sale Advertisement	1	1	600.00	120.00	100.00	2018-04-09	2018-04-18
2	New Company logo	5	5	300.00	60.00	1200.00	2018-05-04	2017-10-31
3	Website Banner	7	7	480.00	96.00	350.00	2018-04-14	2017-06-20
4	Weekly Advertisement	8	12	330.00	66.00	200.00	2018-04-10	2017-10-15
5	Custom Website Image	9	2	450.00	90.00	0.00	NULL	2017-09-27
6	Custom Design Seasonal Template	10	10	360.00	72.00	25.00	2018-04-18	2017-10-23
7	Revised Company logo	11	3	450.00	90.00	400.00	2018-04-02	2016-12-15
8	Newspaper Advertisement	12	20	630.00	126.00	2000.00	2018-03-01	2017-09-12
9	Alternate Company logo	13	11	420.00	84.00	30.00	2018-03-10	2017-06-12
10	Custom Design Template	17	17	630.00	126.00	0.00	NULL	2017-01-29
11	Revised Website image	19	9	150.00	30.00	550.00	2018-04-20	2017-06-26

GraphicDesignServices with Partial Client and Complete Address and Complete Phone Number and Domain

The query results in client information being associated with services, their domain, balance, deposit, amount paid, and dates associated with the service and paid amounts towards balance. Della can use this query to see all details about client and services (payments, dates, domain, etc).

The screenshot shows a Microsoft SQL Server Management Studio window. The title bar reads "GraphicDesignService with Partial Client and Complete Phone Number and Domain.sql - COT-CIS3365-15.Demo_Database (COUGARNET)\jakim (62) - Microsoft SQL Server Management Studio". The query pane contains a complex multi-table join query. The results pane displays a grid of 11 rows, each containing client information, service details, and domain data. The columns include ClientID, FirstName, LastName, StreetAddress, City, State, ZipCodeID, AreaCodeID, Number, ClientDomain, email, ServicesID, Services_Description, GraphicDesignJobID, Balance, and Paid. The status bar at the bottom indicates "Query executed successfully." and shows the date and time as 4/26/2018 12:27 PM.

ClientID	FirstName	LastName	StreetAddress	City	State	ZipCodeID	AreaCodeID	Number	ClientDomain	email	ServicesID	Services_Description	GraphicDesignJobID	Balance
1	Timothy	Lowry	123 6th St.	Melbourne	FL	32904	361	9999999	website.com	tlowry@website.com	1	Spring Sale Advertisement	1	600.00
2	Howard	Setzer	44 Shirley Ave.	West Chicago	IL	60185	417	7413658	got.com	hsetzer@got.com	5	New Company logo	5	300.00
3	Norman	Janas	212 Randall Mill St.	Rolling Meadows	IL	60008	202	1234567	texaslove.com	rjanas@texaslove.com	7	Website Banner	7	480.00
4	Karlyn	Mandeville	10 Indian Spring Rd.	Cumming	GA	30040	254	3571598	thealemonfoho.com	kmmandeville@thealemonfoho.com	8	Weekly Advertisement	12	330.00
5	Emilio	Alphonse	63 Wood Ave.	Reynoldsburg	OH	43068	330	4826716	anyime.com	ealphonse@anyime.com	9	Custom Website Image	2	450.00
6	Liliana	Widman	539 SE. Longfellow Street	Jamaica	NY	11432	401	9102973	doublepumpshotgun.com	lwidman@doublepumpshotgun.com	10	Custom Design Seasonal Template	10	360.00
7	Leena	Phelps	8 William St.	Ithaca	NY	14850	270	5643212	theydonntunderstand.com	lphelps@theydonntunderstand.com	11	Revised Company logo	3	450.00
8	Korey	Larock	9855 Van Dyke Rd.	Sebastian	FL	32958	501	4683216	ballrlink.com	klarock@ballrlink.com	12	Newspaper Advertisement	20	630.00
9	Vicki	Coons	8863 Trenton Road	Giffin	GA	30223	242	8920156	invisibleplane.com	vcoons@invisibleplane.com	13	Alternate Company logo	11	420.00
10	Cara	Pickford	7142 Mechanic Street	Goose Creek	SC	29445	351	5023070	default.com	cpickford@default.com	17	Custom Design Template	17	630.00
11	Max	Waters	227 Broad Street	Valrico	FL	33594	479	888592	superad.com	mwaters@superad.com	19	Revised Website image	9	150.00

GraphicDesignService w/ Partial Client, Complete Address, Complete Phone Number

This query takes the client information and associates it with services, their domainID, graphic design job id, balance, deposit, paid, and dates tables. This simplifies the previous query by simply applying a filter to less information and associating more ID's.

GraphicDesignService w/ Partial Client, Complete Address

This query results in the client information associated with the graphic design and service ID type Della has done for them. Della can use this to filter clients with the service she has done for them in the past.

Cot-CIS3365-15.cougar.net.uh.edu - Remote Desktop Connection

inService with Partial Client and Complete Address.sql - COT-CIS3365-15.Demo_Database (COUGARNET\bharnold (76)) - Microsoft SQL Server Management Studio

```

SELECT sub5.ClientID, sub5.FirstName, sub5.LastName, sub5.StreetAddress, City, [State], ZipCode, ZipCodeID, sub5.PhoneNumberID, sub5.ClientDomainID, sub5.email,
sub5.ServicesID, sub5.[Services_Description], sub5.GraphicDesignJobID, sub5.Balance, sub5.Deposit, sub5.Paid,
PaidToDate, sub5.Service_Date, sub5.TempClient FROM ZipCode
    INNER JOIN (
        SELECT sub4.ClientID, sub4.FirstName, sub4.LastName, StreetAddress, ZipCodeID, sub4.PhoneNumberID, sub4.ClientDomainID, sub4.email,
        sub4.ServicesID, sub4.[Services_Description], sub4.GraphicDesignJobID,
        sub4.Balance, sub4.Deposit, sub4.Paid, PaidToDate, sub4.Service_Date, sub4.TempClient FROM [Address]
    ) sub4 ON sub5.ZipCodeID = sub4.ZipCodeID
    INNER JOIN (
        SELECT Client.ClientID, FirstName, LastName, AddressID, PhoneNumberID, ClientDomainID, email, sub3.ServicesID, sub3.[Services_Description], sub3.GraphicDesignJobID,
        sub3.Balance, sub3.Deposit, Paid, PaidToDate, sub3.Service_Date, TempClient
    ) Client ON sub4.AddressID = Client.AddressID
    INNER JOIN (
        SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.GraphicDesignJobID, sub2.Balance, sub2.Deposit, Paid, PaidToDate, sub2.Service_Date FROM Paid
    ) sub2 ON sub4.ClientID = sub2.ClientID
    INNER JOIN (
        SELECT sub.ServicesID, sub.[Services_Description], sub.ClientID, sub.GraphicDesignJobID, Balance, Deposit, PaidID, Service_Date FROM Balance
    ) sub ON sub.ClientID = sub2.ClientID
    WHERE [Services] GraphicDesignJobID IS NOT NULL ) sub ON sub.ServicesID = [Balance].BalanceID ) sub2
    ON sub2.PaidID = Paid.PaidID ) sub3
    ON sub3.ClientID = Client.ClientID ) sub4

```

ClientID	FirstName	LastName	StreetAddress	City	State	ZipCodeID	PhoneNumberID	ClientDomainID	email	ServicesID	Services_Description	GraphicDesignJobID
1	Timothy	Lowry	123 8th St.	Melbourne	FL	32904	1	1	tlowry@website.com	1	Spring Sale Advertisement	1
2	Howard	Setzer	44 Shirley Ave.	West Chicago	IL	60185	5	5	hsetzer@yot.com	5	New Company logo	5
3	Norman	Janae	212 Randall Mill St.	Rolling Meadows	IL	60008	7	7	rjana@texaselove.com	7	Website Banner	7
4	Karlyn	Mandeville	10 Indian Spring Rd.	Cumming	GA	30040	8	8	kmmandeville@thealamofosho.com	8	Weekly Advertisement	12
5	Emilio	Alphonse	63 Wood Ave.	Reynoldsburg	OH	43068	9	9	ealphonese@onyme.com	9	Custom Website Image	2
6	Liliana	Widman	539 SE Longfellow Street	Jamaica	NY	11432	10	10	lwidman@doublepumpahotgun.com	10	Custom Design Seasonal Template	10
7	Leena	Phelps	8 William St.	Ithaca	NY	14850	11	11	lphelps@theydontunderstand.com	11	Revised Company logo	3
8	Korey	Larock	9855 Van Dyke Rd.	Sebastian	FL	32958	12	12	klarock@belink.com	12	Newspaper Advertisement	20
9	Vicki	Coons	8863 Trenton Road	Giffin	GA	30223	13	13	vcoons@visibleplane.com	13	Alternate Company logo	11
10	Cara	Pickford	7142 Mechanic Street	Goose Creek	SC	29445	17	17	cpickford@default.com	17	Custom Design Template	17

GraphicDesignService w/ Partial Client, Partial Address

This query results in client information associated with just IDs: client domain, phone number, services, and graphic design. Also associates balance, deposit, and paid amounts. Della can use this query to see who has to pay for their requested service by Della or who has paid in total for their service.

Cot-CIS3365-15.cougar.net - Remote Desktop Connection

inService with Partial Client and Partial Address.sql - COT-CIS3365-15.Demo_Database (COUGARNET\bharnold (77)) - Microsoft SQL Server Management Studio

```

SELECT sub4.ClientID, sub4.FirstName, sub4.LastName, StreetAddress, ZipCodeID, sub4.PhoneNumberID, sub4.ClientDomainID, sub4.email,
sub4.ServicesID, sub4.[Services_Description], sub4.GraphicDesignJobID,
sub4.Balance, sub4.Deposit, sub4.Paid, PaidToDate, sub4.Service_Date, sub4.TempClient FROM [Address]
INNER JOIN (
    SELECT Client.ClientID, FirstName, LastName, AddressID, PhoneNumberID, ClientDomainID, email, sub3.ServicesID, sub3.[Services_Description], sub3.GraphicDesignJobID,
    sub3.Balance, sub3.Deposit, Paid, PaidToDate, sub3.Service_Date, TempClient
    FROM Client
    INNER JOIN (
        SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.GraphicDesignJobID, sub2.Balance, sub2.Deposit, Paid, PaidToDate, sub2.Service_Date FROM Paid
        INNER JOIN (
            SELECT sub.ServicesID, sub.[Services_Description], sub.ClientID, sub.GraphicDesignJobID, Balance, Deposit, PaidID, Service_Date FROM Balance
            INNER JOIN (
                SELECT ServicesID, [Services_Description], ClientID, GraphicDesignJobID, BalanceID, PaidID, Service_Date
                FROM [Services] WHERE [Services] GraphicDesignJobID IS NOT NULL ) sub ON sub.BalanceID = [Balance].BalanceID ) sub2
            ON sub2.PaidID = Paid.PaidID ) sub3
        ON sub3.ClientID = Client.ClientID ) sub4
    ON sub4.AddressID = [Address].AddressID

```

Results

ClientID	FirstName	LastName	StreetAddress	ZipCodeID	PhoneNumberID	ClientDomainID	email	ServicesID	Services_Description	GraphicDesignJobID	Balance	Deposit	Paid
1	Timothy	Lowry	123 8th St.	32904	1	1	tlowry@website.com	1	Spring Sale Advertisement	1	600.00	120.00	100
2	Howard	Setzer	44 Shirley Ave.	60185	5	5	hsetzer@got.com	5	New Company logo	5	300.00	60.00	120
3	Norman	Janae	212 Randall Mill St.	60008	7	7	rjanas@exadlove.com	7	Website Banner	7	480.00	96.00	350
4	Karlyn	Mandeville	10 Indian Spring Rd.	30040	8	8	kmandeville@thelamofosho.com	8	Weekly Advertisement	12	330.00	66.00	200
5	Emilio	Alphonse	63 Wood Ave.	43068	9	9	ealphonese@anytime.com	9	Custom Website Image	2	450.00	90.00	0.01
6	Liliana	Widman	539 SE Longfellow Street	11432	10	10	lwidman@doublepumpshotgun.com	10	Custom Design Seasonal Template	10	360.00	72.00	251
7	Leena	Phelps	8 William St.	14850	11	11	lphelps@theydontunderstand.com	11	Revised Company logo	3	450.00	90.00	400
8	Korey	Larock	9855 Van Dyke Rd.	32958	12	12	klarock@atlinink.com	12	Newspaper Advertisement	20	630.00	126.00	200
9	Vicki	Coons	8863 Trenton Road	30223	13	13	vcoons@invisibleplane.com	13	Alternate Company logo	11	420.00	84.00	301
10	Cara	Pickford	7142 Mechanic Street	29445	17	17	cpickford@default.com	17	Custom Design Template	17	630.00	126.00	0.01

GraphicDesignService w/ Partial Client

Results in client first and last name associated with the IDs of everything else. Purpose of this query is provide less information and give Della only the information she needs.

The screenshot shows a Microsoft SQL Server Management Studio window titled "inService with Partial Client.sql - COT-CIS3365-15.Demo_Database (COUGARNET\bharnold (78)) - Microsoft SQL Server Management Studio". The query window contains a complex multi-table join query:

```

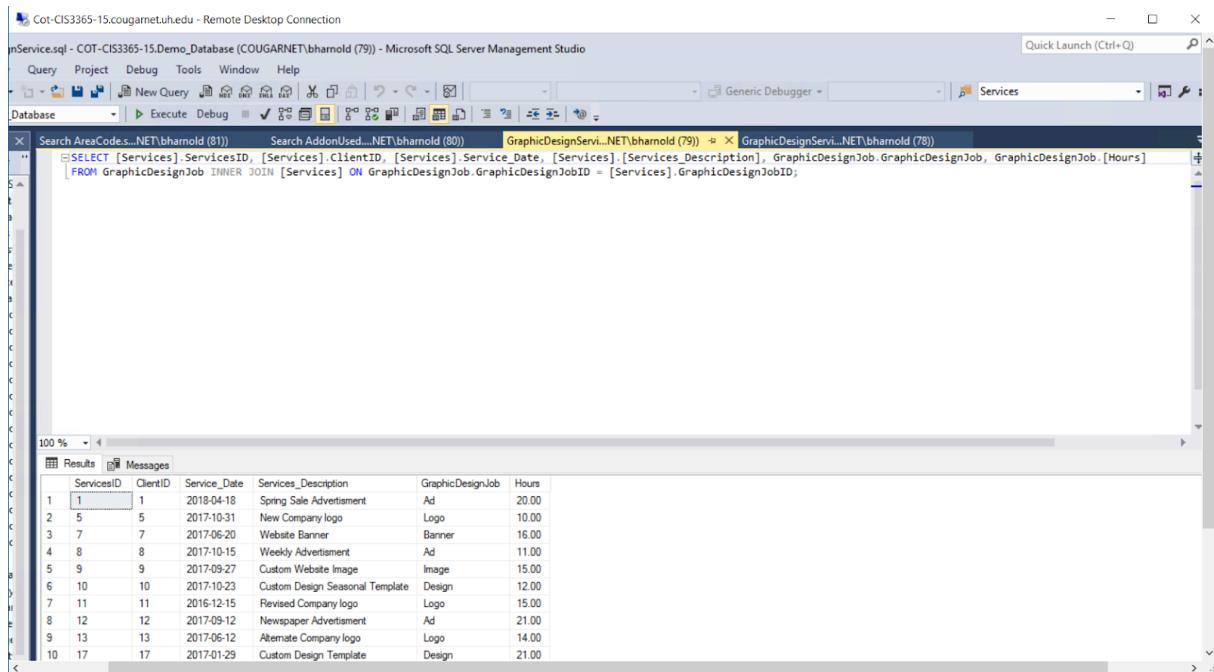
SELECT Client.ClientID, FirstName, LastName, AddressID, PhoneNumberID, ClientDomainID, email,
       sub3.ServicesID, sub3.[Services_Description], sub3.GraphicDesignJobID,
       sub3.Balance, sub3.Deposit, Paid, PaidToDate, sub3.Service_Date, TempClient
  FROM Client
 INNER JOIN (
    SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.GraphicDesignJobID, sub2.Balance, sub2.Deposit, Paid, PaidToDate, sub2.Service_Date
    FROM sub2
    INNER JOIN (
      SELECT sub.ServicesID, sub.[Services_Description], sub.ClientID, sub.GraphicDesignJobID, Balance, Deposit, PaidID, Service_Date
      FROM Balance
      INNER JOIN (
        SELECT ServicesID, [Services_Description], ClientID, GraphicDesignJobID, BalanceID, PaidID, Service_Date
        FROM [Services]
        WHERE [Services].GraphicDesignJobID IS NOT NULL
      ) sub ON sub.BalanceID = [Balance].BalanceID
    ) sub2
    ON sub2.PaidID = Paid.PaidID
  ) sub3
  ON sub3.ClientID = Client.ClientID;
  
```

The results grid below the query shows 10 rows of client data:

ClientID	FirstName	LastName	AddressID	PhoneNumberID	ClientDomainID	email	ServicesID	Services_Description	GraphicDesignJobID	Balance	Deposit	Paid	PaidToDate	Service
1	Timothy	Lowy	1	1	1	tlowy@website.com	1	Spring Sale Advertisement	1	600.00	120.00	100.00	2018-04-09	2018-04-09
2	Howard	Setzer	5	5	5	hsetzer@got.com	5	New Company logo	5	300.00	60.00	1200.00	2018-05-04	2017-05-04
3	Norman	Janas	7	7	7	njanas@texaslove.com	7	Website Banner	7	480.00	96.00	350.00	2018-04-14	2017-04-14
4	Karlyn	Mandeville	8	8	8	kmandeville@heatamofosho.com	8	Weekly Advertisement	12	330.00	66.00	200.00	2018-04-10	2017-04-10
5	Emilio	Alphonse	9	9	9	ealphonse@aryne.com	9	Custom Website Image	2	450.00	90.00	0.00	NULL	2017-04-01
6	Ulliana	Widman	10	10	10	lwidman@doublejumpshotgun.com	10	Custom Design Seasonal Template	10	360.00	72.00	25.00	2018-04-18	2017-04-18
7	Leena	Phelps	11	11	11	lphelps@theydontunderstand.com	11	Revised Company logo	3	450.00	90.00	400.00	2018-04-02	2016-04-02
8	Korey	Larock	12	12	12	klarock@batlink.com	12	Newspaper Advertisement	20	630.00	126.00	2000.00	2018-03-01	2017-03-01
9	Vicki	Coons	13	13	13	vcoons@invisibleplane.com	13	Alternate Company logo	11	420.00	84.00	30.00	2018-03-10	2017-03-10
10	Cara	Pickford	17	17	17	cpickford@default.com	17	Custom Design Template	17	630.00	126.00	0.00	NULL	2017-04-01

GraphicDesignService

Resulting query shows the service associated with the client based on graphic design job and how many hours it took to complete. Della can use this to calculate her price to charge her clients for the graphic design job.



The screenshot shows a Microsoft SQL Server Management Studio window titled "Cot-CIS3365-15.cougar.net.uh.edu - Remote Desktop Connection". The database is "COT-CIS3365-15.Demo_Database". The query window contains the following T-SQL code:

```
SELECT [Services].ServicesID, [Services].ClientID, [Services].Service_Date, [Services].[Services_Description], GraphicDesignJob.GraphicDesignJob, GraphicDesignJob.[Hours]
FROM GraphicDesignJob INNER JOIN [Services] ON GraphicDesignJob.GraphicDesignJobID = [Services].GraphicDesignJobID;
```

The results grid displays the following data:

	ServicesID	ClientID	Service_Date	Services_Description	GraphicDesignJob	Hours
1	1	1	2018-04-18	Spring Sale Advertisement	Ad	20.00
2	5	5	2017-10-31	New Company logo	Logo	10.00
3	7	7	2017-06-20	Website Banner	Banner	16.00
4	8	8	2017-10-15	Weekly Advertisment	Ad	11.00
5	9	9	2017-09-27	Custom Website Image	Image	15.00
6	10	10	2017-10-23	Custom Design Seasonal Template	Design	12.00
7	11	11	2016-12-15	Revised Company logo	Logo	15.00
8	12	12	2017-09-12	Newspaper Advertisment	Ad	21.00
9	13	13	2017-06-12	Alternate Company logo	Logo	14.00
10	17	17	2017-01-29	Custom Design Template	Design	21.00

Search AddonUsed

Query results in the service done with the specified addon and how many times the addon has been for the specific service. Della can use this query to show how many times she has used that addon feature.

The screenshot shows a Microsoft SQL Server Management Studio (SSMS) window. The title bar reads "Cot-CIS3365-15.cougar.net.uh.edu - Remote Desktop Connection" and "Used.sql - COT-CIS3365-15.Demo_Database (COUGARNET\bharnold (80)) - Microsoft SQL Server Management Studio". The main area displays a T-SQL query and its results.

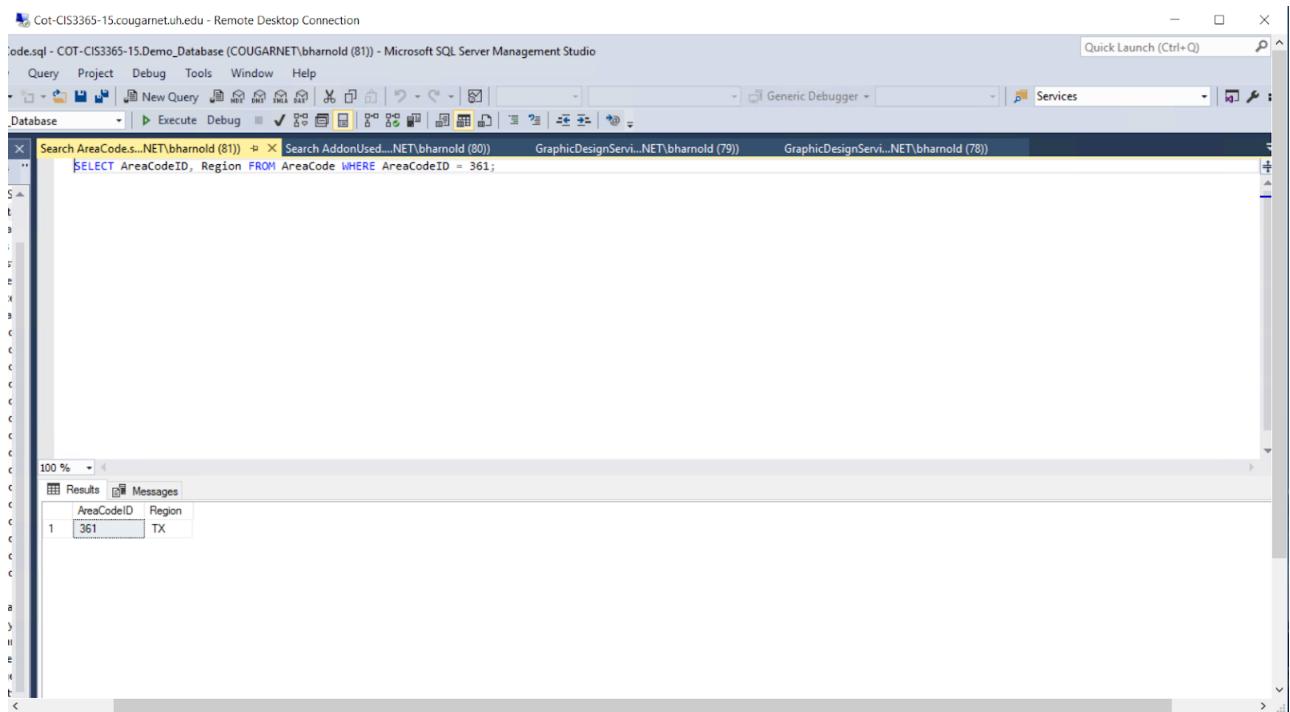
```
SELECT [Services].ServicesID, [Services].[Services_Description], Addons.Addon, Addons.Price, Service2Addon.Quantity
FROM Addons INNER JOIN ([Services] INNER JOIN Service2Addon ON [Services].ServicesID = Service2Addon.ServiceID)
ON Addons.AddonID = Service2Addon.AddonID
WHERE Addons.Addon = 'SEO'
ORDER BY ServicesID;
```

The results grid shows the following data:

	ServicesID	Services_Description	Addon	Price	Quantity
1	2	Wedding Planner Website	SEO	50.00	5
2	4	Online Support Forum	SEO	50.00	10
3	6	Antique Store Website	SEO	50.00	1
4	14	Auction Website	SEO	50.00	20
5	16	Arts and Crafts Online Store	SEO	50.00	30
6	20	Beauty Products Online Store	SEO	50.00	25

Search AreaCode

Query results in the search of which area code belongs to which state. Della can use this as partial information to figure out where her clients could potentially be from.



The screenshot shows a Microsoft SQL Server Management Studio (SSMS) window titled "Cot-CIS3365-15.cougarnet.uh.edu - Remote Desktop Connection". The database is "Demo.sql - COT-CIS3365-15.Demo_Database (COUGARNET\bharnold (81))". The query pane contains the following SQL code:

```
SELECT AreaCodeID, Region FROM AreaCode WHERE AreaCodeID = 361;
```

The results pane displays the following table:

AreaCodeID	Region
1	TX

Search Client

Results in the specific search for a client based on first name. Rather than going through a list, Della can use this query to filter clients based on a letter of the first name or first name in total.

The screenshot shows a Microsoft SQL Server Management Studio (SSMS) interface. The title bar reads "Cot-CIS3365-15.cougar.net.uh.edu - Remote Desktop Connection" and "sql - COT-CIS3365-15.Demo_Database (COUGARNET\bharnold (82)) - Microsoft SQL Server Management Studio". The main area displays a complex multi-table join query:

```

SELECT FirstName AS 'First Name', LastName AS 'Last Name', sub3.StreetAddress AS 'Street Address', sub3.City, sub3.[State], sub3.[Zip Code], CONCAT(sub.AreaCodeID, '-', Number) AS 'Phone Number'
INNER JOIN ( SELECT AreaCodeID, Region FROM AreaCode ) sub
ON sub.AreaCodeID = PhoneNumber.AreaCodeID
INNER JOIN (
    SELECT ClientID, FirstName, LastName, sub2.StreetAddress, sub2.City, sub2.[State], sub2.[Zip Code], PhoneNumberID, ClientDomainID, Email, TempClient FROM
    Client
    INNER JOIN (
        SELECT AddressID, StreetAddress, sub.City, sub.[State], sub.ZipCodeID AS 'Zip Code' FROM [Address]
        INNER JOIN (SELECT ZipCodeID, City, [State], County FROM ZipCode) sub
        ON sub.ZipCodeID = [Address].ZipCodeID ) sub2
        ON sub2.AddressID = Client.AddressID ) sub3
        ON sub3.ClientID = PhoneNumber.PhoneNumberID
WHERE FirstName LIKE 'V%'

```

The results pane shows one row of data:

First Name	Last Name	Street Address	City	State	Zip Code	Phone Number	Email	Temp Client
Vicki	Coons	8863 Trenton Road	Griffin	GA	30223	242-8920156	vcoons@invisibleplane.com	No

Search GraphicDesignService

This query filters the first and last name of the client and what graphic design service was completed for them. Also shows the balance, deposit, amount paid towards balance and dates of when payment was made and services done. Della can use this to find the information of a past service to a specific client.

