Play SQL:
Learning Database Querying using a Game

Mark Brodie
Department of Computer Science
Simpson College
Indianola, IA 51025
mark.brodie@simpson.edu

Abstract

PlaySQL is an Android app that uses a game to teach relational database querying. The app allows people with no previous programming experience to learn SQL programming in a fun, interactive and self-paced way. The app is very different from any existing SQL apps, none of which adopt a game approach.

The app currently covers the core topics of SQL programming, including Select, Where, Group By, Having, Order By, Cross Join, Inner Join and Outer Join. Each topic is explained using a sequence of queries which the user has to solve. The queries are carefully structured to lead the user from simpler to more complex material. Each topic is followed by a quiz in which the player tries to maximize their score by completing queries as quickly as possible. The player can also embark on "missions" in which they try to survive as long as possible by answering questions of varying difficulty over all topics.

Questions are presented in a variety of formats. Sometimes the player has to drag the correct terms onto blank spaces in the query. Other questions require the player to click on the terms of the query in the correct order, or to correct errors in a query, or to rearrange the query to make it correct. The most challenging format has the query terms fluctuating randomly on the screen; the player has to click on each one when it is correct.

The app has been tested with many students, both with and without SQL experience. Developing the app was very instructive in a number of areas that have impacted my classroom pedagogy, including databases, mobile app development and software engineering.

1 Motivation

As a professor of computer science, I am naturally interested in the use of information technology in education. Our students, called by some the “App Generation” [1] and by others "iGen" [2], spend considerable time using mobile apps and games. I wanted to investigate whether a game could be used to provide instruction at the college-level.

Many web-sites and apps deal with material covered in college classes; e.g. Khan Academy, the works of Shakespeare [3] [4], Ancient Greek [5] [6], and others. Although these sites and apps can be used in a college classroom to enhance the educational experience, they cannot substitute for taking a class in the subject, because they lack the personalized feedback from an expert instructor that face-to-face interaction provides. A game allows someone to learn on their own in a fun way. Games have been used effectively for many elementary and middle schools topics, and there are some games that teach introductory programming, although not at the college level.

2 Database Querying

Querying a relational database seemed like a good topic for an educational game. It is a key information technology, and there is a continual need for people with database skills. Database programming requires minimal background knowledge. The highly-structured form of SQL queries provides a natural scaffolding where a game can build slowly from very simple to more complicated queries, allowing the player to learn at their own pace. I wanted to explore the hypothesis that, using a game, any motivated person could learn the basics of SQL programming without assistance from a teacher.

There are many mobile apps that teach database programming, including [7] [8] [9] [10] [11] [12] [13].and more. Some of them are quite good, allowing the user to interact with a database using an interface in which they can create and execute queries. They include text or video tutorials that teach the user how to create queries. However they don't provide much feedback when the user cannot figure out the query, not do they provide much incentive for a student to keep working on the subject when they encounter topics they find hard to understand. None of the existing SQL apps use a game-based approach.

3 The PlaySQL App

The app I have developed is called PlaySQL. Like Ancient Gaul, it is divided into 3 parts: Study, Quiz, and Launch Missions, conveniently denoted by S, Q and L.

Currently there are 12 study sections, covering the topics of Select, Where, Order By, Functions, Aggregate Functions, Distinct, Group By, Having, Cross Join, Inner Join, Outer Join and Like. Each section has some introductory text explaining the concept being taught, followed by a sequence of questions. Each question displays a small database table and describes the information we would like to extract from the table. As explained further below, the app shows the player how to construct a SQL query that retrieves the desired information.

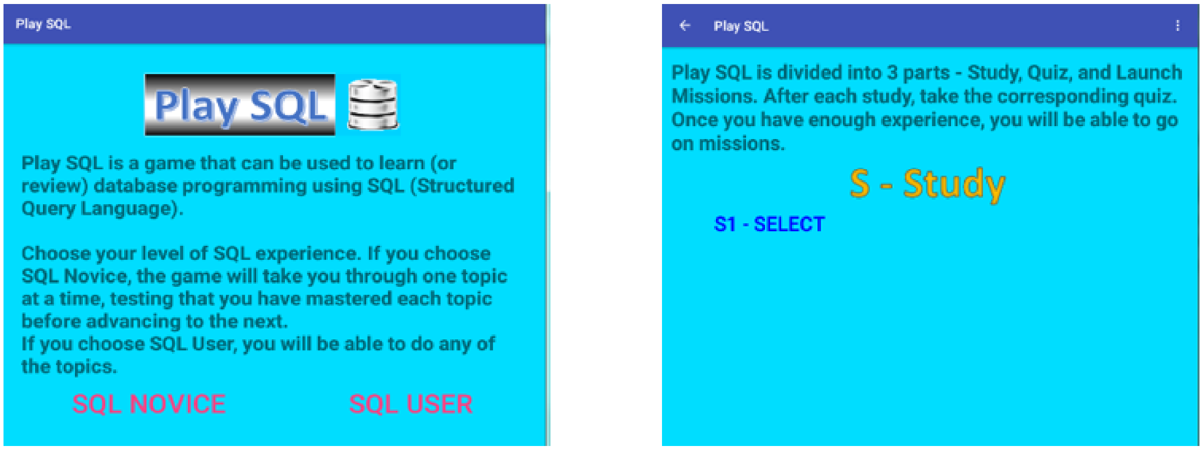
The number of questions varies from section to section, depending on the topic. There are currently about 100 questions over all 12 topics. The questions in each section are carefully ordered so that the player builds on their previous knowledge to move from simple to more difficult questions. If a player gets stuck and cannot solve a problem, they can click a button to see a solution.

Each study section has a quiz, containing randomly ordered questions from the study. The player scores points depending on how quickly they complete each question correctly. The app keeps track of their high-score on each quiz.

After completing studies and quizzes, the player can launch missions. There are currently 3 different missions. The first covers single-table queries (i.e. no joins), the second covers all queries involving two tables (i.e. all the questions on joins), and the third covers all topics.

The player starts with a certain number of lives at the beginning of each mission. A mission consists of a randomly selected set of questions from all the covered topics. Each question has a certain time, depending on its difficulty, in which it must be solved. If the player cannot complete a question before time runs out, they lose a life. A lost life can be regained if the player completes a later question with extra time to spare. The time allowed to complete a question decreases as the player advances further in the mission, thus increasing the challenge as the game proceeds. The mission ends in failure when the player runs out of lives or in success if the player completes all the questions.

The app requires no previous programming experience or knowledge of databases. As illustrated below, each section starts with questions in which it is fairly easy for the player to figure out the correct query. From there each section builds carefully to more sophisticated queries.



Play SQL is a game that can be used to learn (or review) database programming using SQL (Structured Query Language).

Choose your level of SQL experience. If you choose SQL Novice, the game will take you through one topic at a time, testing that you have mastered each topic before advancing to the next.
If you choose SQL User, you will be able to do any of the topics.

Play SQL is divided into 3 parts - Study, Quiz, and Launch Missions. After each study, take the corresponding quiz. Once you have enough experience, you will be able to go on missions.

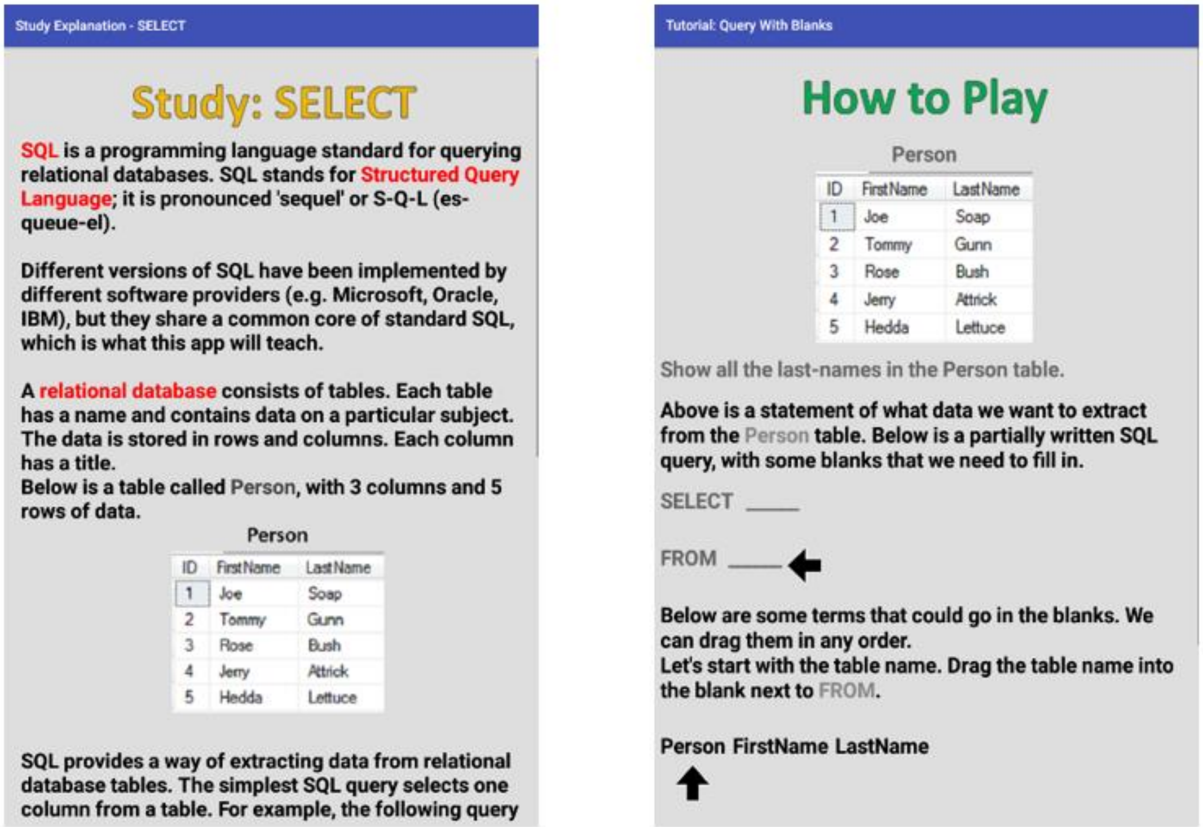
Figure 1: The opening screen (left) and the SQL Novice screen (right).

3.1 The Opening Screen

The app's opening screen is shown in Figure 1 above, on the left. It provides a simple choice between SQL Novice and SQL User. If the user selects SQL Novice, they are taken to the screen shown in Figure 1 on the right. If they select SQL User, they are taken to the screen shown in Figure 9 in section 7.

4 The First Study – SELECT

When the user clicks on S1 – SELECT, they are shown a brief text explanation (shown in Figure 2 below on the left) of what a relational database is and how a simple SELECT statement works. (Note that the text scrolls, so what is shown in the screenshot is not complete.) They are then given a brief tutorial on how to create a simple query by dragging terms into blank spaces (shown in Figure 2 below on the right).



Study: SELECT

SQL is a programming language standard for querying relational databases. SQL stands for Structured Query Language; it is pronounced 'sequel' or S-Q-L (es-queue-el).

Different versions of SQL have been implemented by different software providers (e.g. Microsoft, Oracle, IBM), but they share a common core of standard SQL, which is what this app will teach.

A relational database consists of tables. Each table has a name and contains data on a particular subject. The data is stored in rows and columns. Each column has a title.
Below is a table called Person, with 3 columns and 5 rows of data.

Person

ID	FirstName	LastName
1	Joe	Soap
2	Tommy	Gunn
3	Rose	Bush
4	Jerry	Attrick
5	Hedda	Lettuce

SQL provides a way of extracting data from relational database tables. The simplest SQL query selects one column from a table. For example, the following query

How to Play

Person

ID	FirstName	LastName
1	Joe	Soap
2	Tommy	Gunn
3	Rose	Bush
4	Jerry	Attrick
5	Hedda	Lettuce

Show all the last-names in the Person table.

Above is a statement of what data we want to extract from the Person table. Below is a partially written SQL query, with some blanks that we need to fill in.

SELECT ____

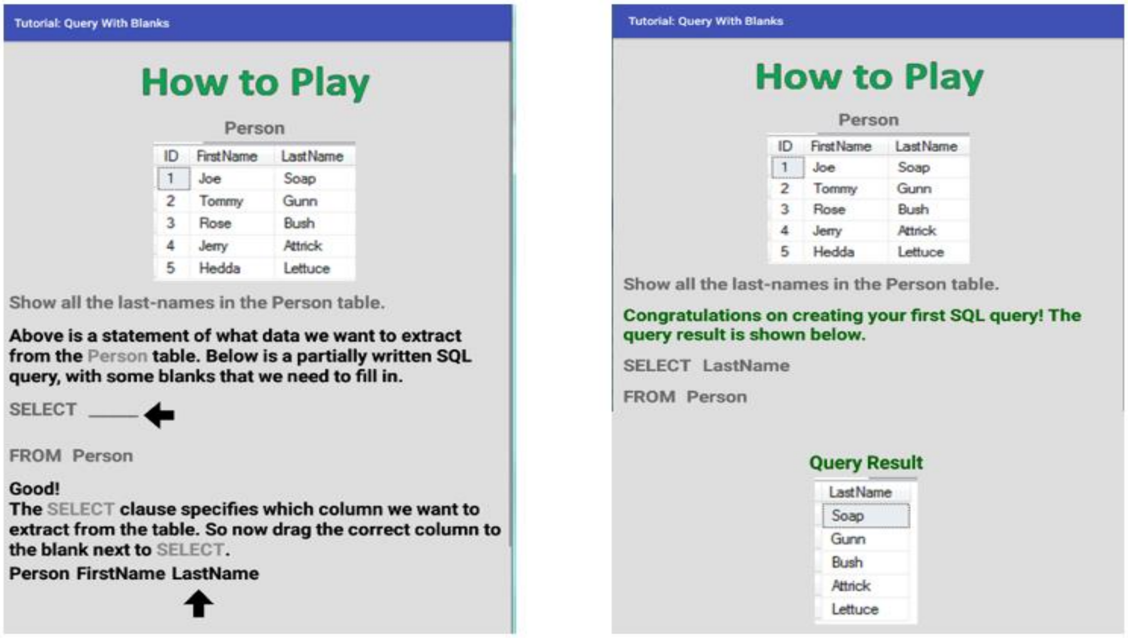
FROM ____ ←

Below are some terms that could go in the blanks. We can drag them in any order.
Let's start with the table name. Drag the table name into the blank next to FROM.

Person FirstName LastName
↑

Figure 2: Some explanatory text (left) and a tutorial on how to make queries (right).

First they drag the table name into the "FROM" clause (shown above on the right). Then they drag the column name into the "SELECT" clause (shown in Figure 3 below on the left). The result on the query is shown (below on the right). That ends the tutorial.



Tutorial: Query With Blanks

How to Play

Person

ID	FirstName	LastName
1	Joe	Soap
2	Tommy	Gunn
3	Rose	Bush
4	Jerry	Attrick
5	Hedda	Lettuce

Show all the last-names in the Person table.

Above is a statement of what data we want to extract from the Person table. Below is a partially written SQL query, with some blanks that we need to fill in.

SELECT ______ 🡐

FROM Person

Good!
The SELECT clause specifies which column we want to extract from the table. So now drag the correct column to the blank next to SELECT.

Person FirstName LastName

🡑

Tutorial: Query With Blanks

How to Play

Person

ID	FirstName	LastName
1	Joe	Soap
2	Tommy	Gunn
3	Rose	Bush
4	Jerry	Attrick
5	Hedda	Lettuce

Show all the last-names in the Person table.

Congratulations on creating your first SQL query! The query result is shown below.