Search GraphicDes...GARNET\akim (51)) -> X

```

SELECT CONCAT(sub8.FirstName, ' ', sub8.LastName) AS 'Name', CONCAT(sub8.StreetAddress, ' ', sub8.City, ' ', sub8.State), ' ', sub8.ZipCodeID) AS 'Address',
CONCAT('(', sub8.AreaCodeID, ')', ' ', Number) AS 'Phone Number', sub8.ClientDomain AS 'Client Domain',
sub8.email, sub8.[Services_Description] AS 'Service', GraphicDesignJob AS 'Job Description', sub8.Balance, sub8.Deposit, sub8.Paid,
PaidToDate, sub8.Service_Date, CASE WHEN sub8.TempClient = 1 THEN 'Yes' ELSE 'No' END AS 'Temp Client' FROM GraphicDesignJob
INNER JOIN (
    SELECT sub7.ClientID, sub7.FirstName, sub7.LastName, sub7.StreetAddress, sub7.City, sub7.[State], sub7.ZipCodeID, AreaCodeID, Number, ClientDomain.ClientDomain,
    sub7.email, sub7.ServicesID, sub7.[Services_Description], sub7.GraphicDesignJobID, sub7.Balance, sub7.Deposit, sub7.Paid,
    PaidToDate, sub7.Service_Date, sub7.TempClient FROM ClientDomain
) INNER JOIN (
    SELECT sub6.ClientID, sub6.FirstName, sub6.LastName, sub6.StreetAddress, sub6.City, sub6.[State], sub6.ZipCodeID, AreaCodeID, Number, sub6.ClientDomainID, sub6.email,
    sub6.ServicesID, sub6.[Services_Description], sub6.GraphicDesignJobID, sub6.Balance, sub6.Deposit, sub6.Paid,
    PaidToDate, sub6.Service_Date, sub6.TempClient FROM PhoneNumber
) INNER JOIN (
    SELECT sub5.ClientID, sub5.FirstName, sub5.LastName, sub5.StreetAddress, City, [State], ZipCode.ZipCodeID, sub5.PhoneNumberID, sub5.ClientDomainID, sub5.email,
    sub5.ServicesID, sub5.[Services_Description], sub5.GraphicDesignJobID, sub5.Balance, sub5.Deposit, sub5.Paid,
    PaidToDate, sub5.Service_Date, sub5.TempClient FROM ZipCode
) INNER JOIN (
    SELECT sub4.ClientID, sub4.FirstName, sub4.LastName, StreetAddress, ZipCodeID, sub4.PhoneNumberID, sub4.ClientDomainID, sub4.email,
    sub4.ServicesID, sub4.[Services_Description], sub4.GraphicDesignJobID,
    sub4.Balance, sub4.Deposit, sub4.Paid, PaidToDate, sub4.Service_Date, sub4.TempClient FROM [Address]
) INNER JOIN (
    SELECT Client.ClientID, FirstName, LastName, AddressID, PhoneNumberID, ClientDomainID, email, sub3.ServicesID, sub3.[Services_Description], sub3.GraphicDesignJobID,
    sub3.Balance, sub3.Deposit, Paid, PaidToDate, sub3.Service_Date, TempClient
    FROM Client
) INNER JOIN (
    SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.GraphicDesignJobID, sub2.Balance, sub2.Deposit, Paid, PaidToDate, sub2.Service_Date FROM Paid
) ON Client.ClientID = sub2.ClientID AND Client.ServicesID = sub2.ServicesID

```

100 % ↻

	Name	Address	Phone Number	Client Domain	email	Service	Job Description	Balance	Deposit	Paid	PaidToDate	Service_Date	Temp Client
1	Vicki Coons	8863 Trenton Road, Griffin, GA 30223	(242) 8920156	invisibleplane.com	vcoons@invisibleplane.com	Alternate Company logo	Logo	420.00	84.00	30.00	2018-03-10	2017-06-12	No

Query executed successfully. | COT-CIS3365-15 (14.0 RTM) | COUGARNET\akim (51) | Demo_Database | 00:00:00 | 1 rows

Search WebsiteDesignService

Same result as Search Graphic Design except filters for website design jobs to clients. Della would first determine the client she is looking for than results in the website design service she completed for that specific client.

```

Search WebsiteDes...GARNET\akim (54) - X
SELECT CONCAT(sub8.FirstName, ' ', sub8.LastName) AS 'Name', CONCAT(sub8.StreetAddress, ' ', sub8.City, ' ', sub8.[State], ' ', sub8.ZipCodeID) AS 'Address',
CONCAT('(', sub8.AreaCodeID, ')', ' ', Number) AS 'Phone Number', sub8.ClientDomain AS 'Client Domain'
sub8.email, sub8.[Services_Description] AS 'Service', WebsiteJob AS 'Job Description', sub8.Balance, sub8.Deposit, sub8.Paid,
PaidToDate, sub8.Service_Date, CASE WHEN sub8.TempClient = 1 THEN 'Yes' ELSE 'No' END AS 'Temp Client' FROM WebsiteJob
INNER JOIN (
SELECT sub7.ClientID, sub7.FirstName, sub7.LastName, sub7.StreetAddress, sub7.City, sub7.[State], sub7.ZipCodeID, AreaCodeID, Number, ClientDomain.ClientDomain,
sub7.email, sub7.ServicesID, sub7.[Services_Description], sub7.WebsiteJobID, sub7.Balance, sub7.Deposit, sub7.Paid,
PaidToDate, sub7.Service_Date, sub7.TempClient FROM ClientDomain
INNER JOIN (
SELECT sub6.ClientID, sub6.FirstName, sub6.LastName, sub6.StreetAddress, sub6.City, sub6.[State], sub6.ZipCodeID, AreaCodeID, Number, sub6.ClientDomainID, sub6.email,
sub6.ServicesID, sub6.[Services_Description], sub6.WebsiteJobID, sub6.Balance, sub6.Deposit, sub6.Paid,
PaidToDate, sub6.Service_Date, sub6.TempClient FROM PhoneNumber
INNER JOIN (
SELECT sub5.ClientID, sub5.FirstName, sub5.LastName, sub5.StreetAddress, City, [State], ZipCode.ZipCodeID, sub5.PhoneNumberID, sub5.ClientDomainID, sub5.email,
sub5.ServicesID, sub5.[Services_Description], sub5.WebsiteJobID, sub5.Balance, sub5.Deposit, sub5.Paid,
PaidToDate, sub5.Service_Date, sub5.Tempclient FROM ZipCode
INNER JOIN (
SELECT sub4.ClientID, sub4.FirstName, sub4.LastName, StreetAddress, ZipCodeID, sub4.PhoneNumberID, sub4.ClientDomainID, sub4.email,
sub4.ServicesID, sub4.[Services_Description], sub4.WebsiteJobID,
sub4.Balance, sub4.Deposit, sub4.Paid, PaidToDate, sub4.Service_Date, sub4.TempClient FROM [Address]
INNER JOIN (
SELECT Client.ClientID, FirstName, LastName, AddressID, PhoneNumberID, ClientDomainID, email, sub3.ServicesID, sub3.[Services_Description], sub3.WebsiteJobID,
sub3.Balance, sub3.Deposit, Paid, PaidToDate, sub3.Service_Date, TempClient
FROM Client
INNER JOIN (

```

100 % - 4

	Name	Address	Phone Number	Client Domain	email	Service	Job Description	Balance	Deposit	Paid	PaidToDate	Service_Date	Temp C
1	Mindi Karls	70 Bowman St., South Windsor, CT 06074	(281) 3308004	dubyastep.com	mkarls@dubyastep.com	Family Website	Small Personal (1-5 pages)	950.00	190.00	725.00	2018-04-06	2017-12-05	No
2	Edward Zoll	275 Arnold Court, Soddy Daisy, TN 37379	(410) 2214563	southlandpark.com	ezoll@southlandpark.com	Music Artist Website	Music Website	1350.00	270.00	250.00	2018-04-03	2017-07-27	No

Query executed successfully.

WebsiteService with Partial Client.sql

This query report shows the website service with the partial client. Since one of Della's Design requirements was to be able to pull up the client and be able to match it to the website service that the client has currently requested. This report will help Della's Design meet this requirement by being able to run a report that will pull up the data of the client and service by ID. It will also show the client first name, last name, and phone number. When it comes to the service, it will match it to the client ID and she will be able to see the service description and the balance that the client owes.

WebsiteService wit...GARNET\iakim (56) # X

```

SELECT Client.ClientID, FirstName, LastName, AddressID, PhoneNumberID, ClientDomainID, email, sub3.ServicesID, sub3.[Services_Description], sub3.Balance, sub3.Deposit, Paid, PaidToDate, sub3.Service_Date, TempClient
FROM Client
INNER JOIN (
    SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.WebsiteJobID, sub2.Balance, sub2.Deposit, Paid, PaidToDate, sub2.Service_Date
    FROM sub2
    INNER JOIN (
        SELECT sub.ServicesID, sub.[Services_Description], sub.ClientID, sub.WebsiteJobID, Balance, Deposit, PaidID, Service_Date
        FROM Balance
        INNER JOIN (
            SELECT ServicesID, [Services_Description], ClientID, WebsiteJobID, BalanceID, PaidID, Service_Date
            FROM [Services]
            WHERE [Services].WebsiteJobID IS NOT NULL
        ) sub
        ON sub2.PaidID = sub.PaidID
    ) sub2
    ON sub3.ClientID = sub2.ClientID;

```

100 % ↻

Results Messages

ClientID	FirstName	LastName	AddressID	PhoneNumberID	ClientDomainID	email	ServicesID	Services_Description	WebsiteJobID	Balance
1	2	Anna	Rhynes	2	2	arhynes@myspace.com	2	Wedding Planner Website	2	1575.00
2	3	Mindi	Karls	3	3	mkarls@dubyastep.com	3	Family Website	1	950.00
3	4	Kent	Steinfieldt	4	4	ksteinfieldt@howtomom.com	4	Online Support Forum	9	2550.00
4	6	Arlean	Bagley	6	6	abagley@gotapplies.com	6	Antique Store Website	7	2125.00
5	14	Masako	Shirah	14	14	mshirah@vaticanagain.com	14	Auction Website	16	3825.00
6	15	Edward	Zoll	15	15	ezoll@southlandpark.com	15	Music Artist Website	20	1350.00
7	16	Rosanne	Gremillion	16	16	rgremillion@pewds.io	16	Arts and Crafts Online Store	12	5825.00
8	18	Bruce	Dalke	18	18	bdalke@admin.com	18	Realtor Website	8	3125.00
9	20	Jorge	Barroso	20	20	jbarroso@astrosworldchamps.com	20	Beauty Products Online Store	11	4725.00

Query executed successfully. | COT-CIS3365-15 (14.0 RTM) | COUGARNET\iakim (56) | Demo_Database | 00:00:00 | 9 rows

WebsiteService with Partial Client and Partial Address.sql

This query report shows the website service with the partial client and the partial address. Della's Design likes to sometimes just search partial parts for the client and the address to make it a short list. In this report only the first name, last name, street address, and zip code are generated which gives her a shorter description for customer information which will fulfill this requirement that she has asked to complete.

The screenshot shows a SQL Server Management Studio window. The top pane displays a complex multi-table JOIN query. The bottom pane shows the results of the query execution, which returned 9 rows. The status bar at the bottom indicates the query was executed successfully.

```

SELECT sub4.ClientID, sub4.FirstName, sub4.LastName, StreetAddress, ZipCodeID, sub4.PhoneNumberID, sub4.ClientDomainID, sub4.email,
sub4.ServicesID, sub4.[Services_Description], sub4.WebsiteJobID,
sub4.Balance, sub4.Deposit, sub4.Paid, PaidToDate, sub4.Service_Date, sub4.TempClient FROM [Address]
INNER JOIN (
    SELECT Client.ClientID, FirstName, LastName, AddressID, PhoneNumberID, ClientDomainID, email, sub3.ServicesID, sub3.[Services_Description],
    sub3.Balance, sub3.Deposit, Paid, PaidToDate, sub3.Service_Date, TempClient
    FROM Client
    INNER JOIN (
        SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.WebsiteJobID, sub2.Balance, sub2.Deposit, Paid, PaidToDate, sub2.TempClient
        FROM sub2
        INNER JOIN (
            SELECT sub.ServicesID, sub.[Services_Description], sub.ClientID, sub.WebsiteJobID, Balance, Deposit, PaidID, Service_Date
            FROM Balance
            INNER JOIN (
                SELECT ServicesID, [Services_Description], ClientID, WebsiteJobID, BalanceID, PaidID, Service_Date
                FROM [Services]
                WHERE [Services].WebsiteJobID IS NOT NULL
            ) sub ON sub.BalanceID = [Balance].BalanceID
        ) sub2
        ON sub2.PaidID = Paid.PaidID
    ) sub3
    ON sub3.ClientID = Client.ClientID
) sub4
ON sub4.AddressID = [Address].AddressID

```

	ClientID	FirstName	LastName	StreetAddress	ZipCodeID	PhoneNumberID	ClientDomainID	email	ServicesID	Services_Description
1	2	Anna	Rhynes	71 Pilgrim Avenue	20815	2	2	arhynes@myspace.com	2	Wedding Planner Website
2	3	Mindi	Karla	70 Bowman St.	6074	3	3	mkarla@dubyastep.com	3	Family Website
3	4	Kent	Steinfeldt	4 Goldfield Rd.	96815	4	4	ksteinfeldt@howtomom.com	4	Online Support Forum
4	6	Arlean	Bagley	514 S. Magnolia St.	32806	6	6	abagley@gotapples.com	6	Antique Store Website
5	14	Masaoko	Shirah	473 Harvey Rd.	85302	14	14	mshirah@vaticanagain.com	14	Auction Website
6	15	Edward	Zoll	275 Arnold Court	37379	15	15	ezoll@southlandpark.com	15	Music Artist Website
7	16	Rosanne	Gremillion	770 S. Willow Avenue	32566	16	16	rgremillion@pewds.io	16	Arts and Crafts Online Store
8	18	Bruce	Dalke	77 S. Prospect Dr.	60515	18	18	bdalke@admin.com	18	Realtor Website
9	20	Jorge	Barroso	1 North Second Drive	3301	20	20	jbarroso@astrosworldchamps.com	20	Beauty Products Online Store

Query executed successfully. COT-CIS3365-15 (14.0 RTM) COUGARNET\iakim (56) | Demo_Database | 00:00:00 | 9 rows

WebsiteServiceWithAddon – Quantity Calculated.sql

This query report shows the Website Service with add on with the quantity calculated. Della's designed currently writes people add on requests on paper and the calculated the total user a calculator and writes it back in which is time consuming and is not able to tell customer their total right away. This report will help her get rid of that issue by allowing her to run a report of the addons and it will give her to total of certain addons that have been added by customers.

WebsiteServiceWithAddon – Totaled.sql

```

SELECT [Services].ServicesID, [Services].[Services_Description] AS 'Service', Addons.Addon, (Addons.Price * Service2Addon.Quantity) AS 'Addon Total'
FROM Addons INNER JOIN ([Services] INNER JOIN Service2Addon ON [Services].ServicesID = Service2Addon.ServiceID)
ON Addons.AddonID = Service2Addon.AddonID ORDER BY ServicesID;

```

100 %

ServicesID	Service	Addon	Addon Total	
1	2	Wedding Planner Website	Unique Theme	250.00
2	2	Wedding Planner Website	Complete New Content Package	500.00
3	2	Wedding Planner Website	SEO	250.00
4	2	Wedding Planner Website	Custom Slideshow	75.00
5	3	Family Website	Content Editing	50.00
6	3	Family Website	Unique Theme	250.00
7	3	Family Website	Customized Font	200.00
8	3	Family Website	Copy Text	150.00
9	4	Online Support Forum	SEO	500.00
10	4	Online Support Forum	Accessibility Features	50.00
11	4	Online Support Forum	Security Features	200.00
12	4	Online Support Forum	Fully Responsive CSS	300.00
13	4	Online Support Forum	Unique Theme	250.00
14	6	Antivirus Software	SEO	50.00

Query executed successfully.

COT-CIS3365-15 (14.0 RTM) | COUGARNET\jakim (56) | Demo_Database | 00:00:00 | 53 rows

WebsiteServiceWithAddon – Totaled.sql

This query report shows the Website Service with add on with the quantity calculated. Della's designed currently writes people add on requests on paper and the calculated the total user a calculator and writes it back in which is time consuming and is not able to tell customer their total right away. This report will help her get rid of that issue but allowing her to run a report of the addons and it will not only give her total of the add ons that have been added by the customer but it will also give her a total of the add ons.

```

SELECT [Services].ServicesID, [Services].[Services_Description] AS 'Service', SUM(Addons.Price * Service2Addon.Quantity) AS 'AddonTotal'
FROM Addons INNER JOIN ([Services] INNER JOIN Service2Addon ON [Services].ServicesID = Service2Addon.ServiceID)
ON Addons.AddonID = Service2Addon.AddonID
GROUP BY [Services].ServicesID, [Services].[Services_Description]
ORDER BY ServicesID;

```

The screenshot shows the execution of the provided SQL query. The results are displayed in a table with three columns: ServicesID, Service, and AddonTotal. The data is as follows:

ServicesID	Service	AddonTotal
1	2 Wedding Planner Website	1075.00
2	3 Family Website	650.00
3	4 Online Support Forum	1300.00
4	6 Antique Store Website	1125.00
5	14 Auction Website	2325.00
6	15 Music Artist Website	850.00
7	16 Arts and Crafts Online Store	3825.00
8	18 Realtor Website	1625.00
9	20 Beauty Products Online Store	3225.00

At the bottom of the results window, a message indicates "Query executed successfully." and provides session details: COT-CIS3365-15 (14.0 RTM), COUGARNET\iakim (56), Demo_Database, 00:00:00, 9 rows.

WebsiteServiceWithAddon.sql

This query report takes the addon price with the quantity of addons that a client desires and multiplies it, it is then stored in “AddonTotal”. When this report is ran it will output the Service ID, name of the service, and it will give the total of the addon. Della’s Design currently does not have a way to keep track of addons that are added to services in an organized way that will help her keep track and give an accurate total to the client. This report will satisfy her requirement to be able to keep track of what addons are being added to which outstanding orders and the prices to be able to invoice the client.

WebsiteServiceWithPartialClientAndCompleteAddress.sql

```

SELECT [Services].ServicesID, [Services].[Services_Description], Addons.Addon, Addons.Price, Service2Addon.Quantity
FROM Addons INNER JOIN ([Services] INNER JOIN Service2Addon ON [Services].ServicesID = Service2Addon.ServiceID)
ON Addons.AddonID = Service2Addon.AddonID ORDER BY ServicesID;

```

Results

ServicesID	Services_Description	Addon	Price	Quantity
1	2	Wedding Planner Website	Unique Theme	250.00
2	2	Wedding Planner Website	Complete New Content Package	100.00
3	2	Wedding Planner Website	SEO	50.00
4	2	Wedding Planner Website	Custom Slideshow	75.00
5	3	Family Website	Content Editing	50.00
6	3	Family Website	Unique Theme	250.00
7	3	Family Website	Customized Font	200.00
8	3	Family Website	Copy Text	50.00
9	4	Online Support Forum	SEO	50.00
10	4	Online Support Forum	Accessibility Features	50.00
11	4	Online Support Forum	Security Features	200.00
12	4	Online Support Forum	Fully Responsive CSS	300.00
13	4	Online Support Forum	Unique Theme	250.00
14	6	Antivirus Software	SEO	50.00

Query executed successfully.

WebsiteService with Partial Client and Complete Address.sql

This query reports represents the website service with the partial client and the complete address for that specific client. Currently Della's Design keeps all her customer data in paper which are stored in file cabinets and every time she wants to access this information she has to go through tons of papers to get to the specific client that she is looking for. This report will help Della's design be able to pull a report of all the current clients that services are being provided to and she will be able to tell the clients first name, last name, and address as well as the email address.

WebsiteService wit...GARNET\iakim (59)) ➔ X

```

SELECT sub5.ClientID, sub5.FirstName, sub5.LastName, sub5.StreetAddress, City, [State], ZipCode.ZipCodeID, sub5.PhoneNumberID, sub5.ClientDomainID, sub5.ServicesID, sub5.[Services_Description], sub5.WebsiteJobID, sub5.Balance, sub5.Deposit, sub5.Paid, PaidToDate, sub5.Service_Date, sub5.TempClient FROM ZipCode
INNER JOIN (
    SELECT sub4.ClientID, sub4.FirstName, sub4.LastName, StreetAddress, ZipCodeID, sub4.PhoneNumberID, sub4.ClientDomainID, sub4.email, sub4.ServicesID, sub4.[Services_Description], sub4.WebsiteJobID, sub4.Balance, sub4.Deposit, sub4.Paid, PaidToDate, sub4.Service_Date, sub4.TempClient FROM [Address]
) INNER JOIN (
    SELECT Client.ClientID, FirstName, LastName, AddressID, PhoneNumberID, ClientDomainID, email, sub3.ServicesID, sub3.[Services_Description], sub3.Balance, sub3.Deposit, Paid, PaidToDate, sub3.Service_Date, TempClient
) FROM Client
INNER JOIN (
    SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.WebsiteJobID, sub2.Balance, sub2.Deposit, Paid, PaidToDate, sub2.TempClient
) ON sub2.ServicesID = sub.[Services_ID]
INNER JOIN (
    SELECT sub.ServicesID, sub.[Services_Description], sub.ClientID, sub.WebsiteJobID, Balance, Deposit, PaidID, Service_Date
) FROM Balance
INNER JOIN (
    SELECT ServicesID, [Services_Description], ClientID, WebsiteJobID, BalanceID, PaidID, Service_Date
) FROM [Services] WHERE [Services].WebsiteJobID IS NOT NULL
) sub ON sub.BalanceID = [Balance].BalanceID
ON sub2.PaidID = Paid.PaidID
ON sub3.ClientID = Client.ClientID
ON sub.ClientID = Client.ClientID

```

100 % ➔ 4

	ClientID	FirstName	LastName	StreetAddress	City	State	ZipCodeID	PhoneNumberID	ClientDomainID	email	ServicesID	Service
1	2	Anna	Rhynes	71 Pilgrim Avenue	Chevy Chase	MD	20815	2	2	arynnes@myspace.com	2	Wedd
2	3	Mindi	Karla	70 Bowman St.	South Windsor	CT	6074	3	3	mkarla@dubyastep.com	3	Family
3	4	Kent	Steinfieldt	4 Goldfield Rd.	Honolulu	HI	96815	4	4	ksteinfieldt@howtomom.com	4	Online
4	6	Adean	Bagley	514 S. Magnolia St.	Orlando	FL	32806	6	6	abagley@gotapples.com	6	Antiqu
5	14	Masaoko	Shirah	473 Harvey Rd.	Glendale	AZ	85302	14	14	mshirah@vaticanagain.com	14	Audio
6	15	Edward	Zoll	275 Arnold Court	Soddy Daisy	TN	37379	15	15	ezoll@southlandpark.com	15	Music
7	16	Rosanne	Gremillion	770 S. Willow Avenue	Navare	FL	32566	16	16	rgremillion@pewds.io	16	Arts ar
8	18	Bruce	Dalke	77 S. Prospect Dr.	Downers Grove	IL	60515	18	18	bdalke@admin.com	18	Realte
9	20	Jorge	Barroso	1 North Second Drive	Concord	NH	3301	20	20	jbarroso@astrosworldchamps.com	20	Beaut

Query executed successfully. | COT-CIS3365-15 (14.0 RTM) | COUGARNET\iakim (59) | Demo_Database | 00:00:00 | 9 rows

WebsiteService with Partial Client and Complete Address and Complete Phone Number.sql

This report shows the website service with the partial client and complete address and complete phone number. Della's Design needs an easier way to look up current client's complete address and complete phone numbers in case she needs to contact them for any questions or issues with their current orders. This report will help Della's Design be able to easily get a list of the clients that are currently receiving services and see the phone number with the area code and the email address and contact them as soon as possible.