SELECT LastName

FROM Person

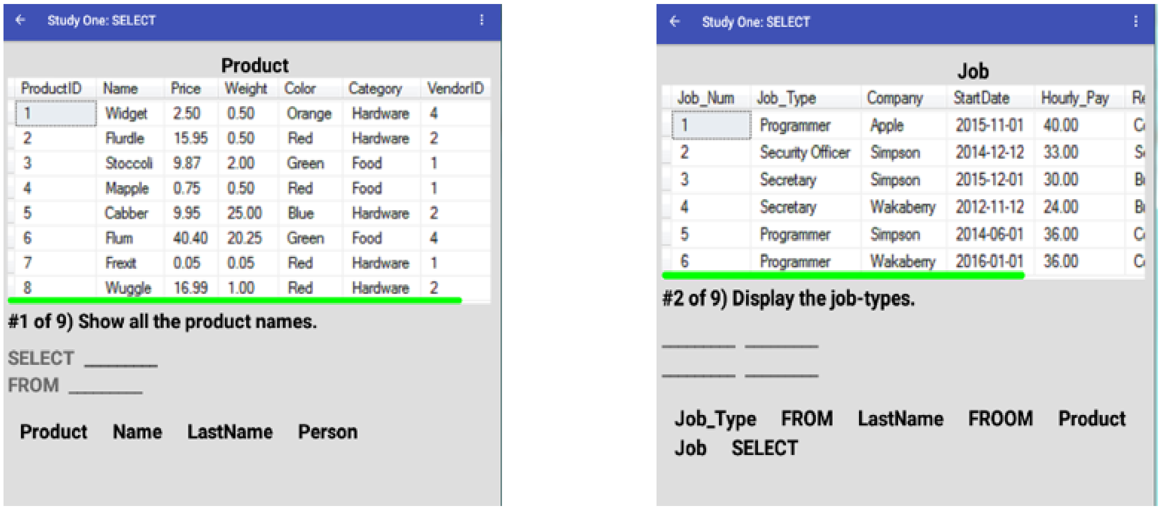
Query Result

LastName
Soap
Gunn
Bush
Attrick
Lettuce

Figure 3: The tutorial continues (left) and completes (right).

4.1 The First Few Questions

After completing the tutorial, the player is given their first question to solve, shown below on the left. They simply have to drag (in any order) the table-name and column-name into the correct blanks to complete the query. The output of the query is then shown. Their second question is shown below on the right. Notice that here they have to drag all four terms to complete the query.



Study One: SELECT

Product

ProductID	Name	Price	Weight	Color	Category	VendorID
1	Widget	2.50	0.50	Orange	Hardware	4
2	Flurdle	15.95	0.50	Red	Hardware	2
3	Stoccoli	9.87	2.00	Green	Food	1
4	Mapple	0.75	0.50	Red	Food	1
5	Cabber	9.95	25.00	Blue	Hardware	2
6	Flum	40.40	20.25	Green	Food	4
7	Frexit	0.05	0.05	Red	Hardware	1
8	Wuggle	16.99	1.00	Red	Hardware	2

#1 of 9) Show all the product names.

SELECT ______
FROM ______

Product Name LastName Person

Study One: SELECT

Job

Job_Num	Job_Type	Company	StartDate	Hourly_Pay	Re
1	Programmer	Apple	2015-11-01	40.00	C
2	Security Officer	Simpson	2014-12-12	33.00	S
3	Secretary	Simpson	2015-12-01	30.00	B
4	Secretary	Wakaberry	2012-11-12	24.00	B
5	Programmer	Simpson	2014-06-01	36.00	C
6	Programmer	Wakaberry	2016-01-01	36.00	C

#2 of 9) Display the job-types.

______ ______
______ ______

Job_Type FROM LastName FROOM Product
Job SELECT

Figure 4: Questions 1 and 2 of the SELECT study.

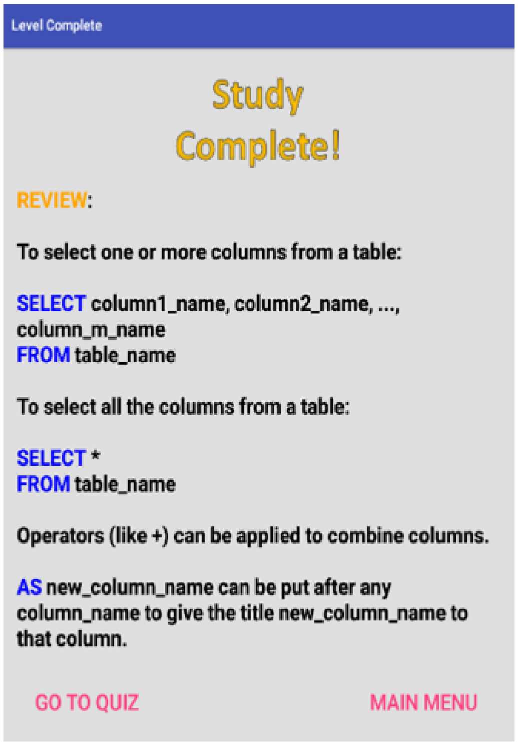
4.2 The Remaining Questions of the First Study

The remaining questions of the SELECT study explain a few more things about the basic SELECT statement. The player is shown how to select more than one column, how to select all the columns using SELECT *, how to name a column in the query output, and how to concatenate two or more strings or string columns (e.g. the first and last-name columns can be concatenated with a space between them to create an output column containing the full-name).

An applause sound is heard each time a question is completed. If a mistake is made (e.g. dragging a term onto the incorrect blank) a raspberry sound is heard. The sound effects can be turned off.

If the player is stuck on any question, they can click a button to see the solution. They can also go back to the previous question to see how it was solved. Since each question builds on the previous one, this is often helpful if they have forgotten how to do something. The player can also at any point skip over the remaining questions and go to the end of the study.

After completing all the study questions, the player is shown a review screen summarizing what they have just learned (shown below on the left). They have the choice of going on to the quiz or going back to the main menu. If they choose the quiz, they see a brief explanation of how the quiz works (shown below on the right).



Study Complete!

REVIEW:

To select one or more columns from a table:

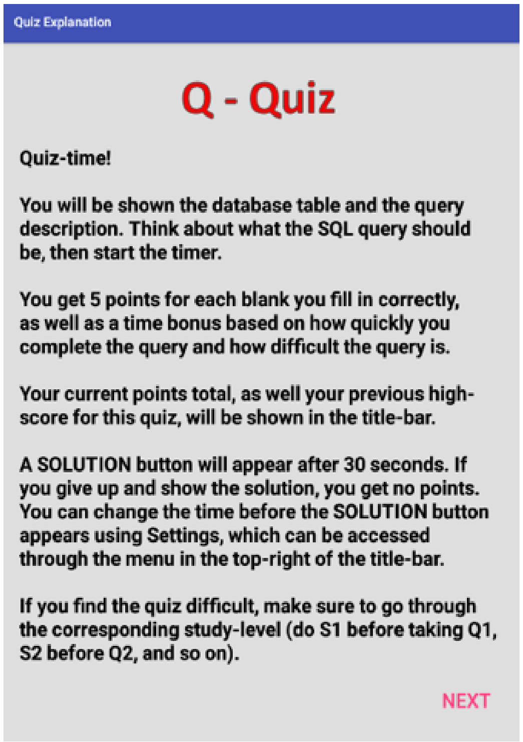
SELECT column1_name, column2_name, ..., column_m_name
FROM table_name

To select all the columns from a table:

SELECT *
FROM table_name

Operators (like +) can be applied to combine columns.

AS new_column_name can be put after any column_name to give the title new_column_name to that column.



Q - Quiz

Quiz-time!

You will be shown the database table and the query description. Think about what the SQL query should be, then start the timer.

You get 5 points for each blank you fill in correctly, as well as a time bonus based on how quickly you complete the query and how difficult the query is.

Your current points total, as well your previous high-score for this quiz, will be shown in the title-bar.

A SOLUTION button will appear after 30 seconds. If you give up and show the solution, you get no points. You can change the time before the SOLUTION button appears using Settings, which can be accessed through the menu in the top-right of the title-bar.

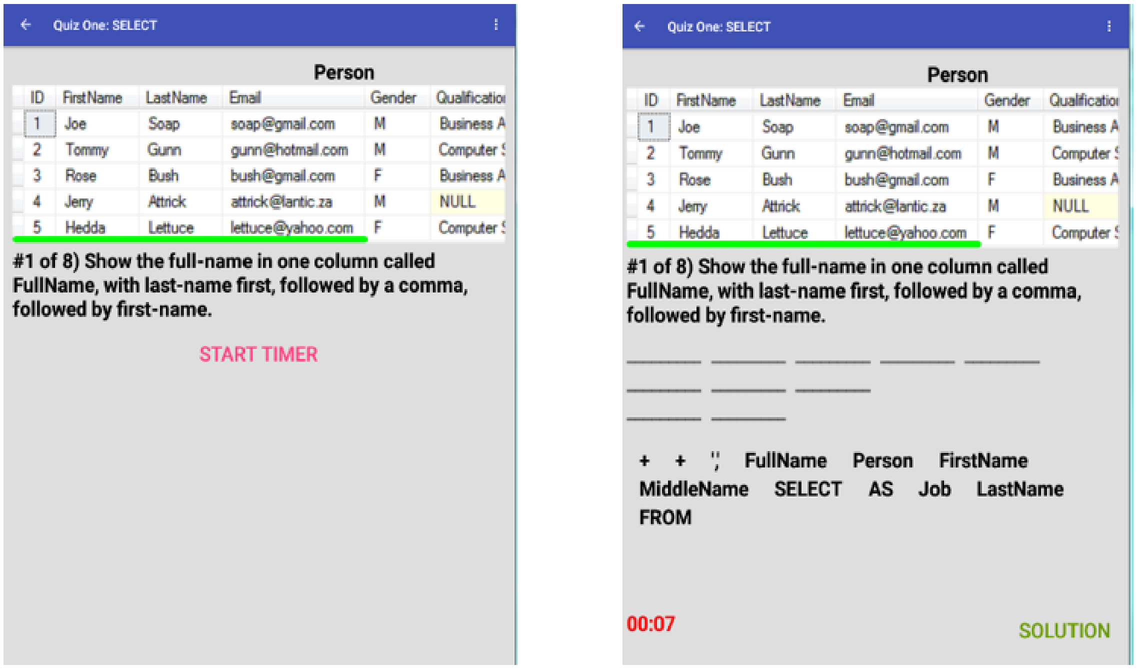
If you find the quiz difficult, make sure to go through the corresponding study-level (do S1 before taking Q1, S2 before Q2, and so on).

Figure 5: Review screen after completing the SELECT study (left).
Explanatory screen for the first quiz (right).

5 The First Quiz

Each quiz consists of the same questions as contained in the study, in random order. However all the blanks need to be filled in for every question in a quiz; no part of the query is provided to the player. Note that sometimes a study will repeat the same question more than once, in order to demonstrate different ways of solving it. In that case, the question will appear only once in the quiz, and the player may solve it in any way that works. Thus a quiz may contain fewer questions than the corresponding study.

An example of the first question in the quiz on SELECT is shown below on the left. The player should think to themselves what the query should be. When they are ready, they start the timer, resulting in the screen on the right (the timer starts at 00:00 obviously – the screen-shot was captured after 7 seconds).



Person

ID	FirstName	LastName	Email	Gender	Qualificatio
1	Joe	Soap	soap@gmail.com	M	Business A
2	Tommy	Gunn	gunn@hotmail.com	M	Computer S
3	Rose	Bush	bush@gmail.com	F	Business A
4	Jerry	Attrick	attrick@lantic.za	M	NULL
5	Hedda	Lettuce	lettuce@yahoo.com	F	Computer S

#1 of 8) Show the full-name in one column called FullName, with last-name first, followed by a comma, followed by first-name.

Person

ID	FirstName	LastName	Email	Gender	Qualificatio
1	Joe	Soap	soap@gmail.com	M	Business A
2	Tommy	Gunn	gunn@hotmail.com	M	Computer S
3	Rose	Bush	bush@gmail.com	F	Business A
4	Jerry	Attrick	attrick@lantic.za	M	NULL
5	Hedda	Lettuce	lettuce@yahoo.com	F	Computer S

#1 of 8) Show the full-name in one column called FullName, with last-name first, followed by a comma, followed by first-name.

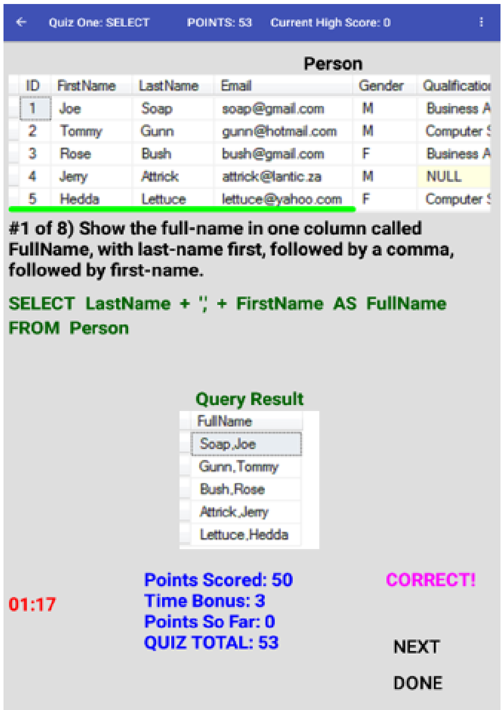
______ ______ ______ ______ ______
______ ______ ______
______ ______

+ + ',' FullName Person FirstName
MiddleName SELECT AS Job LastName
FROM
00:07
SOLUTION

Figure 6: A quiz question before starting the timer (left) and after (right).

The timer continually increments, showing elapsed seconds. The player tries to complete the query as quickly as possible. The points scored depends on how many terms the query contains as well as how long it takes to get the query correct. If they are stuck, at any time they can click the Solution button, but then they get no points for that question.

When a question is solved, the quiz moves on to the next question. The player's score accumulates as they solve questions. Figure 7 below shows the screen after completing the first quiz question (on the left) and the second quiz question (on the right). Note that the second question was shorter. This resulted in a lower number of points scored, but a higher time bonus. The current total and current high-score for the quiz are shown in the title bar.



Person

ID	FirstName	LastName	Email	Gender	Qualificatio
1	Joe	Soap	soap@gmail.com	M	Business A
2	Tommy	Gunn	gunn@hotmail.com	M	Computer
3	Rose	Bush	bush@gmail.com	F	Business A
4	Jerry	Attrick	attrick@lantic.za	M	NULL
5	Hedda	Lettuce	lettuce@yahoo.com	F	Computer

#1 of 8) Show the full-name in one column called FullName, with last-name first, followed by a comma, followed by first-name.

```
SELECT LastName + ',' + FirstName AS FullName
FROM Person
```

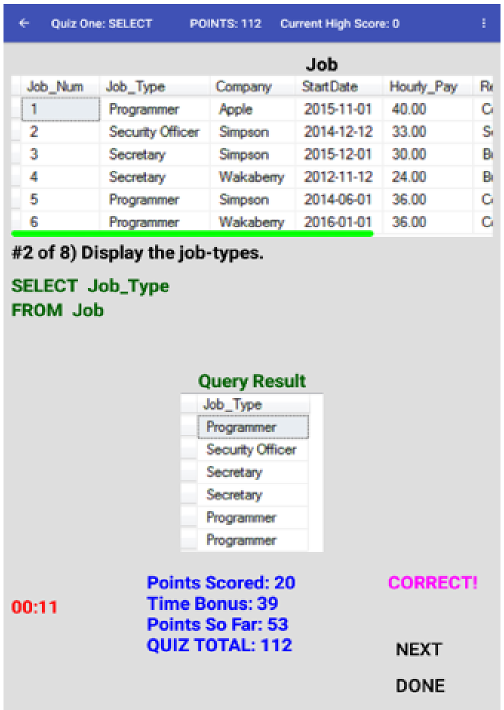
Query Result

FullName
Soap,Joe
Gunn,Tommy
Bush,Rose
Attrick,Jerry
Lettuce,Hedda

01:17

Points Scored: 50
Time Bonus: 3
Points So Far: 0
QUIZ TOTAL: 53

CORRECT!



Job

Job_Num	Job_Type	Company	StartDate	Hourly_Pay	Re
1	Programmer	Apple	2015-11-01	40.00	C
2	Security Officer	Simpson	2014-12-12	33.00	S
3	Secretary	Simpson	2015-12-01	30.00	B
4	Secretary	Wakaberry	2012-11-12	24.00	B
5	Programmer	Simpson	2014-06-01	36.00	C
6	Programmer	Wakaberry	2016-01-01	36.00	C

#2 of 8) Display the job-types.

```
SELECT Job_Type
FROM Job
```

Query Result

Job_Type
Programmer
Security Officer
Secretary
Secretary
Programmer
Programmer

00:11

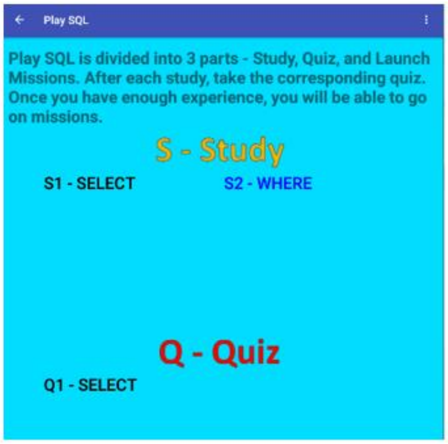
Points Scored: 20
Time Bonus: 39
Points So Far: 53
QUIZ TOTAL: 112

CORRECT!