WebsiteService wit...GARNET\iakim (59) # X

```

FROM [Services] WHERE [Services].WebsiteJobID IS NOT NULL ) sub ON sub.BalanceID = [Balance].BalanceID ) sub2
ON sub2.PaidID = Paid.PaidID ) sub3
ON sub3.ClientID = Client.ClientID ) sub4
ON sub4.AddressID = [Address].AddressID ) sub5
ON sub5.ZipCodeID = ZipCode.ZipCodeID ) sub6
ON sub6.PhoneNumberID = Phonenumber.PhoneNumberID ) sub7
ON sub7.ClientDomainID = ClientDomain.ClientDomainID

```

100 % 4

Results Messages

ClientID	FirstName	LastName	StreetAddress	City	State	ZipCodeID	AreaCodeID	Number	ClientDomain	email	Service
1	2	Anna Rhynes	71 Pilgrim Avenue	Chevy Chase	MD	20815	713	6905413	myspace.com	arhynes@myspace.com	2
2	3	Mindi Karls	70 Bowman St.	South Windsor	CT	6074	281	3308004	dubyastep.com	mkarls@dubyastep.com	3
3	4	Kent Steinfield	4 Goldfield Rd.	Honolulu	HI	96815	209	4531289	howtomon.com	ksteinfield@howtomon.com	4
4	6	Arlan Bagley	514 S. Magnolia St.	Orlando	FL	32806	503	1973482	gotapples.com	abagley@gotapples.com	6
5	14	Masako Shirah	473 Harvey Rd.	Glendale	AZ	85302	630	7841354	vaticanagain.com	mshirah@vaticanagain.com	14
6	15	Edward Zoll	275 Arnold Court	Soddy Daisy	TN	37379	410	2214563	southlandpark.com	ezoll@southlandpark.com	15
7	16	Rosanne Gremillion	770 S. Willow Avenue	Navarre	FL	32566	610	5569520	pewds.io	rgremillion@pewds.io	16
8	18	Bruce Dalke	77 S. Prospect Dr.	Downers Grove	IL	60515	416	8742351	admin.com	bdalke@admin.com	18
9	20	Jorge Barroso	1 North Second Drive	Concord	NH	3301	225	3146795	strosworldchamps.com	jbarroso@astrosworldchamps.com	20

Query executed successfully. COT-CIS3365-15 (14.0 RTM) COUGARNET\iakim (59) Demo_Database 00:00:00 | 9 rows

WebsiteService with Partial Client and Complete Address and Complete Phone Number and Domain.sql

This query reports shows the website service with partial client, complete address, and complete phone number and domain. According to the requirements that Della's design has, she sometimes like to search clients email by domain to see how many she has. This reports will allow her to see the on what client domain clients have. This will be useful to her because she mostly emails users regarding questions and concerns with the services, this report will be really useful to her needs because can just run this report and be able to get the email address for whatever client she is needing to contact.

```

WebsiteService wit...GARNETjakiim (83) # ×
SELECT sub7.ClientID, sub7.FirstName, sub7.LastName, sub7.StreetAddress, sub7.City, sub7.[State], sub7.ZipCodeID, AreaCodeID, Number, ClientDomain, email, sub7.ServicesID, sub7.[Services_Description], sub7.WebsiteJobID, sub7.Balance, sub7.Deposit, sub7.Paid, PaidToDate, sub7.Service_Date, sub7.TempClient FROM ClientDomain
INNER JOIN (
    SELECT sub6.ClientID, sub6.FirstName, sub6.LastName, sub6.StreetAddress, sub6.City, sub6.[State], sub6.ZipCodeID, AreaCodeID, Number, ClientDomain, email, sub6.ServicesID, sub6.[Services_Description], sub6.WebsiteJobID, sub6.Balance, sub6.Deposit, sub6.Paid, PaidToDate, sub6.Service_Date, sub6.TempClient FROM PhoneNumber
INNER JOIN (
        SELECT sub5.ClientID, sub5.FirstName, sub5.LastName, sub5.StreetAddress, City, [State], ZipCode.ZipCodeID, sub5.PhoneNumberID, sub5.ClientDomainID, sub5.ServicesID, sub5.[Services_Description], sub5.WebsiteJobID, sub5.Balance, sub5.Deposit, sub5.Paid, PaidToDate, sub5.Service_Date, sub5.TempClient FROM ZipCode
INNER JOIN (
            SELECT sub4.ClientID, sub4.FirstName, sub4.LastName, StreetAddress, ZipCodeID, sub4.PhoneNumberID, sub4.ClientDomainID, sub4.email, sub4.ServicesID, sub4.[Services_Description], sub4.WebsiteJobID, sub4.Balance, sub4.Deposit, sub4.Paid, PaidToDate, sub4.Service_Date, sub4.TempClient FROM [Address]
INNER JOIN (
                SELECT Client.ClientID, FirstName, LastName, AddressID, PhoneNumberID, ClientDomainID, email, sub3.ServicesID, sub3.[Services_Description], sub3.Balance, sub3.Deposit, Paid, PaidToDate, sub3.Service_Date, TempClient
                FROM Client
            )NINNER JOINN (

```

100 % ▾

Results Messages

ClientID	FirstName	LastName	StreetAddress	City	State	ZipCodeID	AreaCodeID	Number	ClientDomain	email	Service
1	2	Anna	Rhynes	71 Pilgrim Avenue	Chevy Chase	MD	20815	713	6905413	myspace.com	2
2	3	Mindi	Karla	70 Bowman St.	South Windsor	CT	6074	281	3308004	dubyastep.com	3
3	4	Kent	Steinfeldt	4 Goldfield Rd.	Honolulu	HI	96815	209	4531289	howtomom.com	4
4	6	Arlan	Bagley	514 S. Magnolia St.	Orlando	FL	32806	503	1973482	gotapples.com	6
5	14	Masako	Shirah	473 Harvey Rd.	Glendale	AZ	85302	630	7841354	vaticanagain.com	14
6	15	Edward	Zoll	275 Arnold Court	Soddy Daisy	TN	37379	410	2214563	southlandpark.com	15
7	16	Rosanne	Gremillion	770 S. Willow Avenue	Navare	FL	32566	610	5569520	pewds.io	16
8	18	Bruce	Dalke	77 S. Prospect Dr.	Downers Grove	IL	60515	416	8742351	admin.com	18
9	20	Jorge	Barroso	1 North Second Drive	Concord	NH	3301	225	3146795	strosworldchamps.com	20

Query executed successfully. | COT-CIS3365-15 (14.0 RTM) | COUGARNETjakiim (83) | Demo_Database | 00:00:00 | 9 rows

WebsiteService with Balance and Paid.sql

This query report shows the website service with the balance and what has currently been paid to Della's Design. Currently Della's design uses an excel spreadsheet to keep her client account balance and what currently has been paid, but it is kind of hard to keep switching between spreadsheets in order to get to a certain client. Therefore, we have created this report which she can just run and automatically see the service ID and the description which will match it to the client ID and will give her a balance of how much the current client owes and how much the client has paid up to date. This will provide her a more concrete way to keep track of her financials so that clients can pay her.

WebsiteService wit...GARNET\iakim (59) ▾ X

```

SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.WebsiteJobID, sub2.Balance, sub2.Deposit, Paid, PaidToDate, sub2.Service_Date
INNER JOIN (
    SELECT sub.ServicesID, sub.[Services_Description], sub.ClientID, sub.WebsiteJobID, Balance, Deposit, PaidID, Service_Date FROM Balance
) sub ON sub.BalanceID = [Balance].BalanceID
INNER JOIN (
    SELECT ServicesID, [Services_Description], ClientID, WebsiteJobID, BalanceID, PaidID, Service_Date
    FROM [Services] WHERE [Services].WebsiteJobID IS NOT NULL
) sub ON sub2.PaidID = Paid.PaidID

```

100 % ▾

Results Messages

ServicesID	Services_Description	ClientID	WebsiteJobID	Balance	Deposit	Paid	PaidToDate	Service_Date
1	2 Wedding Planner Website	2	2	1575.00	315.00	600.00	2018-04-17	2018-04-06
2	3 Family Website	3	1	950.00	190.00	725.00	2018-04-06	2017-12-05
3	4 Online Support Forum	4	9	2550.00	510.00	0.00	NULL	2017-10-09
4	6 Antique Store Website	6	7	2125.00	425.00	75.00	2018-04-05	2017-07-17
5	14 Auction Website	14	16	3825.00	765.00	800.00	2018-03-14	2017-02-13
6	15 Music Artist Website	15	20	1350.00	270.00	250.00	2018-04-03	2017-07-27
7	16 Arts and Crafts Online Store	16	12	5825.00	1165.00	150.00	2018-04-12	2017-02-20
8	18 Realtor Website	18	8	3125.00	625.00	300.00	2018-04-15	2017-02-14
9	20 Beauty Products Online Store	20	11	4725.00	945.00	1750.00	2018-04-15	2018-03-14

Query executed successfully. | COT-CIS3365-15 (14.0 RTM) | COUGARNET\iakim (59) | Demo_Database | 00:00:00 | 9 rows

WebsiteService DONE.sql

This query report shows the services that have been completed for the client. Currently Della's Design does not have a way to pull a report so she can see what services have been completed, instead she has to go through a lot of paper in order to see what has been completed, which is not the most efficient way since it is not time efficient. This report will allow her to see which client services have been finished, this way she knows what she does not have to worry about anymore, this report will save her time with customer and employees as she can easily see what has been done and what has not been completed so far.

WebsiteService DO...GARNET\jakim (59) ➔ X

```

sub4.Balance, sub4.Deposit, sub4.Paid, PaidToDate, sub4.Service_Date, sub4.TempClient FROM [Address]
INNER JOIN (
SELECT Client.ClientID, FirstName, LastName, AddressID, PhoneNumberID, ClientDomainID, email, sub3.ServicesID, sub3.[Services_Description]
sub3.Balance, sub3.Deposit, Paid, PaidToDate, sub3.Service_Date, TempClient
FROM Client
INNER JOIN (
SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.WebsiteJobID, sub2.Balance, sub2.Deposit, Paid, PaidToDate, sub2.Service_Date
FROM sub2.Servi
SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.WebsiteJobID, Balance, Deposit, PaidID, Service_Date FROM Balance
INNER JOIN (
SELECT ServicesID, [Services_Description], ClientID, WebsiteJobID, BalanceID, PaidID, Service_Date
FROM [Services] WHERE [Services].WebsiteJobID IS NOT NULL ) sub ON sub.BalanceID = [Balance].BalanceID ) sub2
ON sub2.PaidID = Paid.PaidID ) sub3
ON sub3.ClientID = Client.ClientID ) sub4
ON sub4.AddressID = [Address].AddressID ) sub5
ON sub5.ZipCodeID = ZipCode.ZipCodeID ) sub6
ON sub6.PhoneNumberID = Phonenumber.PhoneNumberID ) sub7
ON sub7.ClientDomainID = ClientDomain.ClientDomainID ) sub8
ON sub8.WebsiteJobID = WebsiteJob.WebsiteJobID

```

100 % ➔

Results Messages

	Name	Address	Phone Number	Client Domain	email	Service	Job Description
1	Anna Rhynes	71 Pilgrim Avenue, Chevy Chase, MD 20815	(713) 6905413	myspace.com	arhynes@myspace.com	Wedding Planner Website	Medium Persona
2	Mindi Karls	70 Bowman St., South Windsor, CT 6074	(281) 3308004	dubyastep.com	mkarls@dubyastep.com	Family Website	Small Personal (
3	Kent Steinfield	4 Goldfield Rd., Honolulu, HI 96815	(209) 4531289	howtomom.com	ksteinfield@howtomom.com	Online Support Forum	Forum Website
4	Arlene Bagley	514 S. Magnolia St., Orlando, FL 32806	(503) 1973482	gotapples.com	abagley@gotapples.com	Antique Store Website	Medium Business
5	Masako Shirah	473 Harvey Rd., Glendale, AZ 85302	(630) 7841354	vaticanagain.com	mshirah@vaticanagain.com	Auction Website	Auction Website
6	Edward Zoll	275 Arnold Court, Soddy Daisy, TN 37379	(410) 2214563	southlandpark.com	ezoll@southlandpark.com	Music Artist Website	Music Website
7	Rosanne Gremillion	770 S. Willow Avenue, Navarre, FL 32566	(610) 5569520	pewds.io	rgremillion@pewds.io	Arts and Crafts Online Store	Large Online Sto
8	Bruce Dalke	77 S. Prospect Dr., Downers Grove, IL 60515	(416) 8742351	admin.com	bdalke@admin.com	Realtor Website	Large Business
9	Jorge Barroso	1 North Second Drive, Concord, NH 3301	(225) 3146795	strosworldchamps.com	jbarroso@astrosworldchamps.com	Beauty Products Online Store	Medium Online S

Query executed successfully. | COT-CIS3365-15 (14.0 RTM) | COUGARNET\jakim (59) | Demo_Database | 00:00:00 | 9 rows

WebsiteService Beginnings.sql

This query reports shows the data in which the service was first open. It also gives the clients first name, last name, service description, website job ID, balance ID, paid ID and of course the service date for when the service was first requested. The current system that Della Design currently has does not give her a way to be able to pull a report of when the service started so she is not able to see how much it is taking. This report will allow her to see the service date of the service for when it was entered into the system, which will give her an idea of how long, is taking to complete the service, which she can then show to the client in case any client has questions.

WebsiteService Be...GARNET\iakim (59) ×

```

SELECT FirstName, LastName, sub.Services_Description, sub.WebsiteJobID, sub.BalanceID, sub.PaidID, sub.Service_Date FROM Client
INNER JOIN (
    SELECT ServicesID, [Services_Description], ClientID, WebsiteJobID, BalanceID, PaidID, Service_Date
    FROM [Services] WHERE [Services].WebsiteJobID IS NOT NULL
)
    sub ON sub.ClientID = Client.ClientID

```

100 %

	FirstName	LastName	Services_Description	WebsiteJobID	BalanceID	PaidID	Service_Date
1	Anna	Rhynes	Wedding Planner Website	2	2	2	2018-04-06
2	Mindi	Karls	Family Website	1	3	3	2017-12-05
3	Kent	Steinfieldt	Online Support Forum	9	4	4	2017-10-09
4	Arlan	Bagley	Antique Store Website	7	6	6	2017-07-17
5	Masako	Shirah	Auction Website	16	14	14	2017-02-13
6	Edward	Zoll	Music Artist Website	20	15	15	2017-07-27
7	Rosanne	Gremillion	Arts and Crafts Online Store	12	16	16	2017-02-20
8	Bruce	Dalke	Realtor Website	8	18	18	2017-02-14
9	Jorge	Barroso	Beauty Products Online Store	11	20	20	2018-03-14

Query executed successfully. | COT-CIS3365-15 (14.0 RTM) | COUGARNET\iakim (59) | Demo_Database | 00:00:00 | 9 rows

WebsiteJobServices.sql

This query report outputs the Website Job Services that Della's design currently offers to customers. Even though Della's design has a system to see, what services are offered she wanted to do it digitally. Therefore, this report will allow her to see how many services are currently offered by Della's Design by outputting the services ID, client ID, service description, website job, and the price. This report also helps her see what she currently has whenever she wants to add more.

WebsiteJobService...GARNET\iakim (59) ↗ X

```
SELECT [Services].ServicesID, [Services].ClientID, [Services].Service_Date, [Services].[Services_Description], WebsiteJob.WebsiteJob, We
```

100 % ▶

	Results	Messages				
	ServicesID	ClientID	Service_Date	Services_Description	WebsiteJob	Price
1	2	2	2018-04-06	Wedding Planner Website	Medium Personal (6-10 pages)	500.00
2	3	3	2017-12-05	Family Website	Small Personal (1-5 pages)	300.00
3	4	4	2017-10-09	Online Support Forum	Forum Website	1250.00
4	6	6	2017-07-17	Antique Store Website	Medium Business	1000.00
5	14	14	2017-02-13	Auction Website	Auction Website	1500.00
6	15	15	2017-07-27	Music Artist Website	Music Website	500.00
7	16	16	2017-02-20	Arts and Crafts Online Store	Large Online Store	2000.00
8	18	18	2017-02-14	Realtor Website	Large Business	1500.00
9	20	20	2018-03-14	Beauty Products Online Store	Medium Online Store	1500.00

Query executed successfully. | COT-CIS3365-15 (14.0 RTM) | COUGARNET\iakim (59) | Demo_Database | 00:00:00 | 9 rows

ServiceWithAddons.sql

This query report outputs the current service ID and what addons can be added, as well the price and the quantity. Della's Design can use this report at any time that either an employee or a customer has questions of what addons can be added to the services and their price. This way she does not have to only have paper prints of it but as well a digital way in her computer which she can just run and show the customer on the spot.

ServiceWithAddons...ARNET\iakim (56) # X

```
SELECT Service2Addon.ServiceID, Addons.Addon, Addons.Price, Service2Addon.Quantity
FROM Addons INNER JOIN Service2Addon ON Addons.AddonID = Service2Addon.AddonID;
```

100 % < >

	ServiceID	Addon	Price	Quantity
1	2	Unique Theme	250.00	1
2	2	Complete New Content Package	100.00	5
3	2	SEO	50.00	5
4	2	Custom Slideshow	75.00	1
5	3	Content Editing	50.00	1
6	3	Unique Theme	250.00	1
7	3	Customized Font	200.00	1
8	3	Copy Text	50.00	3
9	4	SEO	50.00	10
10	4	Accessibility Features	50.00	1
11	4	Security Features	200.00	1
12	4	Fully Responsive CSS	300.00	1
13	4	Unique Theme	250.00	1
14	6	SEO	50.00	1
15	6	Copy Text	50.00	5
16	6	Security Features	200.00	1
17	6	Fully Responsive CSS	300.00	1
18	6	Facebook Plugin	50.00	1
19	6	PayPal API Plugin	150.00	1
20	6	Shopping Cart Function	125.00	1
21	14	SEO	50.00	20
22	14	Copy Text	50.00	10
23	14	Security Features	200.00	1
24	14	Fully Responsive CSS	300.00	1
25	14	Facebook Plugin	50.00	1
26	14	PayPal API Plugin	150.00	1
27	14	Shopping Cart Function	125.00	1

Query executed successfully.

COT-CIS3365-15 (14.0 RTM) | COUGARNET\iakim (56) | Demo_Database | 00:00:00 | 53 rows

Search ZipCode.sql

This report shows a query that can search for a specific zip code. Currently Della's design does not have a system to easily look up a zip code for specific customer address. This report will help her easily search the city, state, and county for a specific zip code, which will save her the time of having to search through her paper documentation.

The screenshot shows a SQL query window in SQL Server Management Studio. The query is:

```
SELECT ZipCodeID, City, [State], County FROM ZipCode WHERE ZipCodeID = 78413;
```

The results pane displays a table with the following data:

	ZipCodeID	City	State	County
1	78413	Corpus Christi	TX	Nueces

At the bottom of the window, a status bar indicates:

Query executed successfully. | COT-CIS3365-15 (14.0 RTM) | COUGARNET\iakim (56) | Demo_Database | 00:00:00 | 1 rows

Function - FindJob

This query uses a function that request a specific job that shows when the service was done, description, job type, price, and clientID. Della can use this query to search for when she did a specific job and make the connection to whom she did it for.

The screenshot shows a SQL Server Management Studio (SSMS) interface. At the top, there's a title bar for 'Function - FindJob...GARNET\jakim (57)'. Below it is a code editor containing a T-SQL query:

```
SELECT [Services].ServicesID, [Services].ClientID, [Services].Service_Date, [Services].[Services_Description],  
WebsiteJob.WebsiteJob, WebsiteJob.Price  
FROM WebsiteJob INNER JOIN [Services] ON WebsiteJob.WebsiteJobID = [Services].WebsiteJobID  
WHERE WebsiteJob.WebsiteJobID = dbo.FindJob('Auction Website');
```

Below the code editor is a results grid titled 'Results' with one row of data:

	ServicesID	ClientID	Service_Date	Services_Description	WebsiteJob	Price
1	14	14	2017-02-13	Auction Website	Auction Website	1500.00

At the bottom of the results grid, a status bar displays: 'Query executed successfully.' and 'COT-CIS3365-15 (14.0 RTM) | COUGARNET\jakim (57) | Demo_Database | 00:00:00 | 1 rows'.

Function - FindName

This function calls the first and last name from the client table with other client information. Della can use this to quickly search a certain client than scrolling through a list of clients.

Function - FindName

```

SELECT
    FirstName AS 'First Name',
    LastName AS 'Last Name',
    StreetAddress AS 'Street Address',
    City,
    [State],
    ZipCode.ZipCodeID,
    CONCAT('(', AreaCode.AreaCodeID, ')', ' ', Number) AS 'Phone Number',
    Email, CASE WHEN TempClient = 1 THEN 'Yes' ELSE 'No' END AS 'Temp Client' FROM PhoneNumber
INNER JOIN AreaCode ON AreaCode.AreaCodeID = PhoneNumber.AreaCodeID
INNER JOIN Client ON Client.ClientID = PhoneNumber.PhoneNumberID
INNER JOIN [Address] ON [Address].AddressID = Client.AddressID
INNER JOIN ZipCode ON ZipCode.ZipCodeID = [Address].ZipCodeID
WHERE Client.ClientID = dbo.FindName('Vicki', 'Coons');

```

Results

	First Name	Last Name	Street Address	City	State	ZipCodeID	Phone Number	Email	Temp Client
1	Vicki	Coons	8863 Trenton Road	Griffin	GA	30223	(242) 8920156	vcoons@invisibleplane.com	No

Query executed successfully.

Function - FindRequirement

This function associates requirements with either graphic design or website design job to the clientID and service description. As shown below, the requirement “new logo” associates with graphic design job to the clientID 11 and the description of that service. Dellas can use this function to filter services done by the certain requirement of a specific project.

The screenshot shows a SQL Server Management Studio window. The query pane contains the following T-SQL code:

```
Function - FindRe...GARNET\iakim (57) ↗ X
SELECT [Services].Services_Description, [Services].ClientID, [Services].WebsiteJobID, [Services].GraphicDesignJobID, Requirements FROM [Services]
INNER JOIN Requirements ON [Services].ServicesID = Requirements.ServicesID
WHERE Requirements.RequirementsID = dbo.FindRequirement('New Logo')
```

The results pane displays a single row of data:

	Services_Description	ClientID	WebsiteJobID	GraphicDesignJobID	Requirements
1	Revised Company logo	11	NULL	3	New Logo

At the bottom of the results pane, a message indicates the query was executed successfully.

Function - FindService

This function uses service description to list all services done under that specific description. Also list the addon, price, and how many times that service description was done. Della can use this function to filter certain descriptions of a service.

Function - FindService

```

SELECT [Services].ServicesID, [Services].[Services_Description], Addons.Addon, Addons.Price, Service2Addon.Quantity
FROM Addons INNER JOIN ([Services] INNER JOIN Service2Addon ON [Services].ServicesID = Service2Addon.ServiceID)
ON Addons.AddonID = Service2Addon.AddonID
WHERE [Services].ServicesID = dbo.FindService('Family Website') ORDER BY ServicesID;

```

100 %

Results Messages

ServicesID	Services_Description	Addon	Price	Quantity	
1	3	Family Website	Content Editing	50.00	1
2	3	Family Website	Unique Theme	250.00	1
3	3	Family Website	Customized Font	200.00	1
4	3	Family Website	Copy Text	50.00	3

Query executed successfully.

COT-CIS3365-15 (14.0 RTM) | COUGARNET\jakim (57) | Demo_Database | 00:00:00 | 4 rows

Function - SearchEmail

This function filters domain search by username (email) to pull up their domain and password. Della can use this function to find the domain and password of a client through their email as the email title is their username.

```

Function - SearchEmail.GARNET\jakim (60) ✎ X
SELECT
ClientDomain,
Username,
[Password]
FROM ClientDomain
INNER JOIN
WebmasterLogin ON WebmasterLogin.WebmasterLoginID = ClientDomain.WebmasterLoginID
WHERE ClientDomain.ClientDomainID = dbo.SearchEmail('tlowry@website.com')

```

Results Messages

ClientDomain	Username	Password
website.com	tlowry	password

Query executed successfully. COT-CIS3365-15 (14.0 RTM) COUGARNET\jakim (60) Demo_Database 00:00:00 | 1 rows

GraphicDesignService w/ Partial Client, Complete Address, Complete Phone Number

This query takes the client information and associates it with services, their domainID, graphic design job id, balance, deposit, paid, and dates tables. This simplifies the previous query by simply applying a filter to less information and associating more ID's.

```

GraphicDesignService with Partial Client and Complete Address and Complete Phone Number.sql - COT-CIS3365-15.Demo_Database (COUGARNET\jakim (63)) - Microsoft SQL Server Management Studio
File Edit View Query Project Debug Tools Window Help
File Edit View Query Project Debug Tools Window Help
GraphicDesignService.GARNET\jakim (62) GraphicDesignService.GARNET\jakim (61) GraphicDesignService.GARNET\jakim (57)
SELECT sub6.ClientID, sub6.FirstName, sub6.LastName, sub6.StreetAddress, sub6.City, sub6.[State], sub6.ZipCodeID, AreaCodeID, Number, sub6.ClientDomainID, sub6.email, PaidToDate, sub6.Service_Date, sub6.TempClient FROM PhoneNumber
INNER JOIN (
    SELECT sub5.ClientID, sub5.FirstName, sub5.LastName, sub5.StreetAddress, City, [State], ZipCode.ZipCodeID, sub5.PhoneNumberID, sub5.ClientDomainID, sub5.email, sub5.ServicesID, sub5.[Services_Description], sub5.GraphicDesignJobID, sub5.Balance, sub5.Deposit, sub5.Paid, PaidToDate, sub5.Service_Date, sub5.TempClient FROM ZipCode
    INNER JOIN (
        SELECT sub4.ClientID, sub4.FirstName, sub4.LastName, StreetAddress, ZipCodeID, sub4.PhoneNumberID, sub4.ClientDomainID, sub4.email, sub4.ServicesID, sub4.[Services_Description], sub4.GraphicDesignJobID, sub4.Balance, sub4.Deposit, sub4.Paid, PaidToDate, sub4.Service_Date, sub4.TempClient FROM [Address]
        INNER JOIN (
            SELECT Client.ClientID, FirstName, LastName, AddressID, PhoneNumberID, ClientDomainID, email, sub3.ServicesID, sub3.[Services_Description], sub3.GraphicDesignJobID, sub3.Balance, sub3.Deposit, Paid, PaidToDate, sub3.Service_Date, TempClient
            FROM Client
            INNER JOIN (
                SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.GraphicDesignJobID, sub2.Balance, sub2.Deposit, Paid, PaidToDate, sub2.Service_Date FROM Paid
                INNER JOIN (
                    SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.GraphicDesignJobID, Balance, Deposit, PaidID, Service_Date FROM Balance
                    INNER JOIN (
                        SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.GraphicDesignJobID, Balance, Deposit, PaidID, Service_Date FROM Balance
                        INNER JOIN (
                            SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.GraphicDesignJobID, Balance, Deposit, PaidID, Service_Date FROM Balance
                            INNER JOIN (
                                SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.GraphicDesignJobID, Balance, Deposit, PaidID, Service_Date FROM Balance
                                INNER JOIN (
                                    SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.GraphicDesignJobID, Balance, Deposit, PaidID, Service_Date FROM Balance
                                    INNER JOIN (
                                        SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.GraphicDesignJobID, Balance, Deposit, PaidID, Service_Date FROM Balance
                                        INNER JOIN (
                                            SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.GraphicDesignJobID, Balance, Deposit, PaidID, Service_Date FROM Balance
                                            INNER JOIN (
                                                SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.GraphicDesignJobID, Balance, Deposit, PaidID, Service_Date FROM Balance
                                                INNER JOIN (
                                                    SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.GraphicDesignJobID, Balance, Deposit, PaidID, Service_Date FROM Balance
                                                    INNER JOIN (
                                                        SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.GraphicDesignJobID, Balance, Deposit, PaidID, Service_Date FROM Balance
                                                        INNER JOIN (
                                                            SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.GraphicDesignJobID, Balance, Deposit, PaidID, Service_Date FROM Balance
                                                            INNER JOIN (
                                                                SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.GraphicDesignJobID, Balance, Deposit, PaidID, Service_Date FROM Balance
                                                                INNER JOIN (
                                                                    SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.GraphicDesignJobID, Balance, Deposit, PaidID, Service_Date FROM Balance
                                                                    INNER JOIN (
                                                                        SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.GraphicDesignJobID, Balance, Deposit, PaidID, Service_Date FROM Balance
                                                                        INNER JOIN (
                                                                            SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.GraphicDesignJobID, Balance, Deposit, PaidID, Service_Date FROM Balance
                                                                            INNER JOIN (
                                                                                SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.GraphicDesignJobID, Balance, Deposit, PaidID, Service_Date FROM Balance
                                                                                INNER JOIN (
                                                                                    SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.GraphicDesignJobID, Balance, Deposit, PaidID, Service_Date FROM Balance
                                                                                    INNER JOIN (
                                                                                        SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.GraphicDesignJobID, Balance, Deposit, PaidID, Service_Date FROM Balance
                                                                                        INNER JOIN (
                                                                                            SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.GraphicDesignJobID, Balance, Deposit, PaidID, Service_Date FROM Balance
                                                                                            INNER JOIN (
                                                                                                SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.GraphicDesignJobID, Balance, Deposit, PaidID, Service_Date FROM Balance
                                                                                                INNER JOIN (
                                                                                                    SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.GraphicDesignJobID, Balance, Deposit, PaidID, Service_Date FROM Balance
                                                                                                    INNER JOIN (
                                                                                                        SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.GraphicDesignJobID, Balance, Deposit, PaidID, Service_Date FROM Balance
................................................................