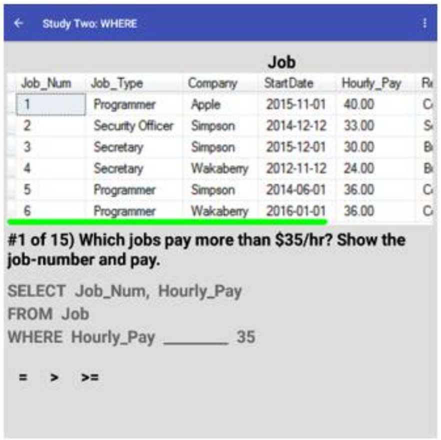
Figure 7: Quiz: question 1 completed (left) and question 2 completed (right).

6 The Second Study - WHERE

After completing the first quiz, the player is returned to the main menu. (Alternatively, they can skip the first quiz after completing the first study and return to the main menu directly.) The main menu now unlocks a second study topic, WHERE, as shown below on the left. The player is shown some explanatory text about WHERE, and then the first question appears, shown below on the right. Note how simple the question is, with just one blank. The choice is between = > and >=. Even a user with no programming experience should be able to figure this out.



Play SQL is divided into 3 parts - Study, Quiz, and Launch Missions. After each study, take the corresponding quiz. Once you have enough experience, you will be able to go on missions.



Job

Job_Num	Job_Type	Company	StartDate	Hourly_Pay	Re
1	Programmer	Apple	2015-11-01	40.00	C
2	Security Officer	Simpson	2014-12-12	33.00	S
3	Secretary	Simpson	2015-12-01	30.00	B
4	Secretary	Wakaberry	2012-11-12	24.00	B
5	Programmer	Simpson	2014-06-01	36.00	C
6	Programmer	Wakaberry	2016-01-01	36.00	C

#1 of 15) Which jobs pay more than $35/hr? Show the job-number and pay.

```
SELECT Job_Num, Hourly_Pay
FROM Job
WHERE Hourly_Pay _______ 35
```

Figure 8: Main Menu shows WHERE (left); first question in WHERE study (right).

The study of WHERE continues through a sequence of questions of increasing difficulty. All the relational operators are introduced, followed by the boolean operators. The player is also shown where order can be switched with both relational and boolean operators, and that relational operators can be used with strings and dates, not just numbers.

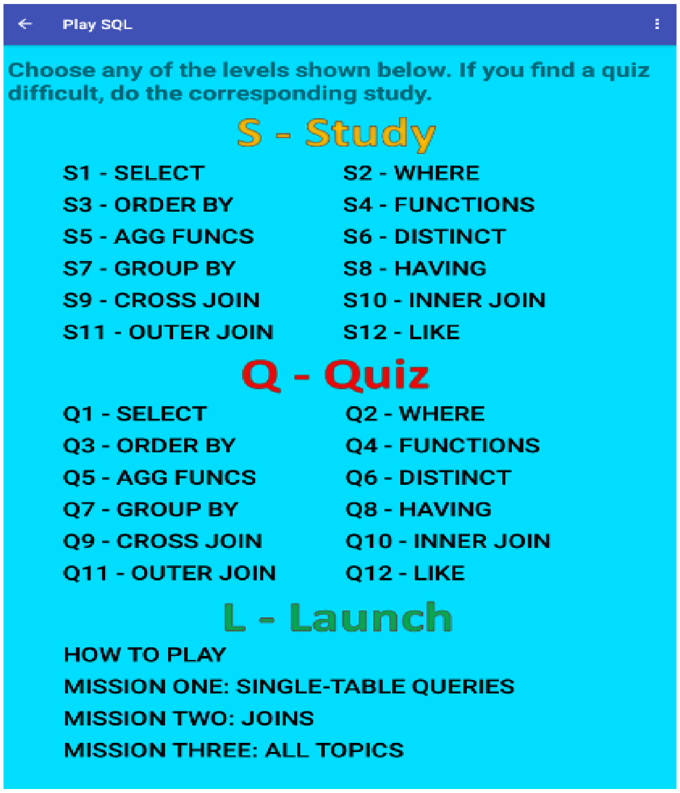
At the end of the WHERE study, the player can take the quiz on WHERE. This works in a similar way as the quiz on SELECT described above.

The player can terminate a quiz at any time. They will still achieve a new high-score if they have exceeded their previous high-score, even if they don't complete all the quiz questions.

7 Other Topics

The user who is a SQL novice should proceed through the app one section at a time. After completing each study section and its quiz, the next section to study is made available. Currently, the topics covered by the app are, in order: Select, Where, Order By, Functions, Aggregate Functions, Distinct, Group By, Having, Order By, Cross Join, Inner Join, Outer Join, Like.

If the user selects SQL User instead of SQL Novice in the very first screen, they can see all the topics, as shown below. They can then do any study or quiz they wish.



Play SQL

Choose any of the levels shown below. If you find a quiz difficult, do the corresponding study.

S - Study

S1 - SELECT
S2 - WHERE
S3 - ORDER BY
S4 - FUNCTIONS
S5 - AGG FUNCS
S6 - DISTINCT
S7 - GROUP BY
S8 - HAVING
S9 - CROSS JOIN
S10 - INNER JOIN
S11 - OUTER JOIN
S12 - LIKE

Q - Quiz

Q1 - SELECT
Q2 - WHERE
Q3 - ORDER BY
Q4 - FUNCTIONS
Q5 - AGG FUNCS
Q6 - DISTINCT
Q7 - GROUP BY
Q8 - HAVING
Q9 - CROSS JOIN
Q10 - INNER JOIN
Q11 - OUTER JOIN
Q12 - LIKE

L - Launch

HOW TO PLAY
MISSION ONE: SINGLE-TABLE QUERIES
MISSION TWO: JOINS
MISSION THREE: ALL TOPICS

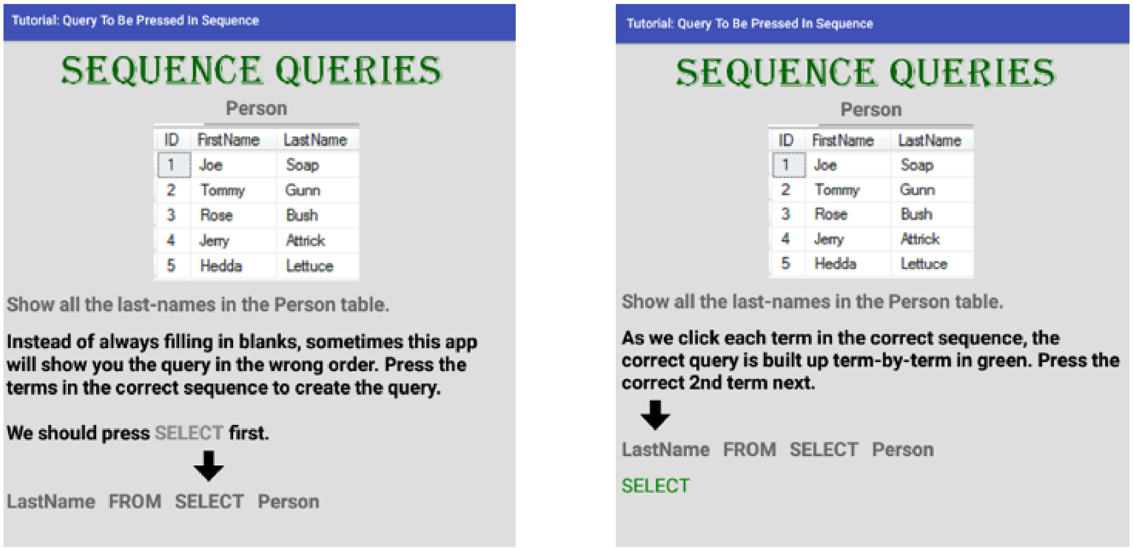
Figure 9: An overview of the entire PlaySQL app.

8 Other Ways of Creating Queries

Dragging tems onto blanks gets a little repetitive after a while. The game provides other ways of creating queries. These are introduced slowly as the player progresses through the game. There are currently 5 different types of queries, described below.

8.1 Sequence Queries

In a sequence query, all the terms of the query are shown, but in the wrong order. The player has to click on the terms in the correct order to create the query. Figure 10 below shows the first two steps of this process. These are screen-shots from the tutorial that explains to the player how to do these queries.



Tutorial: Query To Be Pressed In Sequence

SEQUENCE QUERIES

Person

ID	FirstName	LastName
1	Joe	Soap
2	Tommy	Gunn
3	Rose	Bush
4	Jerry	Attrick
5	Hedda	Lettuce

Show all the last-names in the Person table.

Instead of always filling in blanks, sometimes this app will show you the query in the wrong order. Press the terms in the correct sequence to create the query.

We should press SELECT first.

LastName FROM SELECT Person

Tutorial: Query To Be Pressed In Sequence

SEQUENCE QUERIES

Person

ID	FirstName	LastName
1	Joe	Soap
2	Tommy	Gunn
3	Rose	Bush
4	Jerry	Attrick
5	Hedda	Lettuce

Show all the last-names in the Person table.

As we click each term in the correct sequence, the correct query is built up term-by-term in green. Press the correct 2nd term next.

LastName FROM SELECT Person

SELECT

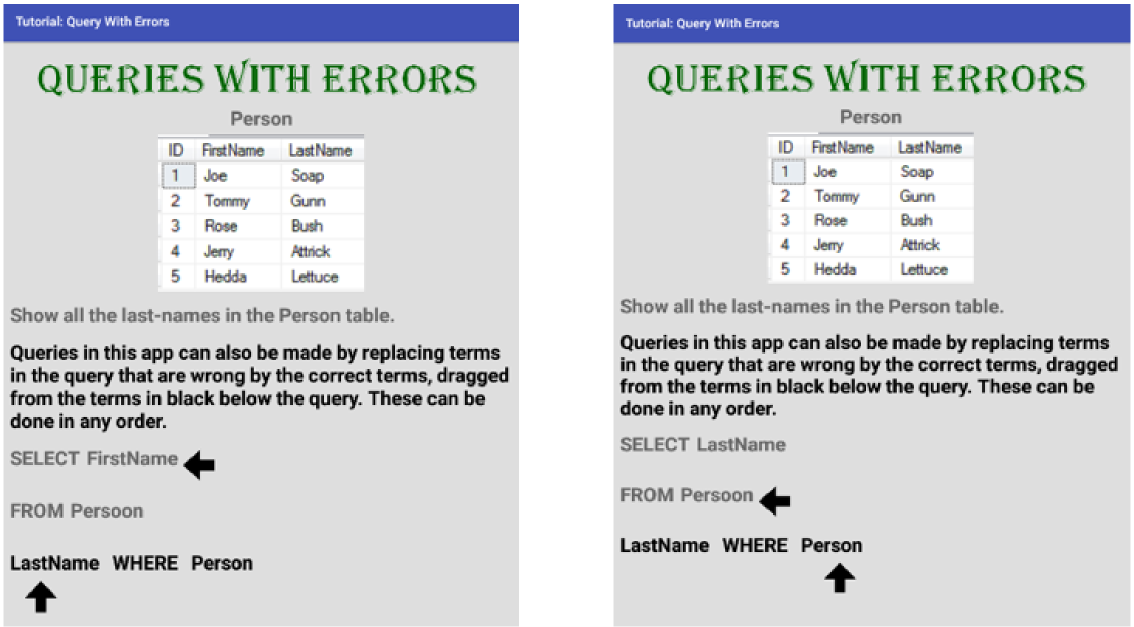
Figure 10: A second way of creating queries – click the terms in the correct sequence.

8.2 Queries With Errors

A third way of creating queries is where the query is shown with some mistakes in it. The player has to fix the query by dragging the correct terms from below into their proper places. In Figure 11 on the next page, the player is shown the query on the left, which has mistakes. These should be corrected, in any order. In the example shown, LastName is dragged to replace FirstName, creating the query shown on the right. Then Person is dragged to replace Persoon, making the query correct.

8.3 Term-Swapping Queries

A fourth way of creating queries is where the query is shown in the incorrect order. The player has to fix the query by swapping terms in the query with each other. In Figure 12 on the next page, the player is shown the query on the left. Terms can be re-arranged in any order. In the example shown, SELECT is swapped with LastName, creating the query on the right. Then LastName is swapped with FROM, making the query correct.



Tutorial: Query With Errors

QUERIES WITH ERRORS

Person

ID	FirstName	LastName
1	Joe	Soap
2	Tommy	Gunn
3	Rose	Bush
4	Jerry	Attrick
5	Hedda	Lettuce

Show all the last-names in the Person table.

Queries in this app can also be made by replacing terms in the query that are wrong by the correct terms, dragged from the terms in black below the query. These can be done in any order.

SELECT FirstName

FROM Persoon

LastName WHERE Person

Tutorial: Query With Errors

QUERIES WITH ERRORS

Person

ID	FirstName	LastName
1	Joe	Soap
2	Tommy	Gunn
3	Rose	Bush
4	Jerry	Attrick
5	Hedda	Lettuce

Show all the last-names in the Person table.

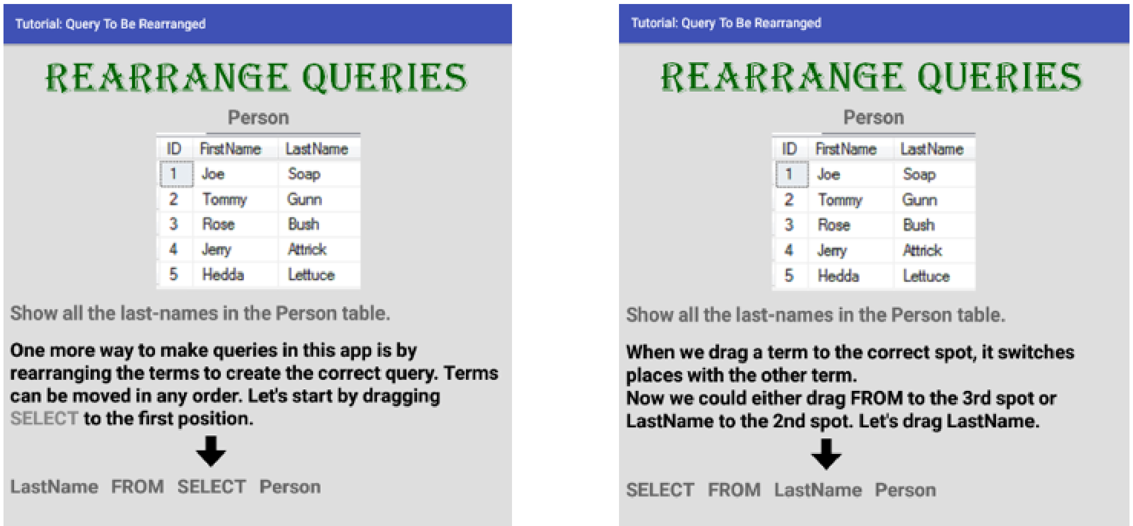
Queries in this app can also be made by replacing terms in the query that are wrong by the correct terms, dragged from the terms in black below the query. These can be done in any order.

SELECT LastName

FROM Persoon

LastName WHERE Person

Figure 11: A third way of creating queries – correct errors in the query.



Tutorial: Query To Be Rearranged

REARRANGE QUERIES

Person

ID	FirstName	LastName
1	Joe	Soap
2	Tommy	Gunn
3	Rose	Bush
4	Jerry	Attrick
5	Hedda	Lettuce

Show all the last-names in the Person table.

One more way to make queries in this app is by rearranging the terms to create the correct query. Terms can be moved in any order. Let's start by dragging SELECT **to the first position.**

LastName FROM SELECT Person

Tutorial: Query To Be Rearranged

REARRANGE QUERIES

Person

ID	FirstName	LastName
1	Joe	Soap
2	Tommy	Gunn
3	Rose	Bush
4	Jerry	Attrick
5	Hedda	Lettuce

Show all the last-names in the Person table.