```

Results Messages

ClientID	FirstName	LastName	StreetAddress	City	State	ZipCodeID	AreaCodeID	Number	ClientDomainID	email	ServicesID	Services_Description	GraphicDesignJobID	Balance	Deposit
1	Timothy	Lowry	123 8th St.	Melbourne	FL	32904	361	9999999	1	tlowry@website.com	1	Spring Sale Advertisement	1	600.00	120.00
2	Howard	Setzer	44 Shirley Ave.	West Chicago	IL	60185	417	7413658	5	hezar@got.com	5	New Company logo	5	300.00	60.00
3	Norman	Janas	212 Randall Mill St.	Polling Meadows	IL	60008	202	1234567	7	njanas@texasslove.com	7	Website Banner	7	480.00	96.00
4	Katlyn	Mandeville	10 Indian Spring Rd.	Cumming	GA	30040	254	3571988	8	kmmandeville@theleadmosho.com	8	Weedy Advertisement	12	330.00	66.00
5	Emilio	Alphonse	63 Wood Ave	Reynoldsburg	OH	43068	330	4826716	9	ealphonse@ryme.com	9	Custom Website Image	2	450.00	90.00
6	Liliana	Widman	539 SE Longfellow Street	Jamaica	NY	11432	401	9102973	10	lwidman@doubleumpumpshotgun.com	10	Custom Design Seasonal Template	10	360.00	72.00
7	Leena	Phelps	8 Willam St.	Ithaca	NY	14850	270	5643212	11	lphelps@hydronunderstand.com	11	Revised Company logo	3	450.00	90.00
8	Korey	Larock	9855 Van Dyke Rd.	Sebastian	FL	32958	501	4683218	12	klarock@btalink.com	12	Newspaper Advertisement	20	630.00	126.00
9	Vicki	Coons	8863 Trenton Road	Giffin	GA	30223	242	8920156	13	vcoons@visibleplane.com	13	Alternate Company logo	11	420.00	84.00
10	Cara	Pickford	7142 Mechanic Street	Goose Creek	SC	29445	351	5023070	17	cpickford@default.com	17	Custom Design Template	17	630.00	126.00
11	Max	Waters	227 Broad Street	Velico	FL	33594	479	8885892	19	mwaters@sperad.com	19	Revised Website image	9	150.00	30.00

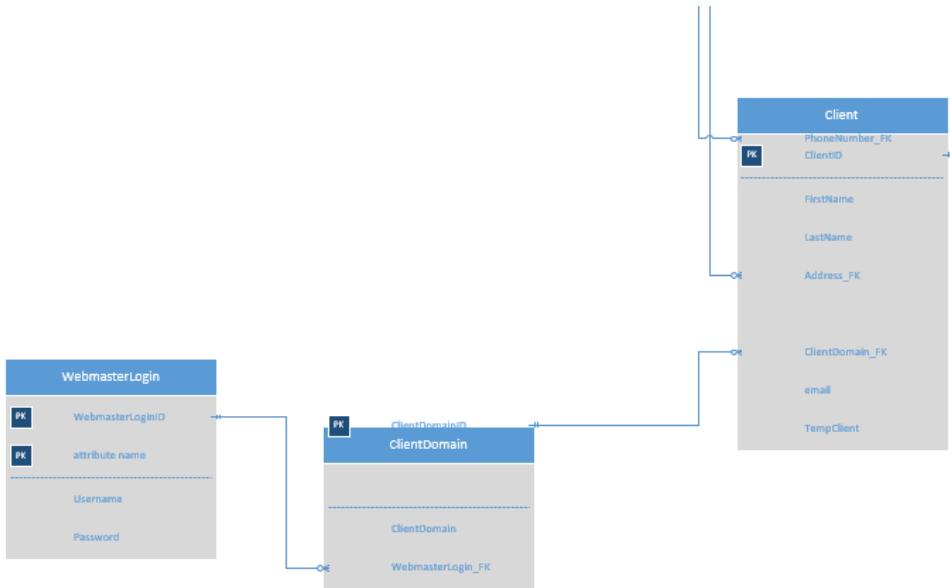
Query executed successfully. COT-CIS3365-15 (14.0 RTM) COUGARNET\jakim (63) Demo_Database 00:00:00 | 11 rows

Key decisions

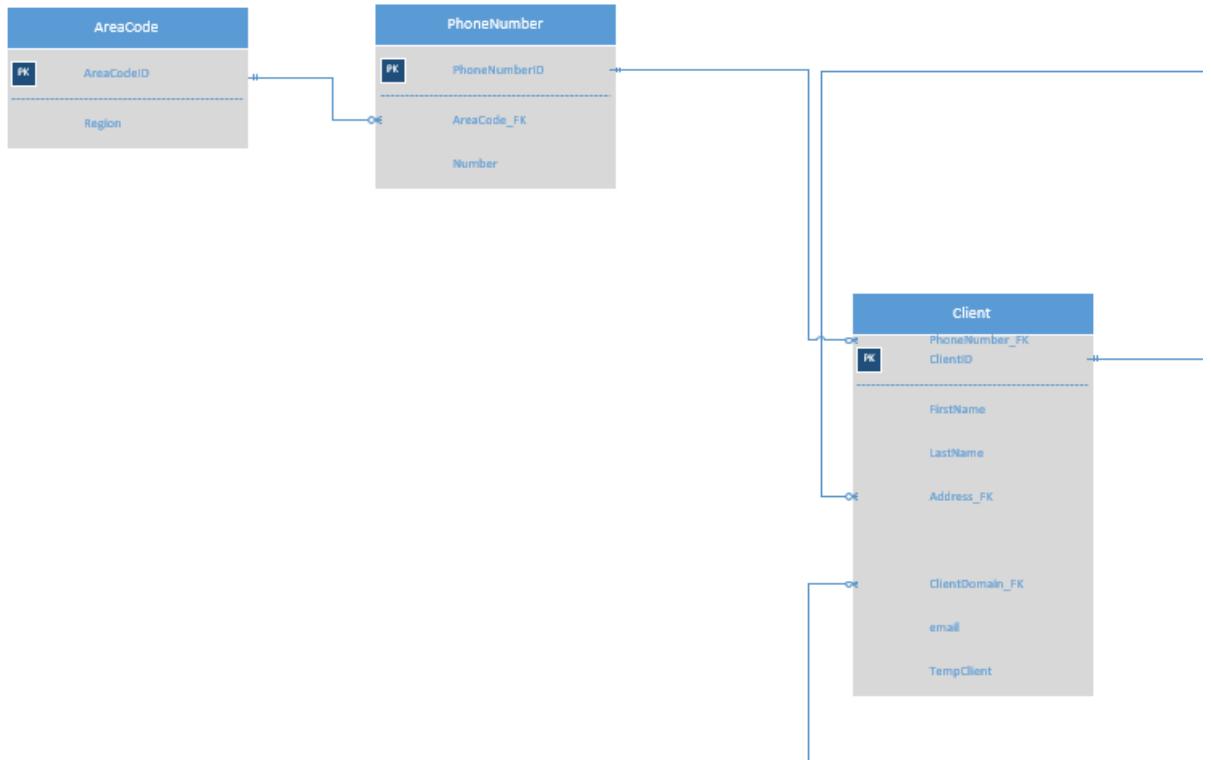
While developing the database, we decided to come up with three tables that would make things easier for Della. First, we decided to create a TempClient entity that would store client info who have inquired about Della's services. If no response is made from them within a week, their information will be deleted automatically. However, if a client decides to further pursue Della services, the information stored in TempClient will be saved for future reference. This will help differentiate interest from customers.

Second, we added a Balance table that would record any unpaid balance (deposit) from Della's customers and be removed when paid. When the balance is paid, the information goes to PaidToDate table which records when payments were made to Della. In order for Della to track cost, we implemented a Requirements table that holds information regarding programs used to track cost of those programs as well as added expenses.

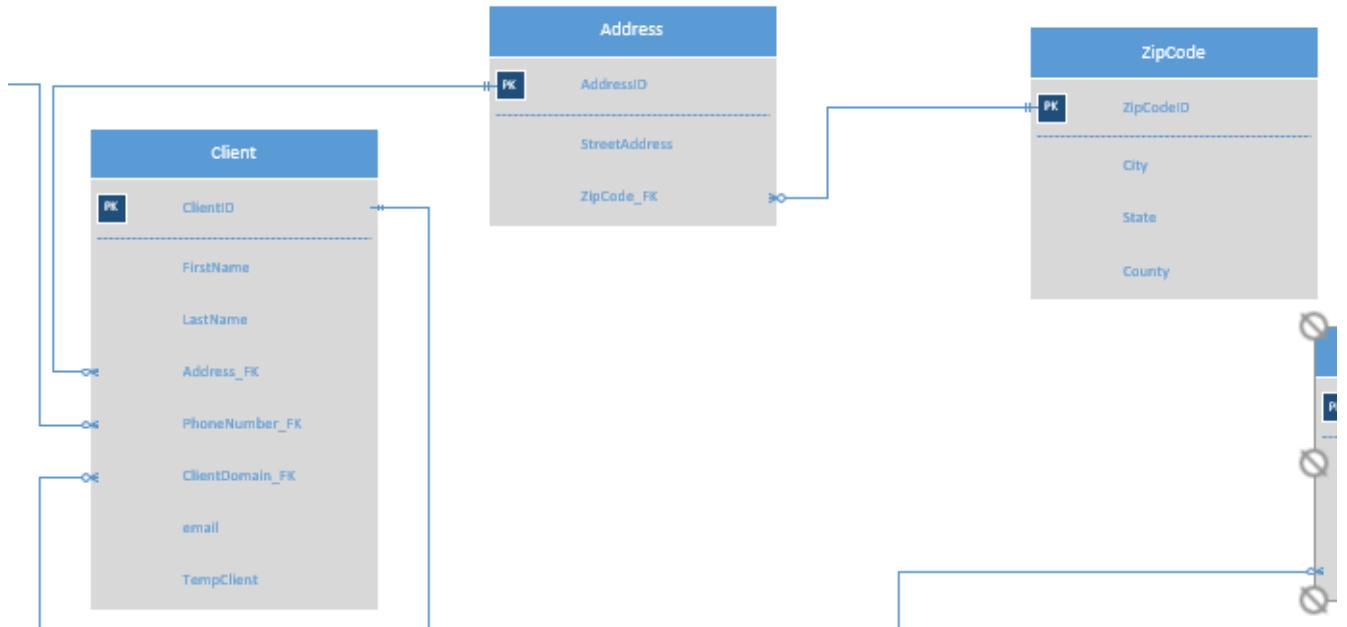
Lastly, we created a Service table which records description of service, service date, Clientid, WebsiteJobid, and GraphicDesignJobid to track past services. Additionally , the WebmasterLogin associates the Client to a username and password.



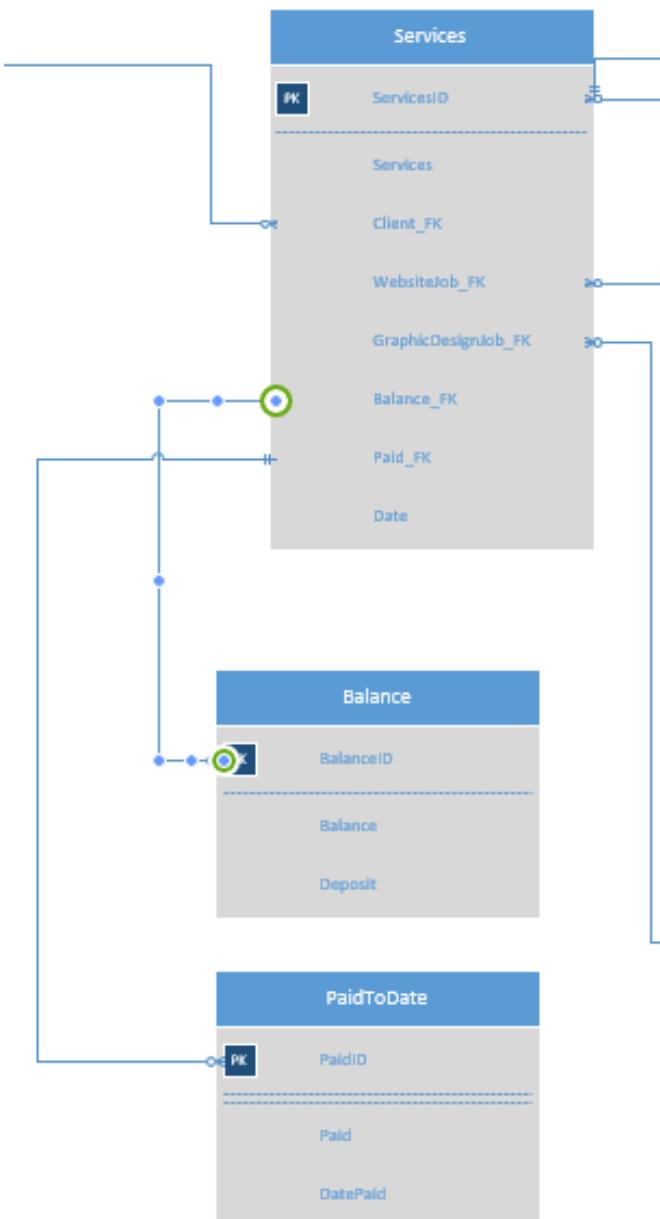
The relationship between WebMasterLogin and ClientDomain is to associate the login credentials with the specified domain associated with the specific client.



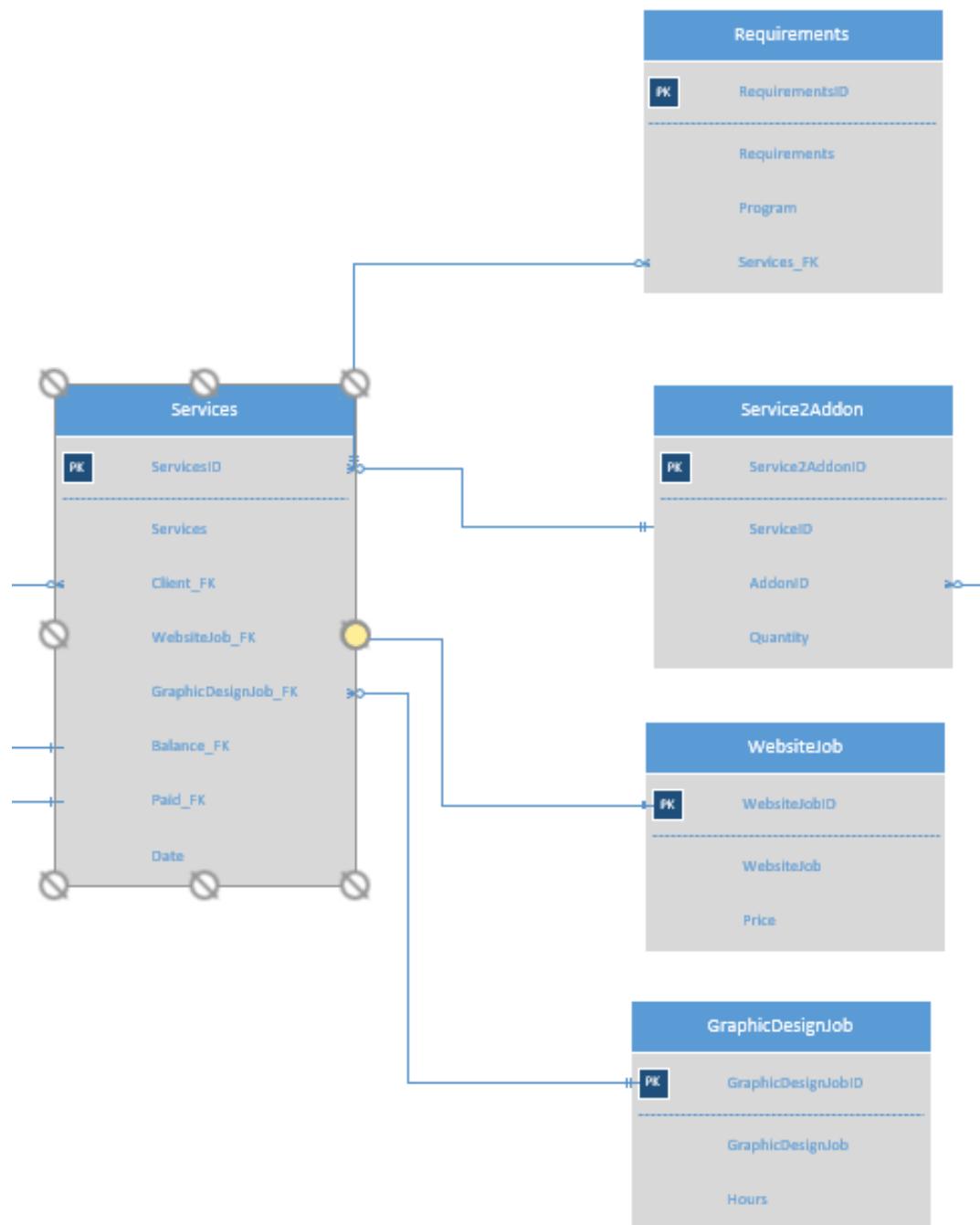
The relationship between client and phone number is provide a contact point for Della to client and the area code is used specify a state from where that phone number is from.



The relationship between client and address and zip code is to associate the client with specified location. Within the client table we added a TempClient which adds clients interested in Della's Design Solution for service but has not fully committed.



The relationship between services and balance what is to be paid to Della for the specific service. PaidToDate specifies how much has been paid from balance to Della on a service.



Relation of services to addon, website job, and graphic job are to determine whether it will be set price for the the service (website job) or an hourly service (graphic design job). This also connects back to client by referencing client information to the type of service they requested.

Testing Process

The testing phase requires multiple areas that need to be tested in order to check for errors and to verify that the highest level of data integrity is achieved. In order to do this, well-structured guidelines need be followed during testing. As a group, we have taken the proper guidelines to verify that the database that we have created is accurate and will work along our Java application to meet our client requirements.

Our database and Java application has been tested through a local SQL server. Our testing phase consisted of two stages, the standalone database and the Java application with the database connection. In order to achieve more accurate results, we decided to test the database first and then the database with the Java application to ensure that it was communicating.

The first phase of testing was our database. The first step that we took to test our database was ensuring that all the tables have been created exactly as it was on the ERD that we created. After we were confident that all the tables were created with the correct attributes, we proceeded to ensure that the data was uploading correctly to the tables with the correct format. After testing that the database connection was successful, we created bulk scripts to load the data from CSV files into the server database. After successfully uploading the data once and we verified that it was accurate and had no nulls, we proceeded to the next step, which was to confirm that our bulk scripts were working correctly. To test our bulk scripts we decided to delete the database and start from the beginning by using the scripts that we had created. When we uploaded the data for a second time, we ensured that it there were no bugs. There were times that as we loaded the data, we realized that our data needed formatting, in this cases we took the opportunity to correct the data and re-uploaded and ensure that it was set to the way we desired. As we compiled our scripts multiple times in the server, we also created different databases within the server to verify the performance of all of our queries and functions.

After running, multiple test we came to a point that we had ensure that our database was running smoothly and we were ready to proceed with our Java application.

The second phase consisted of testing our Java application; we started by testing the connection of our database with the GUI. The first step to enable our Java application to communicate with our database was to open a port in the firewall on the server. After we had a successful connection, we tested that our forms were working by entering data into the forms and verifying that the correct data in the correct format was being input into the database and into the correct tables. After verifying that we were successful in using the forms in the Java GUI to enter data into the database through our Java application we proceeded to test our reports. We ran multiple reports to ensure that the output was correct and it matched the client desired. After we finished testing the database and the Java applications and we were, confident with the results we presented our final product to the client to obtain feedback.

In conclusion, the testing process was long but when we presented the database and Java application to the sponsors, they were impressed with the work that had been done to improve their system. We had some of the users in the shop tested our application and the feedback it was provided was all-positive. As a team, we were successful in the testing stage of our system we verified that every single part of the system was working as expected by our client.

Project Improvements

The possible improvements made after the completion of our database has been prioritized based on what functions could have benefitted the client the most. The priority of further improvements would be to additional changes to any data formatting and parameters within stored procedures. The GUI can be modified to improve its functionality by updating to a more recent release. There can be a clear field button feature added to a later release of the database to allow for a quicker way for the user to clear fields. After the essential features have been added to the database, a lower priority

such as an alert system could be added which would allow the users to be notified of any changes and requirements that may be present. The lowest priority improvement to be designed would be a mobile application for the users to access from their phones.

Any existing data that needs to have its data type modified for better functionality would be a priority for improvement for verification. Any modified parameters within stored procedures that are would need to be verified to check that the statements do not conflict with the database. The necessary updates to the general GUI are essential to the database functionality to the user.

Any new GUI improvements and modifications are not as essential to the database's function, but can improve the user's overall experience using the database can help the users to become more proficient with their database. There are low-priority improvements that are not required such as adding a clear button to save the users' time when clearing multiple fields of data. An alert system for the users for various notifications would improve the communication between the users and the database for any changes to the project to be relayed to the associated user.

The development of a mobile application is a lower priority because the main database needs to be functional and fine-tuned before starting development for another version.

These improvements follow a priority according to what the client and group have agreed upon for the database. Our client and group as for improving the overall functionality of the database with these features have prioritized the group and user hold meetings for discussions to describe.

Priority

1. Data Formatting
2. Parameters within stored procedures
3. General GUI updates
4. GUI modifications
5. Clear button
6. Alert system
7. Mobile application

Project Database Maintenance Issues

With the implementation of the database system for Della's Design Solutions, there is a need to update the functionality of the data in the system while maintaining the integrity of the database. The following will keep the system integrity and ensure that the database has the ability to have data cleanup, backups, restores, and be able to retrieve data. The database should be able to create, alter, bulk insert, and query.

The data within the SQL will need to be tested regularly to verify its integrity and functionality. The relationships will need to be regularly validated and have the connections between the database and GUI verified to ensure that the functionality is correct. The queries will need to be tested and resolved of any errors then rewritten and reloaded. Any unexpected null values will need to be corrected and have the appropriate values input. Any changes should be rewritten and reloaded into the database then have the data reformatted to ensure that the referential integrity is still maintained, and additionally make sure that the database is still functional and connected to the GUI. Referential integrity should be also be checked.

In addition to the testing steps, the maintenance process consists of backing up data, checking the database integrity, and reducing the size of the database when possible. There will be a regular backups of the database during the regular maintenance with the option to restore any backups selected. During maintenance, the reduction of database files and logs where applicable will streamline the application and improve the speed of the database.

The maintenance process requires the regular testing of various features of the database and the maintenance steps to ensure that the database is still functional and its lifespan prolonged for future use.

List of testing problems:

1. Storage space Issues - Oracle files keep growing as usage grows, so you need to monitor physical space as the database files to care if space is available or not.
2. Performance issues - degrading is also a major issue, reasons; the more the clients uses the application you have to keep getting feedback from users and updating your application sometimes to meet up with the varying demands.

- 3. Cost of Maintenance - the recurring cost of maintaining the database
- 5. Need for frequent updates - to maintain the security and functionality
- 6. Network outages - When a system is down, database performance obviously is at its worst. Outages can be caused by database issues such as running out of storage space due to increasing data volumes .

List of testing steps:

- 1. Test data within SQL
- 2. Validate relationships
- 3. Correct functionality
- 4. Check connection between database and GUI

- 5. Testing the data
- 6. Resolving errors
- 7. Using waves to load data
- 8. Testing queries
- 9. Unexpected null values
- 10. Rewrite and reload
- 11. Testing the GUI
- 12. Check the connection
- 13. Reformat the data

- 14. Compiled the create, alter, bulk insert scripts, and queries
- 15. Tested for referential integrity
- 16. Revised code and data
- 17. Test again with full data uploaded
- 18. Connect SQL Server to Java GUI

List of maintenance steps:

1. We perform a full backup of the system on an automated schedule.
2. check database integrity by using the created documentation on how to restore the database back to the original state.
3. Shrink database to reduces the size of the database's data and log files in order to increase efficiency of the database

Lessons Learned

Not Everything is Possible

Since the early stages of our project, many changes were made to the overall project. There were many features that we thought were a possibility to complete for the client but after the analysis of the time constraints and the requirements of the work that it would take to complete; we needed to adjust our project. Our group realized that everything we had planned to do in the beginning not going to be possible to complete. The lesson learned from this adjustment is to look at things in a realistic way and expect to be not able to promise things that are not possible to complete.

Time restraint

A timeline is important for every project that is established to make sure that the project is scheduled for completion in the expected time frame that is agreed with the client. However, this is not always the case, as things can change at any stage of the project. As a team we got too comfortable that we could finish everything since we were ahead of everything which lead to procrastination in many areas of the project. As a result, we ended up doing last minute modification to the documentation right before the project was due to ensure that all the documents were formatted and completed to turn in. However, this semester we were able to be more organized in that sense. As a team we found times that we were all off from work and other classes to meet as a group and work on the database, GUI, and documentation. Assignments were distributed to each team

member to be completed before the next meeting. By using this method, we were able to be more organized this semester and made sure that assignments were done on time and were corrected by the rest of the group if needed.

Keep the scope small

As stated in “Not everything is possible”, we had to shrink the scope of the project with the client because there were so many things that were not going to be possible to complete in the time restraint. As we brought the issue to the client and talked, we had the client discuss what were the most important parts of her system that she would like to be improved the most. As a team we learned that the best way is to start small and not promise a vast amount of things because many times due to lack amount of resources and time restraint not everything is possible.

Hope for the best, expect the worst

Our group would often assume things were going to go smoothly when they did not. As we welcome new team members to the group that were not as familiar with the project as the members who transitions from last year we realized the best would not always be possible. Although new team members had the chance to look over the data to understand the purpose of the project, it was still a bit hard for those new team members to understand it easily. Not only did we have issues with new team members understanding the project, but as a group we also had issues with meetings since everyone had their own schedule of work and classes. It was hard to come across times that we all were available.

The client might not be able to make decisions

When we first talked to our client there were many things that needed to be addressed, but as we began to build the project we realized that there were many things that could not make the final version. As a result we had to cut some of the requirements. In order to do this we had to schedule a meeting with her and our group to discuss what features we thought would be best to keep. As a result we learned that sometimes the client needs to

be realistic and as a group decisions need to be made in order to meet the time frame for the project.

Update Data Dictionary Constantly

One of the biggest mistakes that we faced while creating the database was not keeping the data dictionary up to date as we made changes to the database. This caused a lot of problems towards the end of development since the dictionary was not matching with the table information in the SQL database. This caused confusion for our team member in charge if the data dictionary and it also wasted time that we could have allocated to other documentation. The most important thing we learned from this was that is important to update the data dictionary as you build and make changes to the database that way everything is synced.

Project Summary

In the current system Della's Design Solutions does not have a system in place that is efficient enough for the clients and the owner. Currently Della's Design Solutions manually inputs requests and related prices into a word document. The prices and calculated down payments are done manually. The current system does not provide storage to organize the data of different projects that are being worked on for different clients which turn into a waste of time, resources, and money. Currently the system has unlinked data which makes it potentially vulnerable to the loss of data due to the lack of backups of important clients and company data. Our group's goal is to create a new system that can input client info and their project scope to generate invoices, determine the service prices, calculate deposits for projects, and keep track of hours. The application that we are proposing to Della's Design Solution will be a SQL database in the back end where client information and project requests, services, additional costs and various calculations will be performed based on the parameters of the project scope. The Java application and the SQL database will provide Della's Design an opportunity to be more efficient, reduce data redundancy, and provide better access to client data for quicker, more efficient results

References

Archiveddocs. “Table-Valued User-Defined Functions.” *Microsoft Docs*, [docs.microsoft.com/en-us/previous-versions/sql/sql-server-2008-r2/ms191165\(v=sql.105\)](https://docs.microsoft.com/en-us/previous-versions/sql/sql-server-2008-r2/ms191165(v=sql.105)).

Coronel, Carlos, and Steven Morris. *Database Systems: Design, Implementation, and Management*. Cengage Learning, 2017.

“Execute User-Defined Functions.” *Microsoft TechNet*, [technet.microsoft.com/en-us/library/ms175562\(v=sql.110\).aspx](https://technet.microsoft.com/en-us/library/ms175562(v=sql.110).aspx).

Rothja. “Create User-Defined Functions (Database Engine).” *Microsoft Docs*, docs.microsoft.com/en-us/sql/relational-databases/user-defined-functions/create-user-defined-functions-database-engine?view=sql-server-2017.

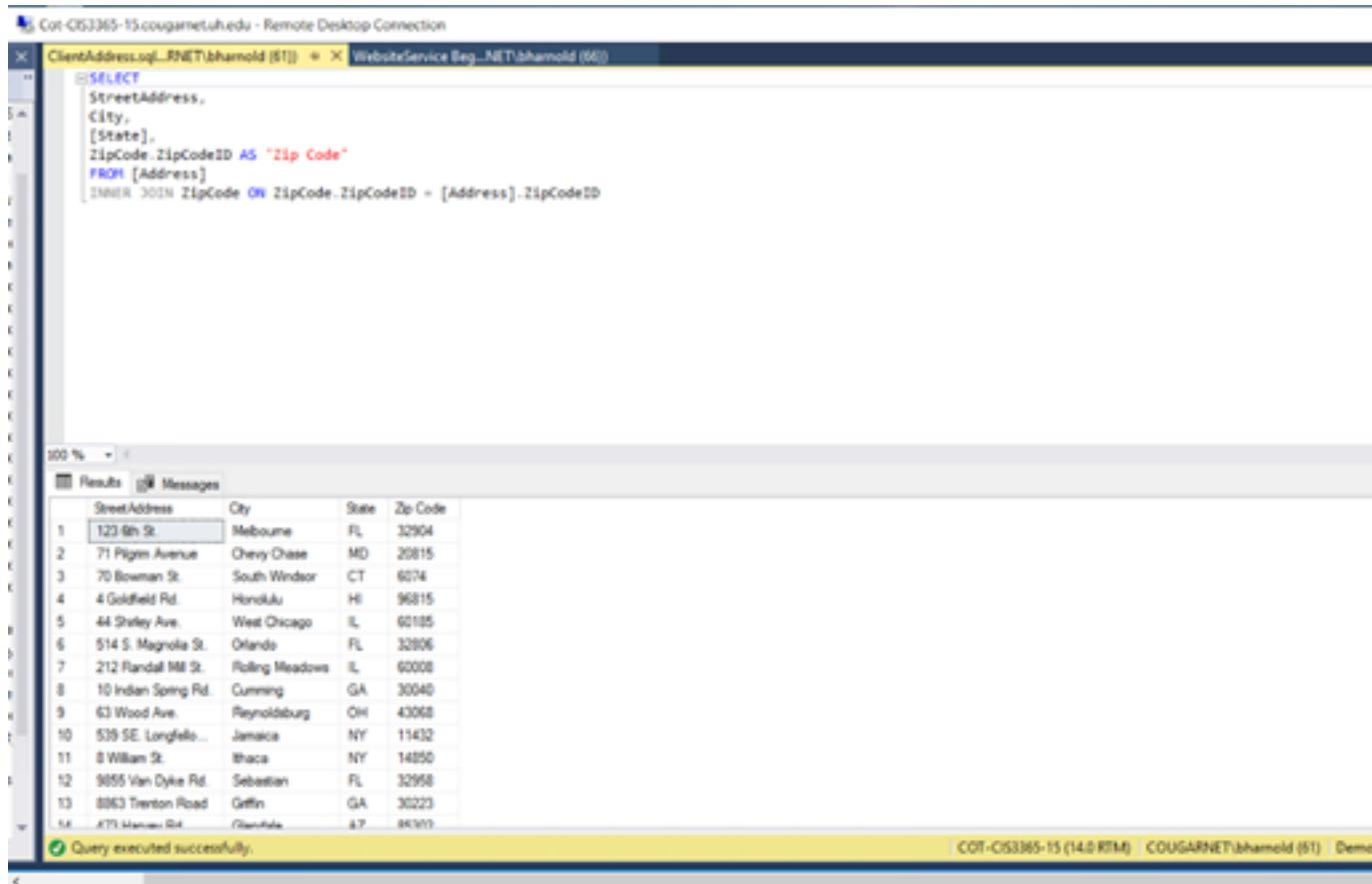
Seifallah. “Seifallah/Dynamic-TableView--Java-Fx-2.0-.” *GitHub*, github.com/seifallah/Dynamic-TableView--Java-Fx-2.0-/blob/master/DynamicTable.java.

Appendices

Brooke Schapiro

ClientAddress

ClientAddress query results all addresses associated with clients. Reason why our client would want to use this is to specify a certain address for a client.



The screenshot shows a Microsoft SQL Server Management Studio window titled 'COT-C63365-15.cougarnet.uh.edu - Remote Desktop Connection'. It has two tabs open: 'ClientAddress.sql..._PNET\bhamold (SI)' and 'WebsiteService [Seq..._NET\bhamold (60)]'. The 'ClientAddress' tab contains a T-SQL query:

```
SELECT
    StreetAddress,
    City,
    [State],
    ZipCode.ZipCodeID AS "Zip Code"
FROM [Address]
INNER JOIN ZipCode ON ZipCode.ZipCodeID = [Address].ZipCodeID
```

The results grid displays 14 rows of address data:

	StreetAddress	City	State	Zip Code
1	123 8th St.	Melbourne	FL	32904
2	71 Pilgrim Avenue	Chevy Chase	MD	20815
3	70 Bowman St.	South Windsor	CT	6074
4	4 Goldfield Rd.	Honolulu	HI	96815
5	44 Shirley Ave.	West Chicago	IL	60185
6	514 S. Magnolia St.	Orlando	FL	32806
7	212 Randal Mill St.	Rolling Meadows	IL	60008
8	10 Indian Spring Rd.	Cumming	GA	30040
9	63 Wood Ave.	Reynoldsburg	OH	43068
10	539 SE Longfellow...	Jamaica	NY	11432
11	8 William St.	Bhaca	NY	14850
12	9855 Van Dyke Rd.	Sebastian	FL	32958
13	8863 Trenton Road	Geffin	GA	30223
14	477 Marconi Rd.	Rowhale	AT	88999

At the bottom of the results grid, a message says 'Query executed successfully.' and the status bar shows 'COT-C63365-15 (14.0 RTM) COUGARNET\bhamold (SI) Demo'.

Function - SearchEmail

This function filters domain search by username (email) to pull up their domain and password. Della can use this function to find the domain and password of a client through their email as the email title is their username.

The screenshot shows a SQL Server Management Studio (SSMS) interface. At the top, there's a dark header bar with the text "Function - SearchEmail" and "COUGARNET\jakim (SO)". Below this is a code editor window containing a T-SQL script:

```
SELECT
    ClientDomain,
    Username,
    [Password]
FROM ClientDomain
INNER JOIN
    WebmasterLogin ON WebmasterLogin.WebmasterLoginID = ClientDomain.WebmasterLoginID
WHERE ClientDomain.ClientDomainID = dbo.SearchEmail('tlowry@website.com')
```

Below the code editor is a results grid titled "Results". It has three columns: "ClientDomain", "Username", and "Password". A single row is displayed, with the "ClientDomain" value "website.com" highlighted in a yellow box. The row data is:

	ClientDomain	Username	Password
1	website.com	tlowry	password

At the bottom of the results grid, a green status bar indicates "Query executed successfully." To the right of the status bar, the server name "COT-CIS3365-15 (14.0 RTM)" and the user "COUGARNET\jakim (SO)" are shown.

Search GraphicDesignService

This query filters the first and last name of the client and what graphic design service was completed for them. Also shows the balance, deposit, amount paid towards balance and dates of when payment was made and services done. Della can use this to find the information of a past service to a specific client.

```
Search GraphicDes...GARNET\jakim (31)  ✘
SELECT CONCAT(sub8.FirstName, ' ', sub8.LastName) AS 'Name', CONCAT(sub8.StreetAddress, ' ', sub8.City, ' ', sub8.[State], ' ', sub8.ZipCodeID,
CONCAT('(', sub8.AreaCodeID, ')', ' ', Number) AS 'Phone Number', sub8.ClientDomain AS 'Client Domain',
sub8.email, sub8.[Services_Description] AS 'Service', GraphicDesignJob AS 'Job Description', sub8.Balance, sub8.Deposit, sub8.Paid,
PaidToDate, sub8.Service_Date, CASE WHEN sub8.TempClient = 1 THEN 'Yes' ELSE 'No' END AS 'Temp Client' FROM GraphicDesignJob
INNER JOIN (
    SELECT sub7.ClientID, sub7.FirstName, sub7.LastName, sub7.StreetAddress, sub7.City, sub7.[State], sub7.ZipCodeID, AreaCodeID, Number, ClientDomain,
    sub7.email, sub7.ServicesID, sub7.[Services_Description], sub7.GraphicDesignJobID, sub7.Balance, sub7.Deposit, sub7.Paid,
    PaidToDate, sub7.Service_Date, sub7.TempClient FROM ClientDomain
) sub7 ON sub7.ClientID = sub8.ClientID
INNER JOIN (
    SELECT sub6.ClientID, sub6.FirstName, sub6.LastName, sub6.StreetAddress, sub6.City, sub6.[State], sub6.ZipCodeID, AreaCodeID, Number, sub6.ClientDomain,
    sub6.ServicesID, sub6.[Services_Description], sub6.GraphicDesignJobID, sub6.Balance, sub6.Deposit, sub6.Paid,
    PaidToDate, sub6.Service_Date, sub6.TempClient FROM PhoneNumber
) sub6 ON sub6.ClientID = sub8.ClientID
INNER JOIN (
    SELECT sub5.ClientID, sub5.FirstName, sub5.LastName, sub5.StreetAddress, City, [State], ZipCode.ZipCodeID, sub5.PhoneNumberID, sub5.ClientDomain,
    sub5.ServicesID, sub5.[Services_Description], sub5.GraphicDesignJobID, sub5.Balance, sub5.Deposit, sub5.Paid,
    PaidToDate, sub5.Service_Date, sub5.TempClient FROM ZipCode
) sub5 ON sub5.ClientID = sub8.ClientID
INNER JOIN (
    SELECT sub4.ClientID, sub4.FirstName, sub4.LastName, StreetAddress, ZipCodeID, sub4.PhoneNumberID, sub4.ClientDomainID, sub4.email,
    sub4.ServicesID, sub4.[Services_Description], sub4.GraphicDesignJobID,
    sub4.Balance, sub4.Deposit, sub4.Paid, PaidToDate, sub4.Service_Date, sub4.TempClient FROM [Address]
) sub4 ON sub4.ClientID = sub8.ClientID
INNER JOIN (
    SELECT Client.ClientID, FirstName, LastName, AddressID, PhoneNumberID, ClientDomainID, email, sub3.ServicesID, sub3.[Services_Description], sub3.Balance,
    sub3.Deposit, Paid, PaidToDate, sub3.Service_Date, TempClient
    FROM Client
) sub3 ON sub3.ClientID = sub8.ClientID
INNER JOIN (
    SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.GraphicDesignJobID, sub2.Balance, sub2.Deposit, Paid, PaidToDate, sub2.Service_Date, sub2.TempClient
    FROM GraphicDesignJob
) sub2 ON sub2.ClientID = sub8.ClientID
)
100 %  → 4
Results Messages
Name Address Phone Number Client Domain email Service Job Description Balance Deposit Paid PaidT
1 Vicki Coons 8863 Trenton Road, Griffin, GA 30223 (404) 8920156 invisibleplane.com vcoons@invisibleplane.com Alternate Company logo 420.00 $4.00 30.00 2018
Query executed successfully. COT-CIS3365-15 (14.0 RTM) COUGARNET\jakim (31)
```

WebsiteService with Partial Client and Complete Address and Complete Phone Number.sql

This report shows the website service with the partial client and complete address and complete phone number. Della's Design needs an easier way to look up current client's complete address and complete phone numbers in case she needs to contact them for any questions or issues with their current orders. This report will help Della's Design be able to easily get a list of the clients that are currently receiving services and see the phone number with the area code and the email address and contact them as soon as possible.

The screenshot shows a SQL Server Management Studio window. At the top, there is a query editor tab labeled "WebsiteService wit...GARNET\jakim (59)" containing the following T-SQL code:

```
FROM [Services] WHERE [Services].WebsiteJobID IS NOT NULL ) sub ON sub.BalanceID = [Balance].BalanceID ) sub2
ON sub2.PaidID = Paid.PaidID ) sub3
ON sub3.ClientID = Client.ClientID ) sub4
ON sub4.AddressID = [Address].AddressID ) sub5
ON sub5.ZipCodeID = ZipCode.ZipCodeID ) sub6
ON sub6.PhoneNumberID = Phonenumbers.PhoneNumberID ) sub7
ON sub7.ClientDomainID = ClientDomain.ClientDomainID
```

Below the code is a results grid titled "Results" showing 9 rows of client data:

ClientID	FirstName	LastName	StreetAddress	City	State	ZipCodeID	AreaCodeID	Number	ClientDomain	Email
1	Anna	Rhynes	71 Pilgrim Avenue	Chevy Chase	MD	20815	713	6905413	myspace.com	arhyne
2	Mindi	Karla	70 Bowman St.	South Windsor	CT	6074	281	3308004	dubyastep.com	mkarla
3	Kent	Steinfieldt	4 Goldfield Rd.	Honolulu	HI	96815	209	4531289	howtomom.com	ksterfi
4	Arlan	Bagley	514 S. Magnolia St.	Orlando	FL	32806	503	1973482	gotapples.com	abagle
5	Masako	Shirah	473 Harvey Rd.	Glendale	AZ	85302	630	7841354	vaticanagain.com	mhishal
6	Edward	Zoll	275 Arnold Court	Soddy Daisy	TN	37379	410	2214563	southlandpark.com	ezoll@
7	Rosanne	Gremillion	770 S. Willow Avenue	Navarre	FL	32566	610	5569520	pewds.io	rgremill
8	Bruce	Dalke	77 S. Prospect Dr.	Downers Grove	IL	60515	416	8742351	admin.com	bdalke
9	Jorge	Barroso	1 North Second Drive	Concord	NH	3301	225	3146795	strosworldchamps.com	jbarros

At the bottom of the screen, a message bar indicates "Query executed successfully." and shows the session details: COT-CIS3365-15 (14.0 RTM) | COUGARNET\jakim (59) | Demo

WebsiteService with Partial Client and Partial Address.sql

This query report shows the website service with the partial client and the partial address. Della's Design likes to sometimes just search partial parts for the client and the address to make it a short list. In this report only the first name, last name, street address, and zip code are generated which gives her a shorter description for customer information which will fulfill this requirement that she has asked to complete.

The screenshot shows the SQL Server Management Studio interface. The top pane displays a complex SQL query, and the bottom pane shows the resulting table of data.

```
WebsiteService wit...GARNET\jakim (56) - X
SELECT sub4.ClientID, sub4.FirstName, sub4.LastName, StreetAddress, ZipCodeID, sub4.PhoneNumberID, sub4.ClientDomainID, sub4.ServicesID, sub4.[Services_Description], sub4.WebsiteJobID, sub4.Balance, sub4.Deposit, sub4.Paid, PaidToDate, sub4.Service_Date, sub4.TempClient
FROM [Address]
INNER JOIN (
    SELECT Client.ClientID, FirstName, LastName, AddressID, PhoneNumberID, ClientDomainID, email, sub3.ServicesID, sub3.Balance, sub3.Deposit, Paid, PaidToDate, sub3.Service_Date, TempClient
    FROM Client
    INNER JOIN (
        SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.WebsiteJobID, sub2.Balance, sub2.Deposit
        INNER JOIN (
            SELECT sub.ServicesID, sub.[Services_Description], sub.ClientID, sub.WebsiteJobID, Balance, Deposit, PaidID, ServiceDate
            INNER JOIN (
                SELECT ServicesID, [Services_Description], ClientID, WebsiteJobID, BalanceID, PaidID, ServiceDate
                FROM [Services] WHERE [Services].WebsiteJobID IS NOT NULL ) sub ON sub.BalanceID = [Balance].BalanceID ) sub2
            ON sub2.PaidID = Paid.PaidID ) sub3
        ON sub3.ClientID = Client.ClientID ) sub4
    ON sub4.AddressID = [Address].AddressID
```

ClientID	FirstName	LastName	StreetAddress	ZipCodeID	PhoneNumberID	ClientDomainID	email	ServicesID
1	2	Anna	Rhynes	71 Pilgrim Avenue	20815	2	arhynes@myspace.com	2
2	3	Mind	Karls	70 Bowman St.	6074	3	mikarls@dubyastep.com	3
3	4	Kent	Steinfeldt	4 Goldfield Rd.	96815	4	ksteinfeldt@howtomom.com	4
4	6	Arlan	Bagley	514 S. Magnolia St.	32806	6	abagley@gotapples.com	6
5	14	Masako	Shish	473 Harvey Rd.	85302	14	mhirah@vaticanagain.com	14
6	15	Edward	Zoll	275 Arnold Court	37379	15	ezoll@southlandpark.com	15
7	16	Rosanne	Gremillion	770 S. Willow Avenue	32566	16	rgremillion@pewds.io	16
8	18	Bruce	Dalke	77 S. Prospect Dr.	60515	18	bdalke@admin.com	18
9	20	Jorge	Baroso	1 North Second Drive	3301	20	jbaroso@astrosworldchamps.com	20

Query executed successfully. COT-CIS3365-15 (14.0 RTM) COUGARNET\jakim (56) Demo

WebsiteServiceWithAddon.sql

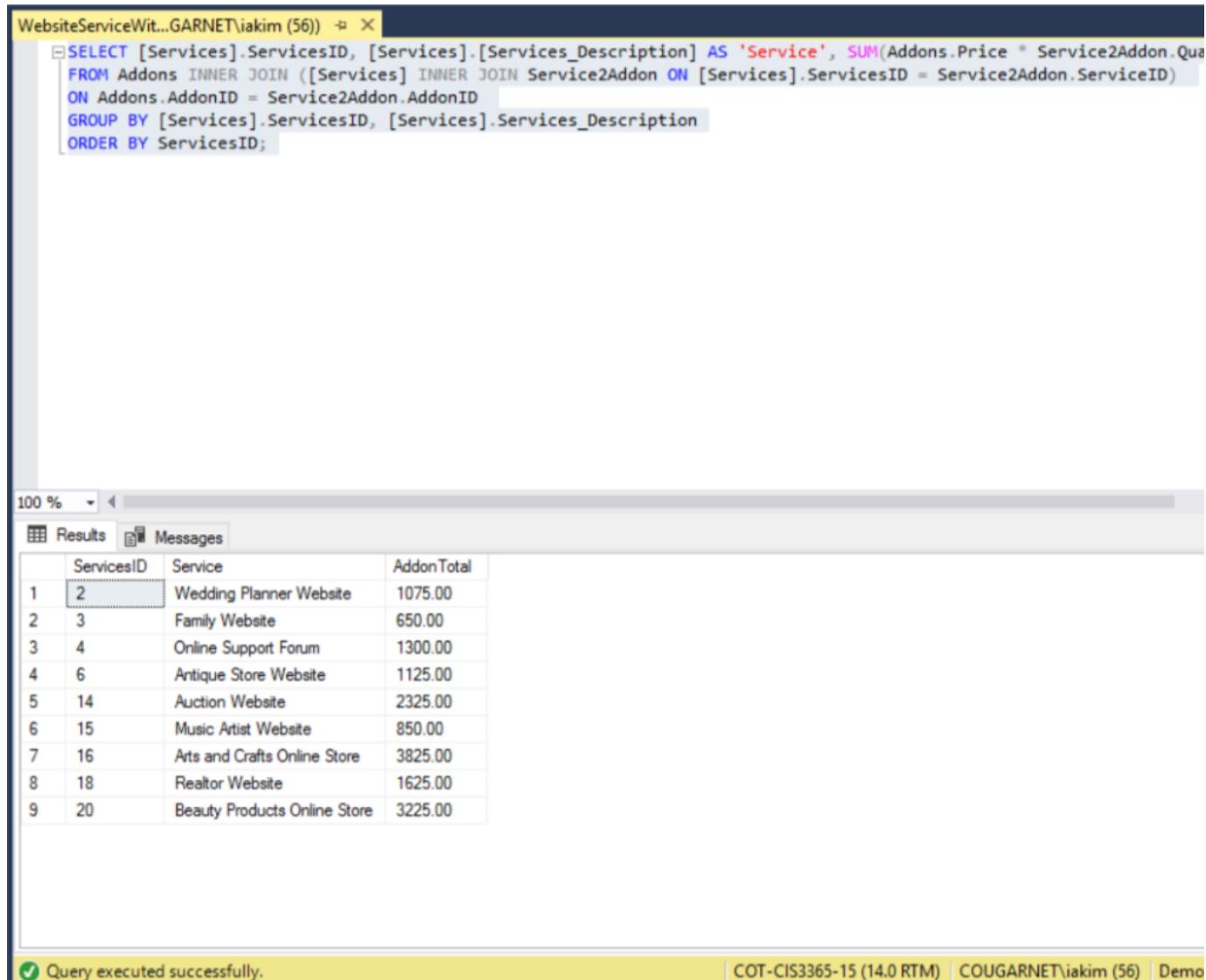
This query report takes the addon price with the quantity of addons that a client desires and multiplies it, it is then stored in “AddonTotal”. When this report is ran it will output the Service ID, name of the service, and it will give the total of the addon. Della’s Design currently does not have a way to keep track of addons that are added to services in an organized way that will help her keep track and give an accurate total to the client. This report will satisfy her requirement to be able to keep track of what addons are being added to which outstanding orders and the prices to be able to invoice the client.

The screenshot shows a SQL Server Management Studio window. At the top, there is a toolbar with icons for file operations. Below the toolbar is a tab bar with 'Results' and 'Messages' tabs, where 'Results' is selected. The main area displays a table of data with columns: ServicesID, Services_Description, Addon, Price, and Quantity. The data consists of 14 rows, each representing a service with its associated addons and their respective prices and quantities. The bottom of the window shows a status bar with the message 'Query executed successfully.'

	ServicesID	Services_Description	Addon	Price	Quantity
1	2	Wedding Planner Website	Unique Theme	250.00	1
2	2	Wedding Planner Website	Complete New Content Package	100.00	5
3	2	Wedding Planner Website	SEO	50.00	5
4	2	Wedding Planner Website	Custom Slideshow	75.00	1
5	3	Family Website	Content Editing	50.00	1
6	3	Family Website	Unique Theme	250.00	1
7	3	Family Website	Customized Font	200.00	1
8	3	Family Website	Copy Text	50.00	3
9	4	Online Support Forum	SEO	50.00	10
10	4	Online Support Forum	Accessibility Features	50.00	1
11	4	Online Support Forum	Security Features	200.00	1
12	4	Online Support Forum	Fully Responsive CSS	300.00	1
13	4	Online Support Forum	Unique Theme	250.00	1
14	5	Business Online Website	CEO	50.00	1

Bryan Arnold

WebsiteServiceWithAddon – Totaled.sql



The screenshot shows a SQL Server Management Studio window. The query pane displays a T-SQL script to calculate the total price for each website service including its add-ons. The results pane shows a grid of 10 rows, each representing a website service with its ID, description, and the total price of its add-ons. A message at the bottom indicates the query was executed successfully.

```
SELECT [Services].ServicesID, [Services].[Services_Description] AS 'Service', SUM(Addons.Price * Service2Addon.Quantity) AS AddonTotal
FROM Addons INNER JOIN ([Services] INNER JOIN Service2Addon ON [Services].ServicesID = Service2Addon.ServiceID)
ON Addons.AddonID = Service2Addon.AddonID
GROUP BY [Services].ServicesID, [Services].[Services_Description]
ORDER BY ServicesID;
```

	ServicesID	Service	AddonTotal
1	2	Wedding Planner Website	1075.00
2	3	Family Website	650.00
3	4	Online Support Forum	1300.00
4	6	Antique Store Website	1125.00
5	14	Auction Website	2325.00
6	15	Music Artist Website	850.00
7	16	Arts and Crafts Online Store	3825.00
8	18	Realtor Website	1625.00
9	20	Beauty Products Online Store	3225.00

Query executed successfully. | COT-CIS3365-15 (14.0 RTM) | COUGARNET\iakim (56) | Demo

Description:

This query report shows the Website Service with add on with the quantity calculated. Della's designed currently writes people add on requests on paper and the calculated the total user a calculator and writes it back in which is time consuming and is not able to tell customer their

total right away. This report will help her get rid of that issue but allowing her to run a report of the addons and it will not only give her total of the add ons that have been added by the customer but it will also give her a total of the add ons.

CleanWebsite DONE.sql

```

GraphicDesignServ..NET\bharnold (60)    Clean WebsiteDONE..ET\bharnold (59)  Object Explorer
SELECT
    CONCAT(FirstName, ' ', LastName) AS 'Name',
    CONCAT(StreetAddress, ' ', City, ' ', [State], ' ',
    ZipCode.ZipCodeID) AS 'Address',
    CONCAT('(', AreaCodeID, ')', ' ', Number) AS 'Phone Number',
    email,
    [Services_Description] AS 'Service',
    WebsiteJob AS 'Job Description',
    Balance,
    Deposit,
    Paid,
    PaidToDate,
    Service_Date, CASE WHEN TempClient = 1 THEN 'Yes' ELSE 'No' END AS 'Temp Client'
FROM WebsiteJob
INNER JOIN [Services] ON [Services].WebsiteJobID = WebsiteJob.WebsiteJobID
INNER JOIN Balance ON Balance.BalanceID = [Services].BalanceID
INNER JOIN Paid ON Paid.PaidID = [Services].PaidID
INNER JOIN Client ON Client.ClientID = [Services].PaidID
INNER JOIN [Address] ON [Address].AddressID = Client.AddressID
INNER JOIN ZipCode ON ZipCode.ZipCodeID = [Address].ZipCodeID
INNER JOIN PhoneNumber ON PhoneNumber.PhoneNumberID = Client.PhoneNumberID
INNER JOIN ClientDomain ON ClientDomain.ClientDomainID = Client.ClientDomainID
    
```

	Name	Address	Phone Number	email	Service	Job Description	Balance	Deposit	Paid	PaidToDate
1	Anna Rhynes	71 Pilgrim Avenue, Chevy Chase, MD 20815	(713) 6905413	arhynes@myspace.com	Wedding Planner Website	Medium Personal (6-10 pages)	1575.00	315.00	600.00	2018-04-17
2	Mindi Karls	70 Bowman St., South Windsor, CT 06074	(281) 3308004	mkarls@dubyastep.com	Family Website	Small Personal (1-5 pages)	950.00	190.00	725.00	2018-04-06
3	Kent Steinfield	4 Goldfield Rd., Honolulu, HI 96815	(209) 4531289	ksteinfieldt@howtomom.com	Online Support Forum	Forum Website	2550.00	510.00	100.00	2018-04-26
4	Arlean Bagley	514 S. Magnolia St., Orlando, FL 32806	(503) 1973482	abagley@gotapple.com	Antique Store Website	Medium Business	2125.00	425.00	75.00	2018-04-05
5	Masako Shirah	473 Harvey Rd., Glendale, AZ 85302	(630) 7841354	mshirah@vaticanagain.com	Auction Website	Auction Website	3825.00	765.00	800.00	2018-03-14
6	Edward Zoll	275 Arnold Court, Soddy Daisy, TN 37379	(410) 2214563	ezoll@southlandpark.com	Music Artist Website	Music Website	1350.00	270.00	250.00	2018-04-03
7	Rosanne Gremillion	770 S. Willow Avenue, Navarre, FL 32566	(610) 5569520	rgremillion@pewds.io	Arts and Crafts Online Store	Large Online Store	5825.00	1165.00	150.00	2018-04-12
8	Bruce Dalke	77 S. Prospect Dr., Downers Grove, IL 60515	(416) 8742351	bdalke@admin.com	Realtor Website	Large Business	3125.00	625.00	300.00	2018-04-15
9	Jorge Barroso	1 North Second Drive, Concord, NH 3301	(225) 3146795	jbarroso@astrosworldchamps.com	Beauty Products Online Store	Medium Online Store	4725.00	945.00	1750.00	2018-04-15

Query executed successfully. | COT-CIS3365-15 (14.0 RTM) | COUGARNET\bharnold

Description: This Query pulls together, important client data and combines it with Website Services w/addons. This creates a complete view in terms of our ERD where client is combined with their services.