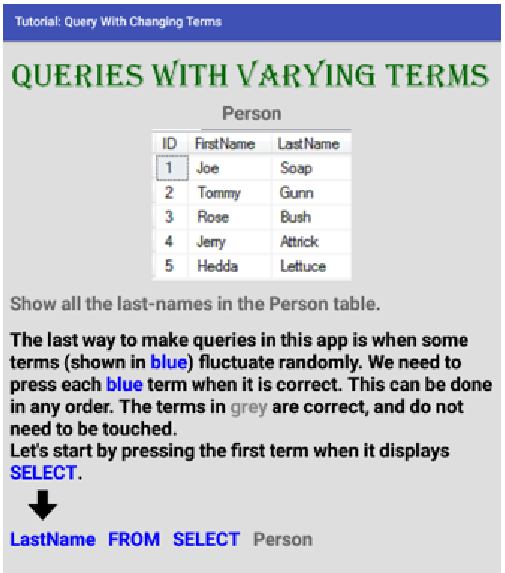
When we drag a term to the correct spot, it switches places with the other term.
Now we could either drag FROM to the 3rd spot or LastName to the 2nd spot. Let's drag LastName.

SELECT FROM LastName Person

Figure 12: A fourth way of creating queries – swapping terms in the query.

8.4 Randomly Fluctuating Queries

The final way of creating queries is where some of the terms in the query fluctuate randomly on the screen. The player has to click on each term when it is correct. The fluctuating terms are shown in blue. If the player clicks on a blue term when it displays the correct value for that position, that term changes color and stops fluctuating. The remaining terms continue to fluctuate until the player clicks on them when they are correct. Figure 13 below attempts to show this, although it is difficult to convey how this works through screen-shots, without seeing the moving terms on the screen.



Tutorial: Query With Changing Terms

QUERIES WITH VARYING TERMS

Person

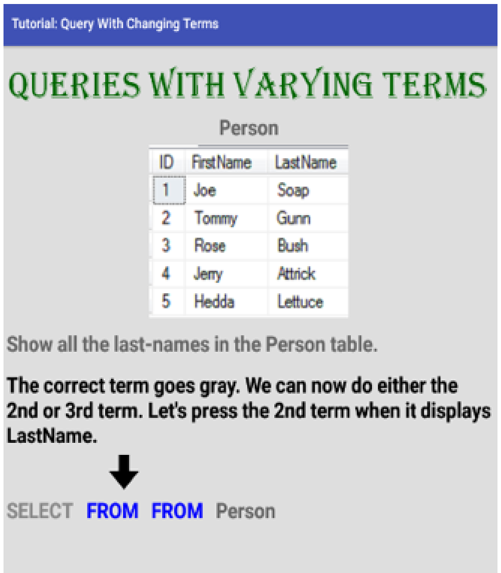
ID	FirstName	LastName
1	Joe	Soap
2	Tommy	Gunn
3	Rose	Bush
4	Jerry	Attrick
5	Hedda	Lettuce

Show all the last-names in the Person table.

The last way to make queries in this app is when some terms (shown in blue) fluctuate randomly. We need to press each blue term when it is correct. This can be done in any order. The terms in grey are correct, and do not need to be touched.
Let's start by pressing the first term when it displays SELECT.

🡇

LastName FROM SELECT Person



Tutorial: Query With Changing Terms

QUERIES WITH VARYING TERMS

Person

ID	FirstName	LastName
1	Joe	Soap
2	Tommy	Gunn
3	Rose	Bush
4	Jerry	Attrick
5	Hedda	Lettuce

Show all the last-names in the Person table.

The correct term goes gray. We can now do either the 2nd or 3rd term. Let's press the 2nd term when it displays LastName.

🡇

SELECT FROM FROM Person

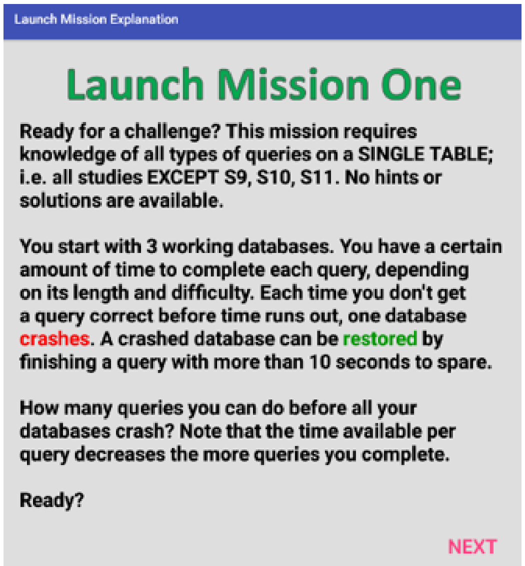
Figure 13: A fifth way of creating queries. The blue terms randomly fluctuate. The player must click on each term (in any order) when it is correct. This example shows the player clicking when the first term is SELECT in the left-screen, resulting in the right screen, where the other terms continue to fluctuate until the player clicks on them.

9 Missions

The final piece of the PlaySQL app are missions. Once a player has studied and taken the quizzes, they can embark on missions. There are currently three missions. Mission One covers all single-table queries (i.e. everything excluding joins), and contains 82 questions. Mission two covers all questions on joins (currently 17 questions), and Mission Three covers all topics (currently 99 questions).

In a mission, the player starts with 3 lives, represented by 3 databases, and encounters a randomly ordered sequence of questions. Each question has a certain amount of time in which it must be completed. This time depends on the difficulty of the query, and also decreases as the player progresses further in the mission. Thus the game becomes more challenging as the player's experience and knowledge increase. If the question is not solved before the time runs out, the player loses a life – a database crashes. A crashed database can be restored if the player solves a later question very quickly (currently it's set to restore a life if they solve a question with 10 seconds to spare). The game ends when the player runs out of lives or completes all the questions in the mission.

Figure 14 below shows the instructions for Mission One on the left. The instructions for Missions Two and Three are similar, except where Mission One has 3 lives, Mission Two provides only 2 lives (since it's shorter), while Mission Three gives 4 lives (since it's longer). The first, randomly selected, question of Mission One is shown on the right, with the timer decreasing. In Figure 15 on the left, time has run out before the player completes the question, so a database crashes – they lose a life. In Figure 15 on the right, they regain the life by completing the next question with more than 10 seconds to spare. And so the mission continues until they run out of lives or solve all the queries.



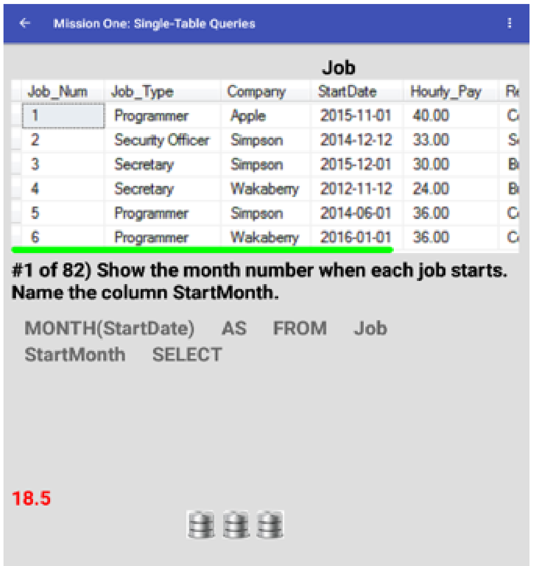
Launch Mission One

Ready for a challenge? This mission requires knowledge of all types of queries on a SINGLE TABLE; i.e. all studies EXCEPT S9, S10, S11. No hints or solutions are available.

You start with 3 working databases. You have a certain amount of time to complete each query, depending on its length and difficulty. Each time you don't get a query correct before time runs out, one database crashes. A crashed database can be restored by finishing a query with more than 10 seconds to spare.

How many queries you can do before all your databases crash? Note that the time available per query decreases the more queries you complete.

Ready?