Function - FindJob.sql

Object Explorer Function - FindJob...NET\barnold (60) X

```

SELECT [Services].ServicesID, [Services].ClientID, [Services].Service_Date, [Services].[Services_Description],
       WebsiteJob.WebsiteJob, WebsiteJob.Price
  FROM WebsiteJob INNER JOIN [Services] ON WebsiteJob.WebsiteJobID = [Services].WebsiteJobID
 WHERE WebsiteJob.WebsiteJobID = dbo.FindJob('Auction Website');

```

100 %

Results Messages

	ServicesID	ClientID	Service_Date	Services_Description	WebsiteJob	Price
1	14	14	2017-02-13	Auction Website	Auction Website	1500.00

Description: This function pulls a JobID from The WebsiteJob and uses it as a search parameter to find all services with that job now

GraphicDesignServiceDone.sql

Object Explorer GraphicDesignService...NET\barnold (59) X

```

SELECT CONCAT(sub8.FirstName, ' ', sub8.LastName) AS 'Name', CONCAT(sub8.StreetAddress, ' ', sub8.City, ' ', sub8.[State], ' ', sub8.ZipCodeID) AS 'Address',
CONCAT('(', sub8.AreaCodeID, ')', ' ', Number) AS 'Phone Number', sub8.ClientDomain AS 'Client Domain',
sub8.email, sub8.[Services_Description] AS 'Service', GraphicDesignJob AS 'Job Description', sub8.Balance, sub8.Deposit, sub8.Paid,
PaidToDate, sub8.Service_Date, CASE WHEN sub8.TempClient = 1 THEN 'Yes' ELSE 'No' END AS 'Temp Client' FROM GraphicDesignJob
INNER JOIN (
    SELECT sub7.ClientID, sub7.FirstName, sub7.LastName, sub7.StreetAddress, sub7.City, sub7.[State], sub7.ZipCodeID, AreaCodeID, Number, ClientDomain.ClientDomain,
    sub7.email, sub7.ServicesID, sub7.[Services_Description], sub7.GraphicDesignJobID, sub7.Balance, sub7.Deposit, sub7.Paid,
    PaidToDate, sub7.Service_Date, sub7.TempClient FROM ClientDomain
    INNER JOIN (
        SELECT sub6.ClientID, sub6.FirstName, sub6.LastName, sub6.StreetAddress, sub6.City, sub6.[State], sub6.ZipCodeID, AreaCodeID, Number, sub6.ClientDomainID, sub6.emai
        sub6.ServicesID, sub6.[Services_Description], sub6.GraphicDesignJobID, sub6.Balance, sub6.Deposit, sub6.Paid,
        PaidToDate, sub6.Service_Date, sub6.TempClient FROM PhoneNumber
        INNER JOIN (
            SELECT sub5.ClientID, sub5.FirstName, sub5.LastName, sub5.StreetAddress, City, [State], ZipCode.ZipCodeID, sub5.PhoneNumberID, sub5.ClientDomainID, sub5.email,
            sub5.ServicesID, sub5.[Services_Description], sub5.GraphicDesignJobID, sub5.Balance, sub5.Deposit, sub5.Paid,
            PaidToDate, sub5.Service_Date, sub5.TempClient FROM ZipCode
            INNER JOIN (
                SELECT sub4.ClientID, sub4.FirstName, sub4.LastName, StreetAddress, ZipCodeID, sub4.PhoneNumberID, sub4.ClientDomainID, sub4.email,
                sub4.ServicesID, sub4.[Services_Description], sub4.GraphicDesignJobID,
                sub4.Balance, sub4.Deposit, sub4.Paid, PaidToDate, sub4.Service_Date, sub4.TempClient FROM [Address]

```

100 %

Results Messages

	Name	Address	Phone Number	Client Domain	email	Service	Job Description	Balance	Deposit	Pa
1	Timothy Lowry	123 6th St., Melbourne, FL 32904	(361) 9999999	website.com	tlowry@website.com	Spring Sale Advertisment	Ad	600.00	120.00	10
2	Howard Setzer	44 Shirley Ave., West Chicago, IL 60185	(417) 7413658	got.com	haetzer@got.com	New Company logo	Logo	300.00	60.00	12
3	Norman Janas	212 Randall Mill St., Rolling Meadows, IL 60008	(202) 1234567	texaslove.com	njanas@texaslove.com	Website Banner	Banner	480.00	96.00	35
4	Karlyn Mandeville	10 Indian Spring Rd., Cumming, GA 30040	(254) 3571598	thealamofosho.com	kmandeville@thealamofosho.com	Weekly Advertisment	Ad	330.00	66.00	20
5	Emilio Aphorne	63 Wood Ave., Reynoldsburg, OH 43068	(330) 4826716	anyme.com	ealphone@anyme.com	Custom Website Image	Image	450.00	90.00	30
6	Liliana Widman	539 SE. Longfellow Street, Jamaica, NY 11432	(401) 9102973	doublepumpshotgun.com	lwidman@doublepumpshotgun.com	Custom Design Seasonal Template	Design	360.00	72.00	24
7	Leena Phelps	8 William St., Ithaca, NY 14850	(270) 5643212	theydontunderstand.com	lphelps@theydontunderstand.com	Revised Company logo	Logo	450.00	90.00	40
8	Korey Larock	9855 Van Dyke Rd., Sebastian, FL 32958	(501) 4683218	batlink.com	klarock@batlink.com	Newspaper Advertisment	Ad	630.00	126.00	20
9	Vicki Coons	8863 Trenton Road, Griffin, GA 30223	(242) 8920156	invisibleplane.com	vcoons@invisibleplane.com	Alternate Company logo	Logo	420.00	84.00	30
10	Cara Pickford	7142 Mechanic Street, Goose Creek, SC 29445	(351) 5023070	default.com	cpickford@default.com	Custom Design Template	Design	630.00	126.00	75
11	Max Waiters	227 Broad Street, Valrico, FL 33594	(479) 8885892	supersad.com	mwaiters@supersad.com	Revised Website image	Image	150.00	30.00	55

Query executed successfully.

COT-CIS3365-15 (14.0 RTM) | COUGARNET\barnold

Description: This query pulls all client information and joins it with GraphicDesignServices this creates a full client view with their services.

GraphicDesignService with Partial Client and Complete Address.sql

```

Object Explorer  GraphicDesignServ...NET\bharnold (59)  X
SELECT sub5.ClientID, sub5.FirstName, sub5.LastName, sub5.StreetAddress, City, [State], ZipCode.ZipCodeID, sub5.PhoneNumberID, sub5.ClientDomainID, sub5.email,
sub5.ServicesID, sub5.[Services_Description], sub5.GraphicDesignJobID, sub5.Balance, sub5.Deposit, sub5.Paid,
PaidToDate, sub5.Service_Date, sub5.TempClient FROM ZipCode
INNER JOIN (
    SELECT sub4.ClientID, sub4.FirstName, sub4.LastName, StreetAddress, ZipCodeID, sub4.PhoneNumberID, sub4.ClientDomainID, sub4.email,
    sub4.ServicesID, sub4.[Services_Description], sub4.GraphicDesignJobID,
    sub4.Balance, sub4.Deposit, sub4.Paid, PaidToDate, sub4.Service_Date, sub4.TempClient FROM [Address]
) sub4 ON sub4.ClientID = ZipCode.ClientID
INNER JOIN (
    SELECT Client.ClientID, FirstName, LastName, AddressID, PhoneNumberID, ClientDomainID, email, sub3.ServicesID, sub3.[Services_Description], sub3.GraphicDesignJobID
    sub3.Balance, sub3.Deposit, Paid, PaidToDate, sub3.Service_Date, TempClient
) Client ON Client.ClientID = sub4.ClientID
INNER JOIN (
    SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.GraphicDesignJobID, sub2.Balance, sub2.Deposit, Paid, PaidToDate, sub2.Service_Date FROM P
) sub2 ON sub2.ClientID = Client.ClientID
INNER JOIN (
    SELECT sub.ServicesID, sub.[Services_Description], sub.ClientID, sub.GraphicDesignJobID, Balance, Deposit, PaidID, Service_Date FROM Balance
) sub ON sub.ClientID = sub2.ClientID
SELECT ServicesID, [Services_Description], ClientID, GraphicDesignJobID, BalanceID, PaidID, Service_Date
FROM [Services] WHERE [Services].GraphicDesignJobID IS NOT NULL ) sub ON sub.BalanceID = [Balance].BalanceID ) sub2
ON sub2.PaidID = Paid.PaidID ) sub3
ON sub3.ClientID = Client.ClientID ) sub4

```

Results

ClientID	FirstName	LastName	StreetAddress	City	State	ZipCodeID	PhoneNumberID	ClientDomainID	email	ServicesID	Services_Description
1	Timothy	Lowry	123 8th St.	Melbourne	FL	32904	1	1	tlowry@website.com	1	Spring Sale Advertisement
2	Howard	Setzer	44 Shirley Ave.	West Chicago	IL	60185	5	5	hsetzer@got.com	5	New Company logo
3	Norman	Janas	212 Randall Mill St.	Rolling Meadows	IL	60008	7	7	njanas@texaslove.com	7	Website Banner
4	Karlyn	Mandeville	10 Indian Spring Rd.	Cumming	GA	30040	8	8	kmmandeville@thealamofoshoh.com	8	Weekly Advertisement
5	Emilio	Alphonse	63 Wood Ave.	Reynoldsburg	OH	43068	9	9	ealphonese@aryme.com	9	Custom Website Image
6	Liliana	Widman	539 SE. Longfellow Street	Jamaica	NY	11432	10	10	lwidman@doublepumpshotgun.com	10	Custom Design Seasonal Template
7	Leena	Phelps	8 William St.	Ithaca	NY	14850	11	11	lphelps@theydontunderstand.com	11	Revised Company logo
8	Korey	Larock	9855 Van Dyke Rd.	Sebastian	FL	32958	12	12	klarock@batlink.com	12	Newspaper Advertisement
9	Vicki	Coons	8863 Trenton Road	Griffin	GA	30223	13	13	vcoons@invisibleplane.com	13	Alternate Company logo
10	Cara	Pickford	7142 Mechanic Street	Goose Creek	SC	29445	17	17	cpickford@default.com	17	Custom Design Template
11	Max	Waiters	227 Broad Street	Valrico	FL	33594	19	19	mwaiters@supersad.com	19	Revised Website image

Query executed successfully.

Description: This Query pulls Part of the Client information and combines it with the complete address to form a full address with a client name.

Search Client.sql

The screenshot shows the SSMS interface with the following details:

- Object Explorer:** Shows a connection to "Search Client.sql -...RNET\bharnold (59)".
- Results Pane:** Displays the query results in a table format.
- Query Text:**

```

SELECT FirstName AS 'First Name', LastName AS 'Last Name', sub3.StreetAddress AS 'Street Address', sub3.City, sub3.[State], sub3.[Zip Code], CONCAT(sub.AreaCodeID, sub.City, sub.State, sub.ZipCodeID) AS 'Full Address'
INNER JOIN ( SELECT AreaCodeID, Region FROM AreaCode ) sub
ON sub.AreaCodeID = PhoneNumber.AreaCodeID
INNER JOIN (
    SELECT ClientID, FirstName, LastName, sub2.StreetAddress, sub2.City, sub2.[State], sub2.[Zip Code], PhoneNumberID, ClientDomainID, Email, TempClient FROM Client
    INNER JOIN (
        SELECT AddressID, StreetAddress, sub.City, sub.[State], sub.ZipCodeID AS 'Zip Code' FROM [Address]
        INNER JOIN (SELECT ZipCodeID, City, [State], County FROM ZipCode) sub
        ON sub.ZipCodeID = [Address].ZipCodeID ) sub2
        ON sub2.AddressID = Client.AddressID ) sub3
        ON sub3.ClientID = PhoneNumber.PhoneNumberID
WHERE FirstName LIKE 'V%'
```
- Results Table:**

	First Name	Last Name	Street Address	City	State	Zip Code	Phone Number	Email	Temp Client
1	Vicki	Coons	8863 Trenton Road	Griffin	GA	30223	242-8920156	vcoons@invisibleplane.com	No

Description: Search a full client table with a FirstName as a parameter and select all clients that fit that criteria.

Wilfredo Sandoval

Function – FindAddon

Description:

The function that was created is called FindAddon. What this function does is searched for different services and addons. This gives you the ServiceID, Service Description, addon, Price, and Quantity. My specific function searches for the SEO add on.

A screenshot of the SQL Server Management Studio interface. The top pane shows a SQL query window with the following code:

```
SQLQuery1.sql - C:\...NET\wasandov (57)*  ✎ X
SELECT [Services].ServicesID, [Services].[Services_Description], Addons.Addon, Addons.Price, Service2Addon.Quantity
FROM Addons INNER JOIN ([Services] INNER JOIN Service2Addon ON [Services] ServicesID = Service2Addon.ServiceID)
ON Addons.AddonID = Service2Addon.AddonID
WHERE Addons.AddonID = dbo.FindAddon('SEO')
ORDER BY ServicesID ;
```

The bottom pane shows the execution results in a grid format:

	ServicesID	Services_Description	Addon	Price	Quantity
1	2	Wedding Planner Website	SEO	50.00	5
2	4	Online Support Forum	SEO	50.00	10
3	6	Antique Store Website	SEO	50.00	1
4	14	Auction Website	SEO	50.00	20
5	16	Arts and Crafts Online Store	SEO	50.00	30
6	20	Beauty Products Online Store	SEO	50.00	25

At the bottom of the results grid, there is a message: "Query executed successfully." To the right of the message, the status bar displays: "COT-CIS3365-15 (14.0 RTM) COUGARNET\wasandov (57) Demo".

GraphicDesignServices with Partial Client and Complete Address and Complete Phone Number and Domain

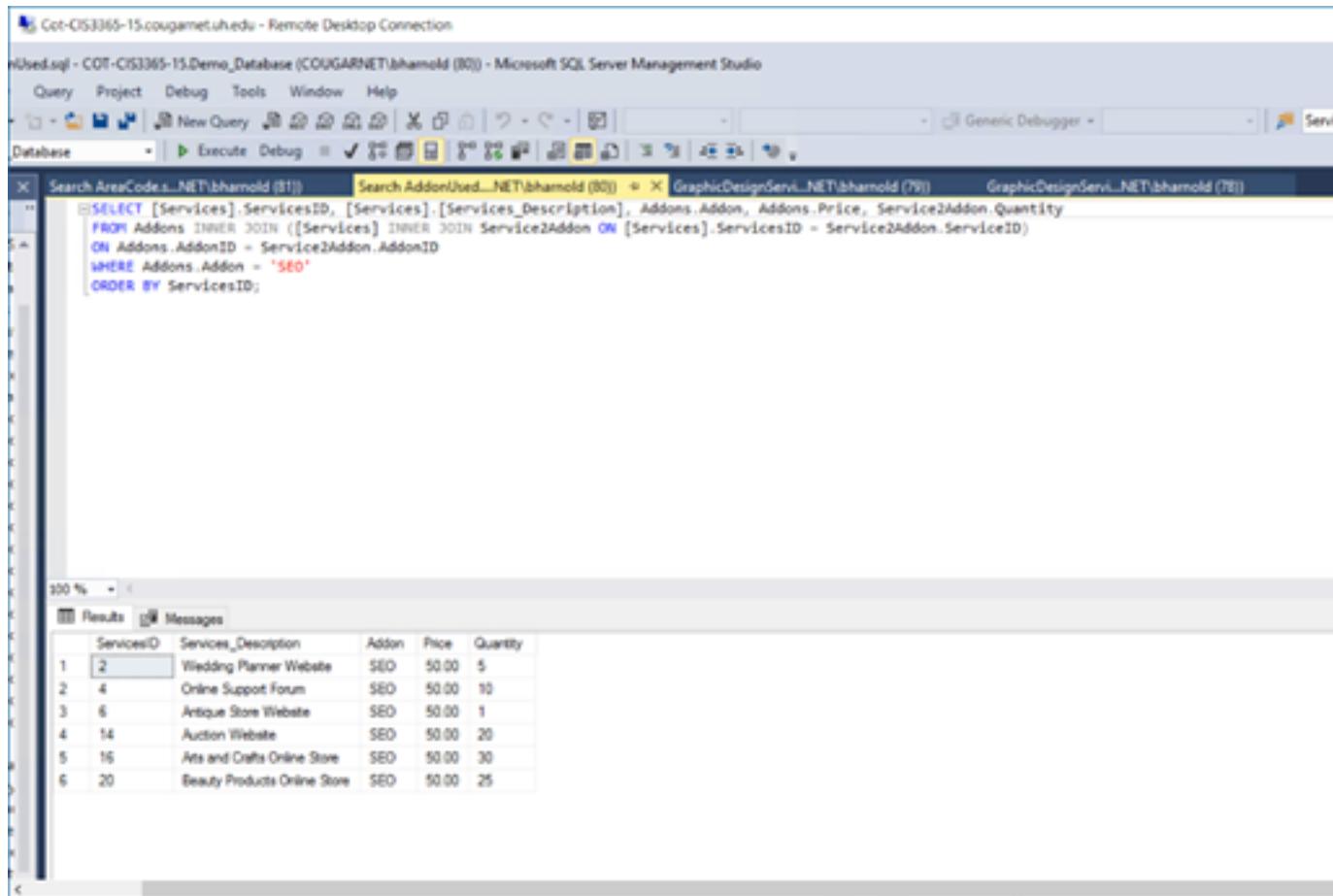
Description:

This query report outputs the graphic design services with partial client, complete address, complete phone number and domain. Currently Della's design does not have a way to pull a report to show her complete information about the client. This query will allow her to be able to pull up a report of the client that will show the complete address, phone number, and the domain. This will help her save time when she does searches for customers.

```
File Edit View Query Project Debug Tools Window Help
New Query Execute Generic Debugger
Demo_Database GraphicDesignService with Partial Client and Complete Address and Complete Phone Number and Domain.sql - COT-CIS3365-15.Demo_Database (COUGARNET\akim (S2)) - Microsoft SQL Server Management Studio
SELECT sub7.ClientID, sub7.FirstName, sub7.LastName, sub7.StreetAddress, sub7.City, sub7.[State], sub7.ZipCodeID, AreaCodeID, Number, ClientDomain.ClientDomainName, sub7.email, sub7.ServicesID, sub7.[Services_Description], sub7.GraphicDesignJobID, sub7.Balance, sub7.Deposit, sub7.Paid, PaidToDate, sub7.Service_Date, sub7.TempClient FROM ClientDomain
INNER JOIN (
    SELECT sub6.ClientID, sub6.FirstName, sub6.LastName, sub6.StreetAddress, sub6.City, sub6.[State], sub6.ZipCodeID, AreaCodeID, Number, sub6.ClientDomainID, sub6.email, sub6.ServicesID, sub6.[Services_Description], sub6.GraphicDesignJobID, sub6.Balance, sub6.Deposit, sub6.Paid, PaidToDate, sub6.Service_Date, sub6.TempClient FROM PhoneNumber
) INNER JOIN (
    SELECT sub5.ClientID, sub5.FirstName, sub5.LastName, sub5.StreetAddress, City, [State], ZipCode.ZipCodeID, sub5.PhoneNumberID, sub5.ClientDomainID, sub5.email, sub5.ServicesID, sub5.[Services_Description], sub5.GraphicDesignJobID, sub5.Balance, sub5.Deposit, sub5.Paid, PaidToDate, sub5.Service_Date, sub5.TempClient FROM ZipCode
) INNER JOIN (
    SELECT sub4.ClientID, sub4.FirstName, sub4.LastName, StreetAddress, ZipCodeID, sub4.PhoneNumberID, sub4.ClientDomainID, sub4.email, sub4.ServicesID, sub4.[Services_Description], sub4.GraphicDesignJobID, sub4.Balance, sub4.Deposit, sub4.Paid, PaidToDate, sub4.Service_Date, sub4.TempClient FROM [Address]
) INNER JOIN (
    SELECT Client.ClientID, FirstName, LastName, AddressID, PhoneNumberID, ClientDomainID, email, sub3.ServicesID, sub3.[Services_Description], sub3.GraphicDesignJobID, sub3.Balance, sub3.Deposit, Paid, PaidToDate, sub3.Service_Date, TempClient
    FROM Client
) ON Client.ClientID = sub4.ClientID
TEMPDB..#Temp1 /
```

ClientID	FirstName	LastName	StreetAddress	City	State	ZipCodeID	AreaCodeID	Number	ClientDomain	email	ServicesID	Services_Description
1	Timothy	Lovry	123 8th St.	Melbourne	FL	32964	361	9999999	website.com	tlovy@website.com	1	Spring Sale Advertisement
2	Howard	Setzer	44 Shaley Ave.	West Chicago	IL	60185	417	7413658	gct.com	haetzer@gct.com	5	New Company logo
3	Norman	Jones	212 Randall Mill St.	Rolling Meadows	IL	60008	202	1234567	teasalive.com	rjones@teasalive.com	7	Website Banner
4	Katlyn	Mandeville	10 Indian Spring Rd.	Cunning	GA	30040	254	3571598	thewalnutofishi.com	kmmandeville@thewalnutofishi.com	8	Weekly Advertisement
5	Emily	Aphonse	63 Wood Ave.	Reynoldsburg	OH	43068	330	4826716	anyone.com	eaphonse@anyone.com	9	Custom Website Image
6	Ulliana	Wildman	539 SE Longfellow Street	Jamaica	NY	11402	401	3102973	doublepumpshotgun.com	lwdman@doublepumpshotgun.com	10	Custom Design Seasonal T
7	Leena	Phelps	8 William St.	Bhace	NY	14850	279	5643212	theydontunderstand.com	lphelps@theydontunderstand.com	11	Revised Company logo
8	Korey	Larsen	9855 Van Dyke Rd.	Sebastian	FL	32958	501	4683218	batlink.com	k.larsen@batlink.com	12	Newspaper Advertisement
9	Vicki	Coone	8863 Trenton Road	Giffin	GA	30223	242	8920156	invisibleplane.com	vcoone@invisibleplane.com	13	Alternate Company logo
10	Cara	Polkford	7142 Mechanic Street	Gosse Creek	SC	29445	351	5023670	default.com	cwickford@default.com	17	Custom Design Template
11	Max	Waters	227 Broad Street	Venice	FL	33584	479	8889992	supersad.com	mwaters@supersad.com	19	Revised Website Image

Search AddonUsed



The screenshot shows a Microsoft SQL Server Management Studio (SSMS) interface. The title bar reads "C:\COT-CIS3365-15.cougamet.uh.edu - Remote Desktop Connection" and "Used.sql - COT-CIS3365-15.Demo_Database (COUGARNET\bhamold (B)) - Microsoft SQL Server Management Studio". The menu bar includes "Query", "Project", "Debug", "Tools", "Window", and "Help". The toolbar has icons for "New Query", "Save", "Run", "Stop", and "Break". The "Database" dropdown is set to "Used". The query window contains the following T-SQL code:

```
SELECT [Services].[ServicesID], [Services].[Services_Description], Addons.Addon, Addons.Price, Service2Addon.Quantity
FROM Addons INNER JOIN ([Services] INNER JOIN Service2Addon ON [Services].ServicesID = Service2Addon.ServiceID)
ON Addons.AddonID = Service2Addon.AddonID
WHERE Addons.Addon = 'SEO'
ORDER BY ServicesID;
```

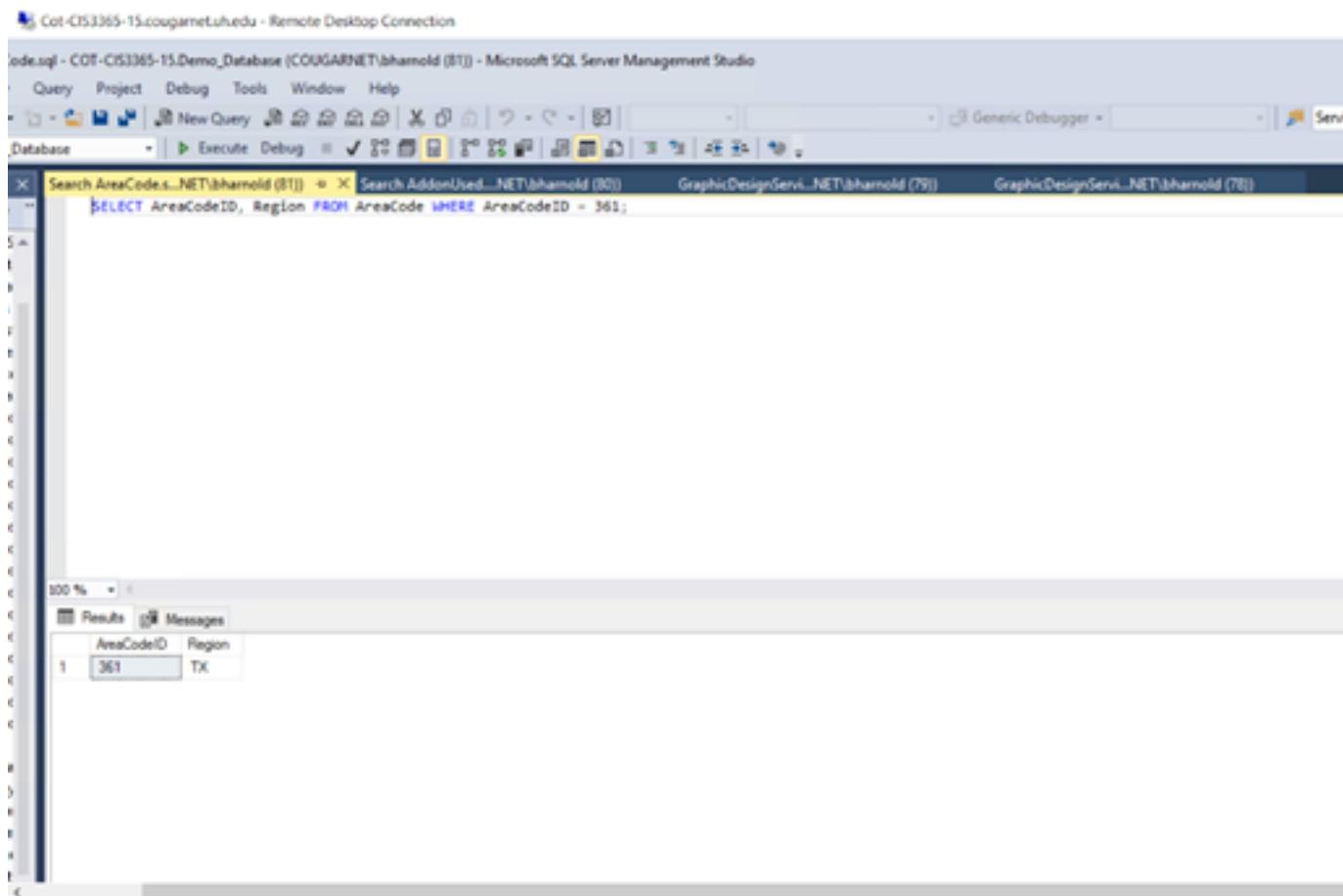
The results window shows a table with the following data:

ServicesID	Services_Description	Addon	Price	Quantity
1	2	SEO	50.00	5
2	4	SEO	50.00	10
3	6	SEO	50.00	1
4	14	SEO	50.00	20
5	16	SEO	50.00	30
6	20	SEO	50.00	25

Description:

This query report uses my search function to search for outputs of the SEO addon that is being used. Currently Della's Design does not have a system that searches for specific addons. This function and query will help her search for this addon easier. This addon is really important because most of the clients look for it. Therefore, it will be useful for her to use.

Search AreaCode



The screenshot shows a Microsoft SQL Server Management Studio (SSMS) interface. The title bar reads "code.sql - COT-CIS3365-15.Demo_Database (COUGARNET\bharnold (BT)) - Microsoft SQL Server Management Studio". The menu bar includes "Query", "Project", "Debug", "Tools", "Window", and "Help". The toolbar has various icons for file operations like New Query, Save, Print, and Execute. The main window displays a query results grid. The query is:

```
SELECT AreaCodeID, Region FROM AreaCode WHERE AreaCodeID = 361;
```

The results grid shows one row:

AreaCodeID	Region
361	TX

Description:

This Query report results in the search of which area code belongs to which state. Della currently does not have a way to search for area codes that belongs to states. This is useful to her and will help her day to day operations when she invoices to customers.

WebsiteService with Partial Client and Complete Address.sql

The screenshot shows a SQL Server Management Studio window with a query results grid. The query retrieves client information including first name, last name, street address, city, state, zip code, phone number, client domain ID, and email. The results grid contains 9 rows of data.

ClientID	FirstName	LastName	StreetAddress	City	State	ZipCodeID	PhoneNumberID	ClientDomainID	email
2	Anna	Rhynes	71 Pilgrim Avenue	Chevy Chase	MD	20815	2	2	arhynes@myspace.com
3	Mindi	Karla	70 Bowman St.	South Windsor	CT	6074	3	3	mkarla@dubyastep.com
4	Kent	Steinfieldt	4 Goldfield Rd.	Honolulu	HI	96815	4	4	ksteinfieldt@howtome.com
6	Adean	Bagley	514 S. Magnolia St.	Orlando	FL	32806	6	6	abagley@gotapples.com
14	Masako	Shirah	473 Harvey Rd.	Glendale	AZ	85302	14	14	mahirah@vaticanagai.com
15	Edward	Zoll	275 Arnold Court	Soddy Daisy	TN	37379	15	15	ezoll@southlandpark.com
16	Rosanne	Gremillion	770 S. Willow Avenue	Navare	FL	32566	16	16	rgremillion@pewds.io
18	Bruce	Dalke	77 S. Prospect Dr.	Downers Grove	IL	60515	18	18	bdalke@admin.com
20	Jorge	Barroso	1 North Second Drive	Concord	NH	3301	20	20	jbarroso@astroworld.com

Query executed successfully.

Description:

This query reports represents the website service with the partial client and the complete address for that specific client. Currently Della's Design keeps all her customer data in paper which are stored in file cabinets and every time she wants to access this information she has to go through tons of papers to get to the specific client that she is looking for. This report will help Della's design be able to pull a report of all the current clients that services are being provided to and she will be able to tell the clients first name, last name, and address as well as the email address.

WebsiteService with Partial Client.sql

The screenshot shows a SQL Server Management Studio window. The top pane displays a complex multi-table JOIN query. The bottom pane shows the results of the query, which is a grid of data from the 'Client' and 'Services' tables. A message bar at the bottom indicates the query was executed successfully.

ClientID	FirstName	LastName	AddressID	PhoneNumberID	ClientDomainID	email	ServicesID	Services_Description
1	Anna	Rhynes	2	2	2	arhynes@myspace.com	2	Wedding Planner Website
2	Mindi	Karla	3	3	3	mkarla@dubyastep.com	3	Family Website
3	Kent	Steinfeldt	4	4	4	ksteinfieldt@howtomom.com	4	Online Support Forum
4	Alean	Bagley	6	6	6	abagley@gotapples.com	6	Antique Store Website
5	Masako	Shirah	14	14	14	mshirah@vaticanagain.com	14	Auction Website
6	Edward	Zoll	15	15	15	ezoll@southlandpark.com	15	Music Artist Website
7	Rosanne	Gremillion	16	16	16	rgremillion@pewds.io	16	Arts and Crafts Online Site
8	Bruce	Dalke	18	18	18	bdalke@admin.com	18	Realtor Website
9	Jorge	Baroso	20	20	20	jbaroso@astrosworldchamps.com	20	Beauty Products Online

Query executed successfully.

Description:

This query report shows the website service with the partial client. Since one of Della's Design requirements was to be able to pull up the client and be able to match it to the website service that the client has currently requested. This report will help Della's Design meet this requirement by being able to run a report that will pull up the data of the client and service by ID. It will also show the client first name, last name, and phone number. When it comes to the service, it will match it to the client ID and she will be able to see the service description and the balance that the client owes.

Floyd Brown

Floyd Brown Individual Query Reports

```

SELECT
    FirstName,
    LastName,
    Services_Description,
    BalanceID,
    PaidID,
    Service_Date
FROM
    Client
INNER JOIN
    [Services] ON [Services].ClientID = Client.ClientID

```

% ▾

FirstName	LastName	Services_Description	BalanceID	PaidID	Service_Date
Timothy	Lowry	Spring Sale Advertisment	1	1	2018-04-18
Anna	Rhynes	Wedding Planner Website	2	2	2018-04-06
Mindi	Karls	Family Website	3	3	2017-12-05
Kent	Steinfeldt	Online Support Forum	4	4	2017-10-09
Howard	Setzer	New Company logo	5	5	2017-10-31
Arlean	Bagley	Antique Store Website	6	6	2017-07-17
Norman	Janas	Website Banner	7	7	2017-06-20
Karyl	Mandeville	Weekly Advertisment	8	8	2017-10-15
Emilio	Alphonse	Custom Website Image	9	9	2017-09-27
Liliana	Widman	Custom Design Seasonal Template	10	10	2017-10-23
Leena	Phelps	Revised Company logo	11	11	2016-12-15
Korey	Larock	Newspaper Advertisment	12	12	2017-09-12
Vicki	Coons	Alternate Company logo	13	13	2017-06-12

This was my first query that retrieved all the relevant information for the customer in which we can use to see what customer needs what material they use for service descri

SQLQuery5.sql - C:\KINE\T\MBROWN\ (50) Object Explorer

```
SELECT
    AreaCode.AreaCodeID,
    Number FROM PhoneNumber
INNER JOIN AreaCode ON AreaCode.AreaCodeID = PhoneNumber.AreaCodeID;
```

0 % ▶

Results Messages

	AreaCodeID	Number
1	361	9999999
1	713	6905413
1	281	3308004
1	209	4531289
1	417	7413658
1	503	1973482
1	202	1234567
1	254	3571598
1	330	4826716
0	401	9102973
1	270	5643212
2	501	4683218
3	242	8920156

This was a basic query call to retrieve the phone number so we can call the customer and let them know that there order is ready.

```
Query1.sql - C:\...NVENT\TUTORIALS\2000\  Object Explorer  SQLQuery2.sql - C:\...NVENT\TUTORIALS\2000\  Command  
SELECT  
    FirstName AS 'First Name',  
    LastName AS 'Last Name',  
    StreetAddress AS 'Street Address',  
    City,  
    [State],  
    ZipCode.ZipCodeID,  
    CONCAT('(', AreaCode.AreaCodeID, ')', ' ', Number) AS 'Phone Number',  
    Email, CASE WHEN TempClient = 1 THEN 'Yes' ELSE 'No' END AS 'Temp Client' FROM PhoneNumber  
INNER JOIN AreaCode ON AreaCode.AreaCodeID = PhoneNumber.AreaCodeID  
INNER JOIN Client ON Client.ClientID = PhoneNumber.PhoneNumberID  
INNER JOIN [Address] ON [Address].AddressID = Client.AddressID  
INNER JOIN ZipCode ON ZipCode.ZipCodeID = [Address].ZipCodeID  
WHERE Client.ClientID = dbo.FindName('Vicki', 'Coons');
```

%

Results Messages

First Name	Last Name	Street Address	City	State	ZipCodeID	Phone Number	Email	Temp
Vicki	Coons	8863 Trenton Road	Giffin	GA	30223	(242) 8920156	vcoons@invisibleplane.com	No

Pulls a relevant customer from a select query which combines a phone number
This was a query using a subquery concatting the customers phone number to give a full description of who our customer is and if they are temporary or consistent customer

```

SQLQuery1.sql - C:\KNE\1\mbrown (30)  X Object Explorer   SQLQuery2.sql - C:\KNE\1\mbrown (24)  Complete - Webma_E1\mbrown (60)
SELECT sub4.ClientID, sub4.FirstName, sub4.LastName, StreetAddress, ZipCodeID, sub4.PhoneNumberID, sub4.ClientDomainID,
sub4.ServicesID, sub4.[Services_Description], sub4.GraphicDesignJobID,
sub4.Balance, sub4.Deposit, sub4.Paid, PaidToDate, sub4.Service_Date, sub4.TempClient FROM [Address]
INNER JOIN (
SELECT Client.ClientID, FirstName, LastName, AddressID, PhoneNumberID, ClientDomainID, email, sub3.ServicesID, sub3.[Serv
sub3.Balance, sub3.Deposit, Paid, PaidToDate, sub3.Service_Date, TempClient
FROM Client
INNER JOIN (
SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.GraphicDesignJobID, sub2.Balance, sub2.Deposit,
INNER JOIN (
SELECT sub.ServicesID, sub.[Services_Description], sub.ClientID, sub.GraphicDesignJobID, Balance, Deposit, PaidID, Service
INNER JOIN (
SELECT ServicesID, [Services_Description], ClientID, GraphicDesignJobID, BalanceID, PaidID, Service_Date
FROM [Services] WHERE [Services].GraphicDesignJobID IS NOT NULL ) sub ON sub.BalanceID = [Balance].BalanceID ) sub2
ON sub2.PaidID = Paid.PaidID ) sub3
ON sub3.ClientID = Client.ClientID ) sub4
ON sub4.AddressID = [Address].AddressID

```

00 % < >

Results Messages

	ClientID	FirstName	LastName	StreetAddress	ZipCodeID	PhoneNumberID	ClientDomainID	email	ServicesID	Ser
1	1	Timothy	Lowry	123 8th St.	32904	1	1	tlowry@website.com	1	Sp
2	5	Howard	Setzer	44 Shirley Ave.	60185	5	5	hsetzer@got.com	5	Ne
3	7	Norman	Janas	212 Randall Mill St.	60008	7	7	njanas@texaslove.com	7	We
4	8	Karlyn	Mandeville	10 Indian Spring Rd.	30040	8	8	kmandeville@thealamofosho.com	8	We
5	9	Emilio	Alphonse	63 Wood Ave.	43068	9	9	ealphonese@anyone.com	9	Cu
6	10	Liliana	Widman	539 SE. Longfellow Street	11432	10	10	lwidman@doublepumpshotgun.com	10	Cu
7	11	Leena	Phelps	8 William St.	14850	11	11	lphelps@theydontunderstand.com	11	Re
8	12	Korey	Larock	9855 Van Dyke Rd.	32958	12	12	klarock@batlink.com	12	Ne
9	13	Vicki	Coons	8863 Trenton Road	30223	13	13	vcoons@invisibleplane.com	13	Alt
10	17	Cara	Pickford	7142 Mechanic Street	29445	17	17	cpickford@default.com	17	Cu
11	19	Max	Waters	227 Broad Street	33594	19	19	mwaters@supersad.com	19	Re

This is select query for multiple tables to pull all relevant information that brings up all the subtotals and totals which uses a sub query.

SQLQuery3.sql - C:\RNET\fmbrown (56)* X Object Explorer SQLQuery2.sql - C:\RNET\fmbrown (54)* Complete - Webma...E:\fmbrown (60)

```

SELECT CONCAT(sub8.FirstName, ' ', sub8.LastName) AS 'Name', CONCAT(sub8.StreetAddress, ' ', sub8.City, ' ', sub8[St
    CONCAT('(', sub8.AreaCodeID, ')', ' ', Number) AS 'Phone Number', sub8.ClientDomain AS 'Client Domain',
    sub8.email, sub8.[Services_Description] AS 'Service', WebsiteJob AS 'Job Description', sub8.Balance, sub8.Deposit, sub8
    PaidToDate, sub8.Service_Date, CASE WHEN sub8.TempClient = 1 THEN 'Yes' ELSE 'No' END AS 'Temp Client' FROM WebsiteJob
    INNER JOIN (
        SELECT sub7.ClientID, sub7.FirstName, sub7.LastName, sub7.StreetAddress, sub7.City, sub7.[State], sub7.ZipCodeID, AreaC
        sub7.email, sub7.ServicesID, sub7.[Services_Description], sub7.WebsiteJobID, sub7.Balance, sub7.Deposit, sub7.Paid,
        PaidToDate, sub7.Service_Date, sub7.TempClient FROM ClientDomain
    INNER JOIN (
        SELECT sub6.ClientID, sub6.FirstName, sub6.LastName, sub6.StreetAddress, sub6.City, sub6.[State], sub6.ZipCodeID, AreaC
        sub6.ServicesID, sub6.[Services_Description], sub6.WebsiteJobID, sub6.Balance, sub6.Deposit, sub6.Paid,
        PaidToDate, sub6.Service_Date, sub6.TempClient FROM PhoneNumber
    INNER JOIN (
        SELECT sub5.ClientID, sub5.FirstName, sub5.LastName, sub5.StreetAddress, City, [State], ZipCode.ZipCodeID, sub5.PhoneNu
        sub5.ServicesID, sub5.[Services_Description], sub5.WebsiteJobID, sub5.Balance, sub5.Deposit, sub5.Paid,
        PaidToDate, sub5.Service_Date, sub5.TempClient FROM ZipCode
    INNER JOIN (
        SELECT sub4.ClientID, sub4.FirstName, sub4.LastName, StreetAddress, ZipCodeID, sub4.PhoneNumberID, sub4.ClientDomainID
        sub4.ServicesID, sub4.[Services_Description], sub4.WebsiteJobID
    
```

100 % 4

Results Messages

Phone Number	Client Domain	email	Service	Job Description	Balance	Deposit	Paid
1 713) 6905413	myspace.com	ahynes@myspace.com	Wedding Planner Website	Medium Personal (6-10 pages)	1575.00	315.00	600.00
2 281) 3308004	dubyastep.com	mikals@dubyastep.com	Family Website	Small Personal (1-5 pages)	950.00	190.00	725.00
3 209) 4531289	howtomom.com	ksteinfield@howtomom.com	Online Support Forum	Forum Website	2550.00	510.00	100.00
4 503) 1973482	gotapples.com	abagley@gotapples.com	Antique Store Website	Medium Business	2125.00	425.00	75.00
5 530) 7841354	vaticanagain.com	mshirish@vaticanagain.com	Auction Website	Auction Website	3825.00	765.00	800.00
6 410) 2214563	southlandpark.com	ezoll@southlandpark.com	Music Artist Website	Music Website	1350.00	270.00	250.00
7 510) 5569520	pewds.io	rgremillion@pewds.io	Arts and Crafts Online Store	Large Online Store	5825.00	1165.00	150.00
8 416) 8742351	admin.com	bdalke@admin.com	Realtor Website	Large Business	3125.00	625.00	300.00
9 225) 3146795	strosworldchamps.com	jbaroso@strosworldchamps.com	Beauty Products Online Store	Medium Online Store	4725.00	945.00	1750.00

This is the query for combining balance and deposit for the services

The screenshot shows a SQL Server Management Studio window. At the top, there are three tabs: 'SQLQuery4.sql - C:\RNET\fmbrown (59)', 'Object Explorer', and 'SQLQuery2.sql - C:\RNET\fmbrown (56)'. Below the tabs is a code editor containing a SELECT statement:

```
SELECT [Services].ServicesID, [Services].[Services_Description] AS 'Service', Addons.Addon, (Addons.Price * ServicesID) AS 'Addon Total'  
FROM Addons INNER JOIN ([Services] INNER JOIN Service2Addon ON [Services].ServicesID = Service2Addon.ServiceID)  
ON Addons.AddonID = Service2Addon.AddonID ORDER BY ServicesID;
```

Below the code editor is a results grid titled 'Results'. The grid has four columns: 'ServicesID', 'Service', 'Addon', and 'Addon Total'. The data is as follows:

	ServicesID	Service	Addon	Addon Total
1	2	Wedding Planner Website	Unique Theme	250.00
2	2	Wedding Planner Website	Complete New Content Package	500.00
3	2	Wedding Planner Website	SEO	250.00
4	2	Wedding Planner Website	Custom Slideshow	75.00
5	3	Family Website	Content Editing	50.00
6	3	Family Website	Unique Theme	250.00
7	3	Family Website	Customized Font	200.00
8	3	Family Website	Copy Text	150.00
9	4	Online Support Forum	SEO	500.00
10	4	Online Support Forum	Accessibility Features	50.00
11	4	Online Support Forum	Security Features	200.00
12	4	Online Support Forum	Fully Responsive CSS	300.00
13	4	Online Support Forum	Unique Theme	250.00

At the bottom of the results grid, a message says 'Query executed successfully.' To the right, it shows the session information: 'COT-CIS3365-15 (14.0 RTM) | COUGARNET\fmbrown (59)'.

This is my function query that calculates add ons

IL-Hwan Kim (Andy)

Complete Client

Results in all client information being shown. Decifiers between a temp client and actual client as well list all information associated with that client. She can use this query to see how many clients she has as well as potential clients.

C:\COT-CIS3365-15\cougarinet\harnold (69) - Remote Desktop Connection

ClientPhone.sql -- RNET\harnold (69) Function - FindAddl..NET\harnold (67) ClientLogin.sql -- RNET\harnold (64)

```

SELECT
    FirstName AS "First Name",
    LastName AS "Last Name",
    StreetAddress AS "Street Address",
    City,
    [State],
    ZipCode.ZipCodeID,
    CONCAT('(', AreaCode.AreaCodeID, ')', ' ', Number) AS "Phone Number",
    Email, CASE WHEN TempClient = 1 THEN "Yes" ELSE "No" END AS "Temp Client" FROM PhoneNumber
    INNER JOIN AreaCode ON AreaCode.AreaCodeID = PhoneNumber.AreaCodeID
    INNER JOIN Client ON Client.ClientID = PhoneNumber.PhoneNumberID
    INNER JOIN [Address] ON [Address].AddressID = Client.AddressID
    INNER JOIN ZipCode ON ZipCode.ZipCodeID = [Address].ZipCodeID

```

Results

	First Name	Last Name	Street Address	City	State	ZipCodeID	Phone Number	Email	Temp Client
1	Timothy	Lovry	123 Bth St.	Melbourne	FL	32904	(011) 9999999	tlowry@website.com	No
2	Anna	Rhynes	71 Pilgrim Avenue	Chevy Chase	MD	20815	(011) 6905413	arhynes@myspace.com	No
3	Mindi	Karla	70 Bowman St.	South Windsor	CT	60716	(011) 3300004	mkarla@dubyatop.com	No
4	Kent	Stenfield	4 Goldfield Rd.	Honolulu	HI	96815	(010) 4531289	kstenfield@howtonom.com	Yes
5	Howard	Setzer	44 Shately Ave.	West Chicago	IL	60185	(417) 7413658	hsetzer@got.com	No
6	Arlan	Bagley	514 S. Magnolia St.	Orlando	FL	32806	(503) 1973482	abagley@gotapple.com	No
7	Norman	Janas	212 Randall Mill St.	Rolling Meadows	IL	60008	(012) 1234567	rjanas@hexuslove.com	No
8	Kaylyn	Mandeville	10 Indian Spring Rd.	Gumming	GA	30040	(254) 3571598	kmmandeville@thealamofosho.com	No
9	Emile	Aphonse	63 Wood Ave.	Reynoldsburg	OH	43068	(330) 4826716	ewphonese@onyme.com	No
10	Ullana	Widman	539 SE Longfellow Street	Jamaica	NY	11432	(401) 9102973	lwidman@doublepumpshotgun.com	Yes
11	Leonna	Phelps	8 William St.	Bhaca	NY	14850	(270) 5643212	lp Phelps@theydontunderstand.com	No
12	Kerry	Larock	9855 Van Dyke Rd.	Sebastian	FL	32958	(501) 4683218	klarock@batnik.com	No
13	Vicki	Coons	8863 Trenton Road	Geffin	GA	30223	(242) 8520156	vcoons@invisibleplane.com	No
14	Meredith	Shurah	#71 Mainave Rd.	Chenault	LA	84310	(619) 7861544	mshurah@bluebeamans.com	Yes

Query executed successfully.

COT-CIS3365-15 (14.0 RTM) | COUGARNET\harnold (69) | Demo

Function - FindRequirement

This function associates requirements with either graphic design or website design job to the clientID and service description. As shown below, the requirement “new logo” associates with graphic design job to the clientID 11 and the description of that service. Dellas can use this function to filter services done by the certain requirement of a specific project.

Function - FindRe...GARNET\iakim (57) ↗ X

```
SELECT [Services].Services_Description, [Services].ClientID, [Services].WebsiteJobID, [Services].GraphicDesignJobID, Requirements FROM [Services]
INNER JOIN Requirements ON [Services].ServicesID = Requirements.ServicesID
WHERE Requirements.RequirementsID = dbo.FindRequirement('New Logo')
```

100 % ⏪

Results Messages

	Services_Description	ClientID	WebsiteJobID	GraphicDesignJobID	Requirements
1	Revised Company logo	11	NULL	3	New Logo

Query executed successfully. | COT-CIS3365-15 (14.0 RTM) | COUGARNET\iakim (57)

GraphicDesignService with BALANCE AND PAID

This query shows when the client made a payment towards a balance and when the service began. Della can use this to determine which clients have paid either deposit or full amount towards balance in association with service date to paid date.

Cot-CIS3365-15.cougar.netuh.edu - Remote Desktop Connection

Database Execute Debug File View Insert Tools Options Help

GraphicDesignService...NET\bharnold (73) > GraphicDesignService...NET\bharnold (72) GraphicDesignService...NET\bharnold (71) Function - FindDo...NET\bharnold (70)

```

SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.GraphicDesignJobID, sub2.Balance, sub2.Deposit, Paid, PaidToDate, sub2.Service_Date
INNER JOIN (
    SELECT sub.ServicesID, sub.[Services_Description], sub.ClientID, sub.GraphicDesignJobID, Balance, Deposit, PaidID, Service_Date
    FROM [Services]
    WHERE [Services].GraphicDesignJobID IS NOT NULL
) sub ON sub.BalanceID = [Balance].BalanceID
ON sub2.PaidID = Paid.PaidID

```

Results Messages

ServicesID	Services_Description	ClientID	GraphicDesignJobID	Balance	Deposit	Paid	PaidToDate	Service_Date
1	Spring Sale Advertisement	1	1	600.00	120.00	100.00	2018-04-09	2018-04-18
2	New Company logo	5	5	300.00	60.00	1200.00	2018-05-04	2017-10-31
3	Website Banner	7	7	480.00	96.00	350.00	2018-04-14	2017-06-20
4	Weekly Advertisement	8	12	330.00	66.00	200.00	2018-04-10	2017-10-15
5	Custom Website Image	9	2	450.00	90.00	0.00	NULL	2017-09-27
6	Custom Design Seasonal Template	10	10	360.00	72.00	25.00	2018-04-18	2017-10-23
7	Revised Company logo	11	3	450.00	90.00	400.00	2018-04-02	2016-12-15
8	Newspaper Advertisement	12	20	630.00	126.00	2000.00	2018-03-01	2017-09-12
9	Alternate Company logo	13	11	420.00	84.00	30.00	2018-03-10	2017-06-12
10	Custom Design Template	17	17	630.00	126.00	0.00	NULL	2017-01-29
11	Revised Website image	19	9	150.00	30.00	550.00	2018-04-20	2017-06-26

GraphicDesignService w/ Partial Client, Complete Address, Complete Phone Number

This query takes the client information and associates it with services, their domainID, graphic design job id, balance, deposit, paid, and dates tables. This simplifies the previous query by simply applying a filter to less information and associating more ID's.

The screenshot shows a Microsoft SQL Server Management Studio window. The title bar reads "GraphicDesignService with Partial Client and Complete Address and Complete Phone Number and Domain.sql - COT-CIS3365-15.Demo_Database (COUGARNET\iakim (62)) - Microsoft SQL Server Management Studio". The query pane contains a complex multi-table join query. The results pane displays a table with 11 columns: ClientID, FirstName, LastName, StreetAddress, City, State, ZipCodeID, AreaCodeID, Number, ClientDomain, and email. The data includes 11 rows of client information. The status bar at the bottom shows "Query executed successfully." and the session details "COT-CIS3365-15 (14.0 RTM) COUGARNET\iakim (62)".

ClientID	FirstName	LastName	StreetAddress	City	State	ZipCodeID	AreaCodeID	Number	ClientDomain	email	ServicesID	Services_Description
1	Timothy	Lowry	123 6th St.	Melbourne	FL	32904	361	9999999	website.com	tlowry@website.com	1	Spring Sale Advertisement
2	Howard	Setzer	44 Shirley Ave.	West Chicago	IL	60185	417	7413658	got.com	hsetzer@got.com	5	New Company logo
3	Norman	Janas	212 Randall Mill St.	Rolling Meadows	IL	60008	202	1234567	texaslove.com	njanas@texaslove.com	7	Website Banner
4	Katlyn	Mandeville	10 Indian Spring Rd.	Cumming	GA	30040	254	3571598	thealamofosho.com	kmandeville@thealamofosho.com	8	Weekly Advertisement
5	Emilio	Alphonse	63 Wood Ave.	Reynoldsburg	OH	43068	330	4826716	anyone.com	ealphonese@anyone.com	9	Custom Website Image
6	Liliana	Widman	539 SE. Longfellow Street	Jamaica	NY	11432	401	9102973	doublepumpshotgun.com	lwidman@doublepumpshotgun.com	10	Custom Design Seasonal Tie
7	Leena	Phelps	8 William St.	Ithaca	NY	14850	270	5643212	theydontunderstand.com	lphelps@theydontunderstand.com	11	Revised Company logo
8	Korey	Larock	9855 Van Dyke Rd.	Sebastian	FL	32958	501	4683218	batlink.com	klarock@batlink.com	12	Newspaper Advertisement
9	Vicki	Coons	8863 Trenton Road	Griffin	GA	30223	242	8920156	invisibleplane.com	vcoons@invisibleplane.com	13	Alternate Company logo
10	Cara	Pickford	7142 Mechanic Street	Goose Creek	SC	29445	351	5023070	default.com	cpickford@default.com	17	Custom Design Template
11	Max	Walters	227 Broad Street	Valrico	FL	33594	479	8885892	supersad.com	mwaiters@supersad.com	19	Revised Website Image

WebsiteService with Balance and Paid.sql

This query report shows the website service with the balance and what has currently been paid to Della's Design. Currently Della's design uses an excel spreadsheet to keep her client account balance and what currently has been paid, but it is kind of hard to keep switching between spreadsheets in order to get to a certain client. Therefore, we have created this report which she can just run and automatically see the service ID and the description which will match it to the client ID and will give her a balance of how much the current client owes and how much the client has paid up to date. This will provide her a more concrete way to keep track of her financials so that clients can pay her.