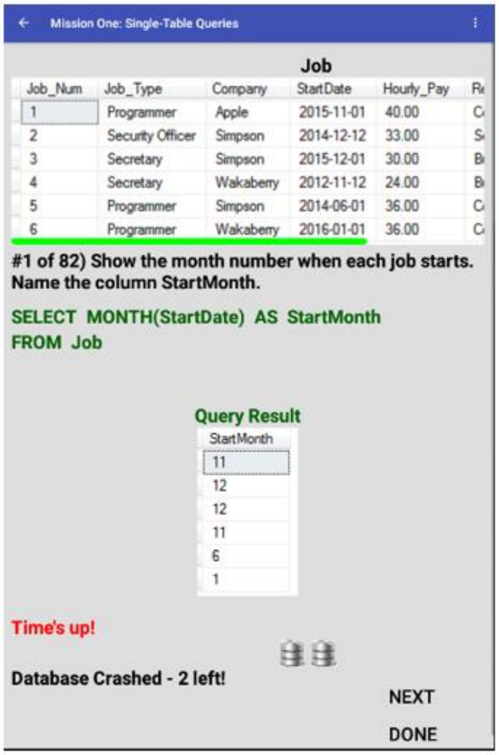
Job

Job_Num	Job_Type	Company	StartDate	Hourly_Pay	R
1	Programmer	Apple	2015-11-01	40.00	C
2	Security Officer	Simpson	2014-12-12	33.00	S
3	Secretary	Simpson	2015-12-01	30.00	B
4	Secretary	Wakaberry	2012-11-12	24.00	B
5	Programmer	Simpson	2014-06-01	36.00	C
6	Programmer	Wakaberry	2016-01-01	36.00	C

#1 of 82) Show the month number when each job starts. Name the column StartMonth.

```
MONTH(StartDate)   AS   FROM   Job
StartMonth   SELECT
```

Figure 14. Instructions for Mission One (left). First Question in Mission One (right). The timer decreases. If the player does not solve the query before time runs out, they lose a life (see Figure 15 below on the left).



Job

Job_Num	Job_Type	Company	StartDate	Hourly_Pay	R
1	Programmer	Apple	2015-11-01	40.00	C
2	Security Officer	Simpson	2014-12-12	33.00	S
3	Secretary	Simpson	2015-12-01	30.00	B
4	Secretary	Wakaberry	2012-11-12	24.00	B
5	Programmer	Simpson	2014-06-01	36.00	C
6	Programmer	Wakaberry	2016-01-01	36.00	C

#1 of 82) Show the month number when each job starts. Name the column StartMonth.

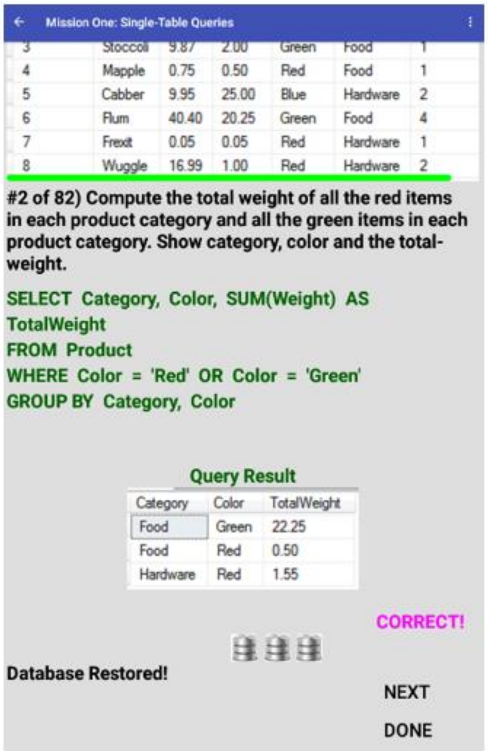
```
SELECT MONTH(StartDate) AS StartMonth
FROM Job
```

Query Result

StartMonth
11
12
12
11
6
1

Time's up!

Database Crashed - 2 left!



3	Stoccoli	9.87	2.00	Green	Food	1
4	Mapple	0.75	0.50	Red	Food	1
5	Cabber	9.95	25.00	Blue	Hardware	2
6	Flum	40.40	20.25	Green	Food	4
7	Frexit	0.05	0.05	Red	Hardware	1
8	Wuggle	16.99	1.00	Red	Hardware	2

#2 of 82) Compute the total weight of all the red items in each product category and all the green items in each product category. Show category, color and the total-weight.

```
SELECT Category, Color, SUM(Weight) AS
TotalWeight
FROM Product
WHERE Color = 'Red' OR Color = 'Green'
GROUP BY Category, Color
```

Query Result

Category	Color	TotalWeight
Food	Green	22.25
Food	Red	0.50
Hardware	Red	1.55

CORRECT!

Database Restored!

Figure 15. The player loses a life by running out of time on Question 1 (left). They regain the life by solving Question 2 with more than 10 seconds to spare (right).

10 Testing

During the development of the app I worked with a variety of undergraduate students, some who had taken a database class and some who had not. The students provided valuable feedback and many of their ideas have been incorporated into the app. I have not used the app in the classroom yet, though I hope to do so soon.

I believe the app could be effectively used by someone with no database experience to learn the basics of SQL programming, but this has yet to be demonstrated. The app still needs additional work before being ready to be made available on the PlayStore. For example, the current version doesn't run effectively on tablets or phones with smaller screen sizes.

11 Conclusion

PlaySQL is an Android app that uses a game to teach the basic of SQL querying. The app allows a user with no previous programming experience to learn database programming in a self-guided style. The material is carefully structured to proceed from simple to more complicated queries. The player receives feedback that helps to improve their querying ability.

Developing the app was very enjoyable and instructive. I learned a great deal in a number of areas, which will help improve my pedagogy. These areas include Android app development, databases, and software development.

I feel that for some topics, games could be effectively used to teach college-level material in a way that could supplement traditional teaching methods. On the other hand, games are not appropriate for all topics. For example, normalization is a major part of our introductory database class. It is not clear to me how, or even if, normalization could be effectively conveyed using a game or an app.

Acknowledgments

This work was made possible by the generous support of Simpson College during a sabbatical.

Many Simpson students helped provide feedback as the app was developed. These include Elise Tauer, George Saucedo, Karson Richardson, Ryan Policheri, Kaleb Klepel, Taylor Gehrls, Josh Dietrich, Geoff Converse, Mark Becker, Sam Adeniyi.

The author also thanks the creators of Android's FlexBoxLayout [14], available on GitHub, which proved invaluable in allowing for flexible layout of the SQL queries in the app.

References

[1] H. G. &. K. Davis, The App Generation: How Today’s Youth Navigate Identity, Intimacy, and Imagination in a Digital World, Yale University Press, 2013.
[2] J. M. Twenge, iGen: Why Today’s Super-Connected Kids Are Growing Up Less Rebellious, More Tolerant, Less Happy--and Completely Unprepared for Adulthood--and What That Means for the Rest of Us, Atria Books, 2017.
[3] "Shakespeare app on iTunes," PlayShakespeare.com, 03 August 2014. [Online]. Available: https://itunes.apple.com/us/app/shakespeare/id285035416?mt=8. [Accessed 09 March 2018].
[4] "Shakespeare Complete Works on Google Play," Hongshee.com, 16 Feb 2014. [Online]. Available: https://play.google.com/store/apps/details?id=com.hongshee.mobile.anbook.shakespeare&hl=en. [Accessed 09 March 2018].
[5] P. Hudson, "Ancient Greek on on iTunes," 01 Nov 2013. [Online]. Available: https://itunes.apple.com/us/app/ancient-greek/id463233342?mt=8. [Accessed 09 March 2018].
[6] D. Arthur, "LP Ancient Greek on Google Play," 10 May 2014. [Online]. Available: https://play.google.com/store/apps/details?id=ca.libphil.lang.a.ancientgreek. [Accessed 09 March 2018].
[7] "Learn SQL", Android App by SoloLearn, [Online]. https://play.google.com/store/apps/details?id=com.sololearn.sql. [Accessed 09 March 2018].
[8] "Learn SQL Query", Android app by SS Media Labs, [Online]. https://play.google.com/store/apps/details?id=com.samaysoftware.sql. [Accessed 09 March 2018].
[9] "SQL Query Executor" by iKosmikSoftware, [Online]. https://play.google.com/store/apps/details?id=com.ikosmik.app.adhoc.android.free. [Accessed 09 March 2018].
[10] "SQL Practice Pro" by RandomAppsInc, [Online]. https://play.google.com/store/apps/details?id=randomappsinc.com.sqlpracticeplus. [Accessed 09 March 2018].
[11] "SQL Tutorial", by Ajay Katri, [Online]. https://play.google.com/store/apps/details?id=in.ajaykhatri.sqltutorial. [Accessed 09 March 2018].
[12] "SQL Tutorial – Kiwi Learn" by kiwilabs, [Online]. https://play.google.com/store/apps/details?id=com.kiwilearn.sqltutorial. [Accessed 09 March 2018].
[13] "Tutorial for SQL" by Orange Technologies, [Online]. https://play.google.com/store/apps/details?id=com.gd.tutorialforsql. [Accessed 09 March 2018].
[14] "Flexbox for Android" availabe on Github, [Online] https://github.com/google/flexbox-layout. [Accessed 14 March 2018].