WebsiteService wit...GARNET\jakim (59) ↗ X

```

SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.WebsiteJobID, sub2.Balance, sub2.Deposit,
    INNER JOIN (
        SELECT sub.ServicesID, sub.[Services_Description], sub.ClientID, sub.WebsiteJobID, Balance, Deposit, PaidID, Service_Date
        FROM [Services] WHERE [Services].WebsiteJobID IS NOT NULL ) sub ON sub.BalanceID = [Balance].BalanceID ) sub2
    ON sub2.PaidID = Paid.PaidID

```

100 % ←

Results Messages

	ServicesID	Services_Description	ClientID	WebsiteJobID	Balance	Deposit	Paid	PaidToDate	Service_Date
1	2	Wedding Planner Website	2	2	1575.00	315.00	600.00	2018-04-17	2018-04-06
2	3	Family Website	3	1	950.00	190.00	725.00	2018-04-06	2017-12-05
3	4	Online Support Forum	4	9	2550.00	510.00	0.00	NULL	2017-10-09
4	6	Antique Store Website	6	7	2125.00	425.00	75.00	2018-04-05	2017-07-17
5	14	Auction Website	14	16	3825.00	765.00	800.00	2018-03-14	2017-02-13
6	15	Music Artist Website	15	20	1350.00	270.00	250.00	2018-04-03	2017-07-27
7	16	Arts and Crafts Online Store	16	12	5825.00	1165.00	150.00	2018-04-12	2017-02-20
8	18	Realtor Website	18	8	3125.00	625.00	300.00	2018-04-15	2017-02-14
9	20	Beauty Products Online Store	20	11	4725.00	945.00	1750.00	2018-04-15	2018-03-14

Query executed successfully. | COT-CIS3365-15 (14.0 RTM) | COUGARNET\jakim (59) | Demo

Barret Margin

ClientLogin

ClientLogin results in matching the domain with the specific username and password of the client. This will allow Della to relate her clients to their domain with login credentials.

```
ClientLogin.sql -...UGARNET\iakim (57) ↵ X
SELECT
    ClientDomain,
    Username,
    [Password]
FROM ClientDomain
INNER JOIN
    WebmasterLogin ON WebmasterLogin.WebmasterLoginID = ClientDomain.WebmasterLoginID
```

100 % ▾

Results Messages

	ClientDomain	Username	Password
1	website.com	tlowry	password
2	myspace.com	arhyne	12345
3	dubyastep.com	mkarls	totallynotpassword
4	howtomom.com	ksteinfieldt	baby2018
5	got.com	hsetzer	dragon
6	gotapples.com	abagley	timcook
7	texasslove.com	njanas	ilovetexas
8	thealamofo sho.com	kmadenville	rememberthealamo
9	anyme.com	ealphonese	ninja
10	doublepumpshotgun.com	lwidman	fortnite
11	theydontunderstand.com	lphelps	str0nger
12	batlink.com	klarock	darkknight7
13	invisibleplane.com	vcoons	wonderwoman
14	uaticanacain.com	mahirah	iamuicius

Query executed successfully. | COT-CIS3365-15 (14.0 RTM) | COUGARNET\iakim (57)

Function - FindDomain

Filters domain with login credentials based on domain of choice. Della can use this to find a specific domain for a certain client.

Function - FindDo...GARNET\iakim (57) ✎ ×

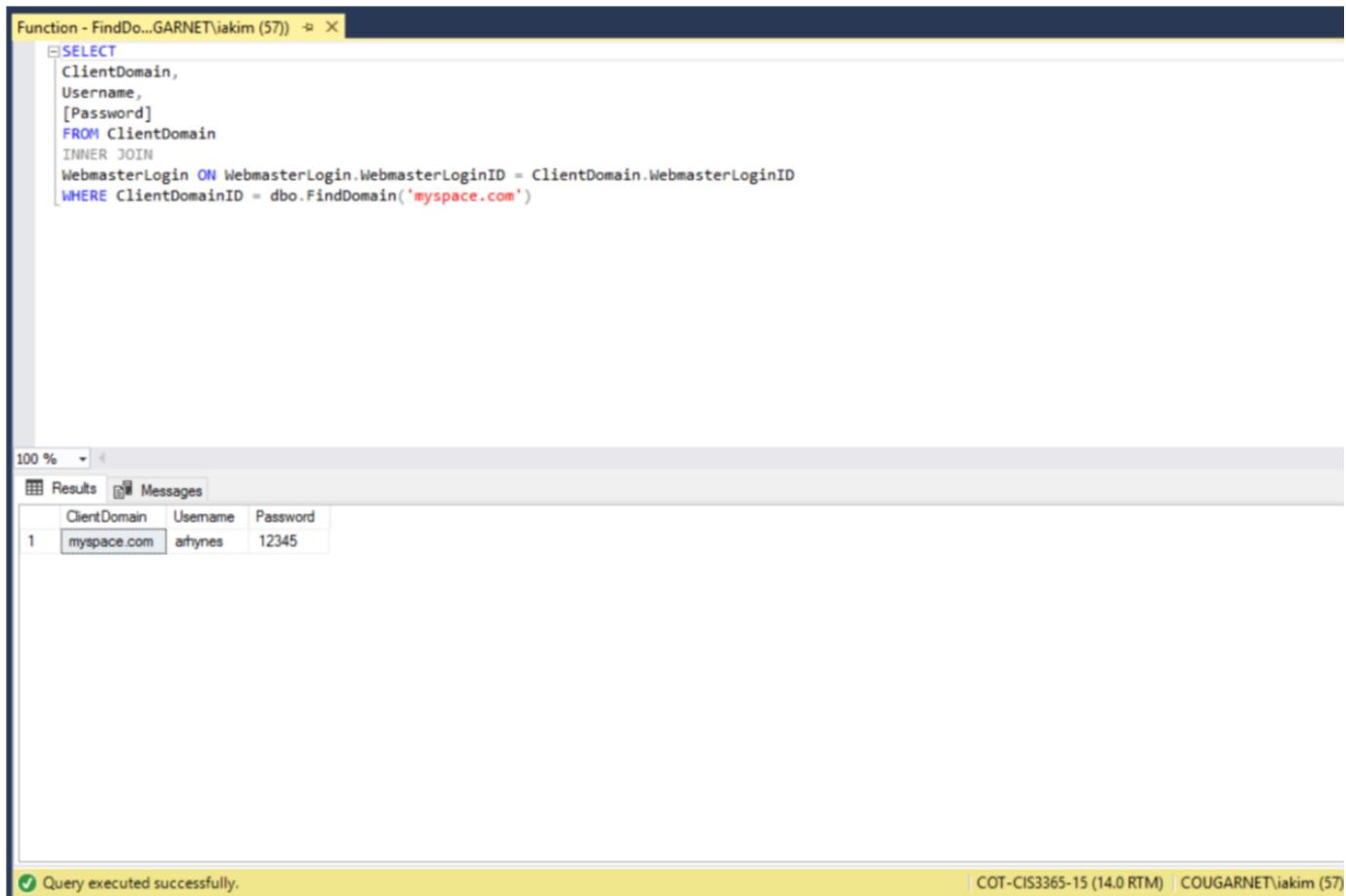
```
SELECT
    ClientDomain,
    Username,
    [Password]
FROM ClientDomain
INNER JOIN
WebmasterLogin ON WebmasterLogin.WebmasterLoginID = ClientDomain.WebmasterLoginID
WHERE ClientDomainID = dbo.FindDomain('myspace.com')
```

100 %

Results Messages

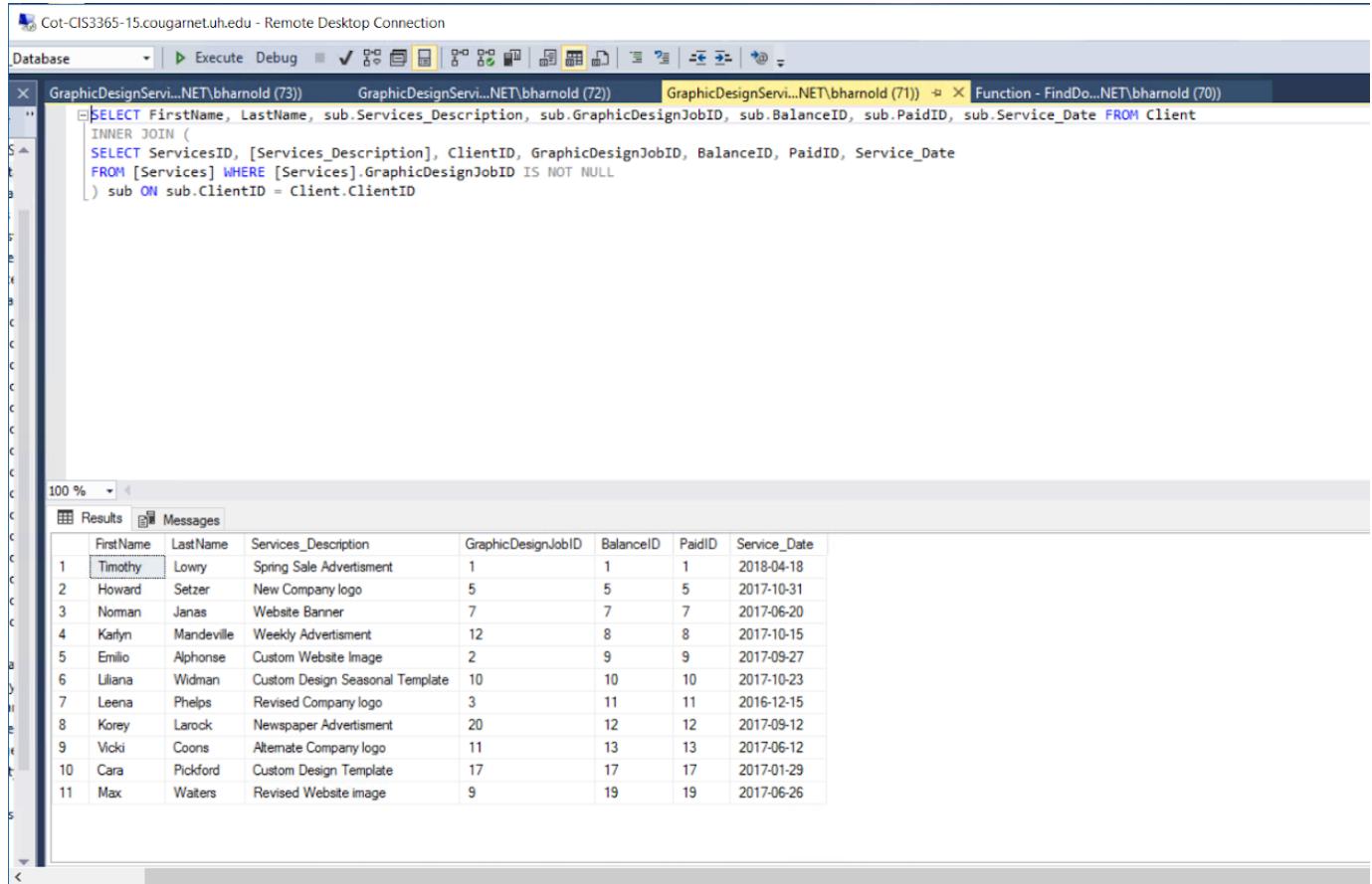
	ClientDomain	Username	Password
1	myspace.com	arhynes	12345

Query executed successfully. | COT-CIS3365-15 (14.0 RTM) | COUGARNET\iakim (57)

A screenshot of SQL Server Management Studio (SSMS) showing a query results window. The title bar says "Function - FindDo...GARNET\iakim (57)". The query window contains a SELECT statement. The results pane shows a table with three columns: ClientDomain, Username, and Password. There is one row returned, with values: ClientDomain is 'myspace.com', Username is 'arhynes', and Password is '12345'. A status bar at the bottom indicates the query was executed successfully.

GraphicDesignService Beginning

This query results association between client information, services, balance, and paid. Based off of client information, it will result in the description of the service, job type, balance, paid, and when the service took place. Della can use this query to see past services she has done.



The screenshot shows a Microsoft SQL Server Management Studio (SSMS) window titled "Cot-CIS3365-15.cougarnet.uh.edu - Remote Desktop Connection". The query pane displays a T-SQL script that joins the "Client" and "Services" tables to find graphic design jobs. The results pane shows a grid of 11 rows with columns: FirstName, LastName, Services_Description, GraphicDesignJobID, BalanceID, PaidID, and Service_Date.

```
SELECT FirstName, LastName, sub.Services_Description, sub.GraphicDesignJobID, sub.BalanceID, sub.PaidID, sub.Service_Date FROM Client
INNER JOIN (
    SELECT ServicesID, [Services_Description], ClientID, GraphicDesignJobID, BalanceID, PaidID, Service_Date
    FROM [Services] WHERE [Services].GraphicDesignJobID IS NOT NULL
) sub ON sub.ClientID = Client.ClientID
```

	FirstName	LastName	Services_Description	GraphicDesignJobID	BalanceID	PaidID	Service_Date
1	Timothy	Lowry	Spring Sale Advertisment	1	1	1	2018-04-18
2	Howard	Setzer	New Company logo	5	5	5	2017-10-31
3	Norman	Janas	Website Banner	7	7	7	2017-06-20
4	Karylyn	Mandeville	Weekly Advertisment	12	8	8	2017-10-15
5	Emilio	Alphonse	Custom Website Image	2	9	9	2017-09-27
6	Liliana	Widman	Custom Design Seasonal Template	10	10	10	2017-10-23
7	Leena	Phelps	Revised Company logo	3	11	11	2016-12-15
8	Korey	Larock	Newspaper Advertisment	20	12	12	2017-09-12
9	Vicki	Coons	Alternate Company logo	11	13	13	2017-06-12
10	Cara	Pickford	Custom Design Template	17	17	17	2017-01-29
11	Max	Waiters	Revised Website image	9	19	19	2017-06-26

Search ZipCode.sql

This report shows a query that can search for a specific zip code. Currently Della's design does not have a system to easily look up a zip code for specific customer address. This report will help her easily search the city, state, and county for a specific zip code, which will save her the time of having to search through her paper documentation.

Search ZipCode.sql...GARNET\iakim (56) ↗ X

```
SELECT ZipCodeID, City, [State], County FROM ZipCode WHERE ZipCodeID = 78413;
```

100 %

Results Messages

	ZipCodeID	City	State	County
1	78413	Corpus Christi	TX	Nueces

Query executed successfully. | COT-CIS3365-15 (14.0 RTM) | COUGARNET\iakim (56) | Demo

The screenshot shows a SQL query window in SSMS. The query is: "SELECT ZipCodeID, City, [State], County FROM ZipCode WHERE ZipCodeID = 78413;". The results pane displays a single row: ZipCodeID 78413, City Corpus Christi, State TX, and County Nueces. A message at the bottom indicates the query was executed successfully.

WebsiteService with Partial Client and Complete Address and Complete Phone Number.sql

This report shows the website service with the partial client and complete address and complete phone number. Della's Design needs an easier way to look up current client's complete address and complete phone numbers in case she needs to contact them for any questions or issues with their current orders. This report will help Della's Design be able to easily get a list of the clients that are currently receiving services and see the phone number with the area code and the email address and contact them as soon as possible.

WebsiteService wit...GARNET\iakim (59) ↗ X

```

FROM [Services] WHERE [Services].WebsiteJobID IS NOT NULL ) sub ON sub.BalanceID = [Balance].BalanceID ) sub2
ON sub2.PaidID = Paid.PaidID ) sub3
ON sub3.ClientID = Client.ClientID ) sub4
ON sub4.AddressID = [Address].AddressID ) sub5
ON sub5.ZipCodeID =ZipCode.ZipCodeID ) sub6
ON sub6.PhoneNumberID = Phonenumber.PhoneNumberID ) sub7
ON sub7.ClientDomainID = ClientDomain.ClientDomainID

```

100 % ▶

Results Messages

	ClientID	FirstName	LastName	StreetAddress	City	State	ZipCodeID	AreaCodeID	Number	ClientDomain	email
1	2	Anna	Rhynes	71 Pilgrim Avenue	Chevy Chase	MD	20815	713	6905413	myspace.com	arhyne
2	3	Mindi	Karls	70 Bowman St.	South Windsor	CT	6074	281	3308004	dubyastep.com	mkarls
3	4	Kent	Steinfeldt	4 Goldfield Rd.	Honolulu	HI	96815	209	4531289	howtomom.com	ksteinf
4	6	Arlean	Bagley	514 S. Magnolia St.	Orlando	FL	32806	503	1973482	gotapples.com	abagle
5	14	Masako	Shirah	473 Harvey Rd.	Glendale	AZ	85302	630	7841354	vaticanagain.com	mshiral
6	15	Edward	Zoll	275 Arnold Court	Soddy Daisy	TN	37379	410	2214563	southlandpark.com	ezoll@
7	16	Rosanne	Gremillion	770 S. Willow Avenue	Navarre	FL	32566	610	5569520	pewds.io	rgremill
8	18	Bruce	Dalke	77 S. Prospect Dr.	Downers Grove	IL	60515	416	8742351	admin.com	bdalke
9	20	Jorge	Barroso	1 North Second Drive	Concord	NH	3301	225	3146795	strosworldchamps.com	jbarros

Query executed successfully. | COT-CIS3365-15 (14.0 RTM) | COUGARNET\iakim (59) | Demo

Vivian Salako

Function - FindService

This function uses service description to list all services done under that specific description. Also list the addon, price, and how many times that service description was done. Della can use this function to filter certain descriptions of a service.

Function - FindSer...GARNET\iakim (57) ↗ X

```
SELECT [Services].ServicesID, [Services].[Services_Description], Addons.Addon, Addons.Price, Service2Addon.Quantity
FROM Addons INNER JOIN ([Services] INNER JOIN Service2Addon ON [Services].ServicesID = Service2Addon.ServiceID)
ON Addons.AddonID = Service2Addon.AddonID
WHERE [Services].ServicesID = dbo.FindService('Family Website') ORDER BY ServicesID;
```

100 % < Results Messages

	ServicesID	Services_Description	Addon	Price	Quantity
1	3	Family Website	Content Editing	50.00	1
2	3	Family Website	Unique Theme	250.00	1
3	3	Family Website	Customized Font	200.00	1
4	3	Family Website	Copy Text	50.00	3

Query executed successfully. COT-CIS3365-15 (14.0 RTM) | COUGARNET\iakim (57)

GraphicDesignService w/ Partial Client

Results in client first and last name associated with the IDs of everything else. Purpose of this query is provide less information and give Della only the information she needs.

Cot-CIS3365-15.cougar.net.uh.edu - Remote Desktop Connection

InService with Partial Client.sql - COT-CIS3365-15.Demo_Database (COUGARNET\bharnold (78)) - Microsoft SQL Server Management Studio

Query Project Debug Tools Window Help

Database Execute Debug Generic Debugger Services

Search AreaCodes...NET\bharnold (81) Search AddonUsed...NET\bharnold (80) GraphicDesignServ...NET\bharnold (79) GraphicDesignServ...NET\bharnold (78)

```

SELECT Client.ClientID, FirstName, LastName, AddressID, PhoneNumberID, ClientDomainID, email, sub3.ServicesID, sub3.[Services_Description], sub3.Gra
sub3.Balance, sub3.Deposit, Paid, PaidToDate, sub3.Service_Date, TempClient
FROM Client
INNER JOIN (
    SELECT sub2.ServicesID, sub2.[Services_Description], sub2.ClientID, sub2.GraphicDesignJobID, sub2.Balance, sub2.Deposit, Paid, PaidToDate, sub2.Serv
    INNER JOIN (
        SELECT sub.ServicesID, sub.[Services_Description], sub.ClientID, sub.GraphicDesignJobID, Balance, Deposit, PaidID, Service_Date FROM Balance
        INNER JOIN (
            SELECT ServicesID, [Services_Description], ClientID, GraphicDesignJobID, BalanceID, PaidID, Service_Date
            FROM [Services] WHERE [Services].GraphicDesignJobID IS NOT NULL ) sub ON sub.BalanceID = [Balance].BalanceID ) sub2
        ON sub2.PaidID = Paid.PaidID ) sub3
    ON sub3.ClientID = Client.ClientID;

```

Results Messages

ClientID	FirstName	LastName	AddressID	PhoneNumberID	ClientDomainID	email	ServicesID	Services_Description	GraphicDesignJobID	Balance
1	Timothy	Lowry	1	1	1	tlowry@website.com	1	Spring Sale Advertisment	1	600.00
2	Howard	Setzer	5	5	5	hsetzer@got.com	5	New Company logo	5	300.00
3	Norman	Janas	7	7	7	njanas@texaslove.com	7	Website Banner	7	480.00
4	Karlyn	Mandeville	8	8	8	kmandeville@thealamofosho.com	8	Weekly Advertisment	12	330.00
5	Emilio	Alphonse	9	9	9	ealphonese@anyme.com	9	Custom Website Image	2	450.00
6	Liliana	Widman	10	10	10	lwidman@doublepumpshotgun.com	10	Custom Design Seasonal Template	10	360.00
7	Leena	Phelps	11	11	11	lphelps@theydontunderstand.com	11	Revised Company logo	3	450.00
8	Korey	Larock	12	12	12	klarock@attlink.com	12	Newspaper Advertisement	20	630.00
9	Vicki	Coons	13	13	13	vcoons@invisibleplane.com	13	Alternate Company logo	11	420.00
10	Cara	Pickford	17	17	17	cpickford@default.com	17	Custom Design Template	17	630.00

ServiceWithAddons.sql

This query report outputs the current service ID and what addons can be added, as well the price and the quantity. Della's Design can use this report at any time that either an employee or a customer has questions of what addons can be added to the services and their price. This way she does not have to only have paper prints of it but as well a digital way in her computer which she can just run and show the customer on the spot.

ServiceWithAddons...ARNET\iakim (56) X

```
SELECT Service2Addon.ServiceID, Addons.Addon, Addons.Price, Service2Addon.Quantity
FROM Addons INNER JOIN Service2Addon ON Addons.AddonID = Service2Addon.AddonID;
```

100 % <

Results Messages

ServiceID	Addon	Price	Quantity
1	2 Unique Theme	250.00	1
2	2 Complete New Content Package	100.00	5
3	2 SEO	50.00	5
4	2 Custom Slideshow	75.00	1
5	3 Content Editing	50.00	1
6	3 Unique Theme	250.00	1
7	3 Customized Font	200.00	1
8	3 Copy Text	50.00	3
9	4 SEO	50.00	10
10	4 Accessibility Features	50.00	1
11	4 Security Features	200.00	1
12	4 Fully Responsive CSS	300.00	1
13	4 Unique Theme	250.00	1
14	6 SEO	50.00	1
15	6 Copy Text	50.00	5
16	6 Security Features	200.00	1
17	6 Fully Responsive CSS	300.00	1
18	6 Facebook Plugin	50.00	1
19	6 PayPal API Plugin	150.00	1
20	6 Shopping Cart Function	125.00	1
21	14 SEO	50.00	20
22	14 Copy Text	50.00	10
23	14 Security Features	200.00	1
24	14 Fully Responsive CSS	300.00	1
25	14 Facebook Plugin	50.00	1
26	14 PayPal API Plugin	150.00	1
27	14 Shopping Cart Function	125.00	1

Query executed successfully. | COT-CIS3365-15 (14.0 RTM) | COUGARNET\iakim (56) | Demo_

WebsiteJobServices.sql

This query report outputs the Website Job Services that Della's design currently offers to customers. Even though Della's design has a system to see, what services are offered she wanted to do it digitally. Therefore, this report will allow her to see how many services are currently offered by Della's Design by outputting the services ID, client ID, service description, website job, and the price. This report also helps her see what she currently has whenever she wants to add more.

WebsiteJobService...GARNET\iakim (59) X

```
SELECT [Services].ServicesID, [Services].ClientID, [Services].Service_Date, [Services].[Services_Description], WebsiteJob.WebsiteJobID
FROM WebsiteJob INNER JOIN [Services] ON WebsiteJob.WebsiteJobID = [Services].WebsiteJobID;
```

100 %

	ServicesID	ClientID	Service_Date	Services_Description	WebsiteJob	Price
1	2	2	2018-04-06	Wedding Planner Website	Medium Personal (6-10 pages)	500.00
2	3	3	2017-12-05	Family Website	Small Personal (1-5 pages)	300.00
3	4	4	2017-10-09	Online Support Forum	Forum Website	1250.00
4	6	6	2017-07-17	Antique Store Website	Medium Business	1000.00
5	14	14	2017-02-13	Auction Website	Auction Website	1500.00
6	15	15	2017-07-27	Music Artist Website	Music Website	500.00
7	16	16	2017-02-20	Arts and Crafts Online Store	Large Online Store	2000.00
8	18	18	2017-02-14	Realtor Website	Large Business	1500.00
9	20	20	2018-03-14	Beauty Products Online Store	Medium Online Store	1500.00

Query executed successfully.

COT-CIS3365-15 (14.0 RTM) | COUGARNET\iakim (59) | Demo

WebsiteService Beginnings.sql

This query reports shows the data in which the service was first open. It also gives the clients first name, last name, service description, website job ID, balance ID, paid ID and of course the service date for when the service was first requested. The current system that Della Design currently has does not give her a way to be able to pull a report of when the service started so she is not able to see how much it is taking. This report will allow her to see the service date of the service for when it was entered into the system, which will give her an idea of how long, is taking to complete the service, which she can then show to the client in case any client has questions.

```
WebsiteService Be...GARNET\iakim (59)) ⇡ X
SELECT FirstName, LastName, sub.Services_Description, sub.WebsiteJobID, sub.BalanceID, sub.PaidID, sub.Service_Da
INNER JOIN (
    SELECT ServicesID, [Services_Description], ClientID, WebsiteJobID, BalanceID, PaidID, Service_Date
    FROM [Services] WHERE [Services].WebsiteJobID IS NOT NULL
)
) sub ON sub.ClientID = Client.ClientID
```

100 %

	FirstName	LastName	Services_Description	WebsiteJobID	BalanceID	PaidID	Service_Date
1	Anna	Rhynes	Wedding Planner Website	2	2	2	2018-04-06
2	Mindi	Karls	Family Website	1	3	3	2017-12-05
3	Kent	Steinfeldt	Online Support Forum	9	4	4	2017-10-09
4	Arlean	Bagley	Antique Store Website	7	6	6	2017-07-17
5	Masako	Shirah	Auction Website	16	14	14	2017-02-13
6	Edward	Zoll	Music Artist Website	20	15	15	2017-07-27
7	Rosanne	Gremillion	Arts and Crafts Online Store	12	16	16	2017-02-20
8	Bruce	Dalke	Realtor Website	8	18	18	2017-02-14
9	Jorge	Baroso	Beauty Products Online Store	11	20	20	2018-03-14

Query executed successfully.

COT-CIS3365-15 (14.0 RTM) | COUGARNET\iakim (59) | Demo