Table of Contents

Login

Abstract Code

- User enters *username* and *password* into input fields
- User select *user type* in the radio buttons.
- When ***Sign in*** button is clicked:
    - Query User table:
      SELECT userID, username, password FROM User WHERE username='$username';
    - If *user type* is Customer, then:
        - If user record is not found:
            - Go to the **Registration** form.
        - Else if user record is found but $*password* does not match User.password:
            - Go back to **Login** form, with error message.
        - Else:
            - Go to **Customer Main Menu** form.
    - Else if *user type* is Clerk, then:
        - If user record is not found:
            - Go back to **Login** form, with error message.
        - Else if user record is found but $*password* does not match User.password:
            - Go back to **Login** form, with error message.
        - Else if user record is found and $password matches User.password:
            - SELECT has_logged_in_before FROM Clerk WHERE userID = '$userID';
            - If has_logged_in_before is false
                - Change password by entering new password twice
                - If both passwords are the same, then:
                    - Write new password to User.*password*
                        - UPDATE User SET `password` = '$password' WHERE userID = '$userID';
                    - Update has_logged_in_before on Clerk
                        - UPDATE Clerk SET has_logged_in_before = True WHERE userID = '$userID';
                    - Go to **Clerk Main Menu** form.
                - Else go back to Enter new password twice again
            - Else:
                - Go to **Clerk Main Menu** form.

Table of Contents

Customer Main Menu / Navigation

Abstract Code

- Show View Profile, Check Tool Availability, Make Reservation, Purchase Tool, and Exit (or Logout) links
- Get the user *firstname* and display “Welcome *firstname*!” in the navigation bar.
    - SELECT first_name, userID FROM User
      WHERE userID = '$userID';
- Upon:
    - Click **View Profile** link - Jump to the **View Profile** task.
    - Click **Check Tool Availability** link - Jump to the **Check Availability** task.
    - Click **Make Reservation** link - Jump to the **Make Reservation** task.
    - Click **Purchase Tool** link - Jump to the **Purchase Tool** task.
    - Click **Exit (or Logout)** link - Logout from the system.

Table of Contents

Register

Abstract Code

- User enters *first name*, *middle name*, *last name*, *home phone*, *work phone*, *cell phone*, *username, email, password, street address, city, state,* and *zip code* into input fields plus select *primary phone number.* In addition, the user enters the credit card information, which includes: *name on credit card*, *credit card number*, *expiration month*, *expiration year* and *CVC.*
    - The primary phone number selected cannot be null.
- If data validation is successful for all input fields, then:
- When ***Register*** button is clicked:
    - SELECT username FROM User WHERE username='$username';
    - If *username* already exists, then:
        - Go back to **Registration** form, already populated but with error message on *username*
    - Else:
        - START TRANSACTION;
        - INSERT INTO User (username, password, email, first_name, middle_name, last_name)
          VALUES ('$username', '$password', '$email', '$first_name', '$middle_name', '$last_name');
        - INSERT INTO Customer (userID, zip_code, street, city, state)
          VALUES (LAST_INSERT_ID(), '$zip_code', 'street','city','state');
            - --If HomePhone entered then:
                - INSERT INTO HomePhone (userID, area_code, phone_number, extension) VALUES (LAST_INSERT_ID(), '$area_code', '$phone_number', '$extension');
            - --If WorkPhone entered then:
                - INSERT INTO WorkPhone (userID, area_code, phone_number, extension) VALUES (LAST_INSERT_ID(), '$area_code', '$phone_number', '$extension');
            - --If CellPhone entered then:
                - INSERT INTO CellPhone (userID, area_code, phone_number, extension) VALUES (LAST_INSERT_ID(), '$area_code', '$phone_number', '$extension');

Table of Contents

- INSERT INTO PrimaryPhone (userID, area_code, phone_number, extension) VALUES (LAST_INSERT_ID(), '$area_code', '$phone_number', '$extension');
- INSERT INTO CreditCard (userID, name, cred_number, exp_month, exp_year, cvc) VALUES (LAST_INSERT_ID(), '$name', '$cred_num', '$exp_month', '$exp_year', '$cvc');
- COMMIT;

Table of Contents

View Profile

Abstract Code

- User clicked on ***View Profile*** link from **Main Menu / Nav Bar** form.
- Run the **View Profile** task: query the information about the customer and their reservations where $username is the username of the current user using the system from the HTTP session/cookie.
    - Find the current Customer information: *email*, *full name*, cell *phone*, work *phone*, home *phone*, primary *phone,* and *address*
      SELECT email, first_name, middle_name, last_name, zip_code, street, city, state
      FROM User INNER JOIN Customer on User.userID= Customer.userID
      INNER JOIN PrimaryPhone on User.userID=PrimaryPhone.userID
      INNER JOIN HomePhone on User.userID=HomePhone.userID
      INNER JOIN CellPhone on User.userID=CellPhone.userID
      INNER JOIN WorkPhone on User.userID=WorkPhone.userID
      WHERE User.userID = '$userID';
    - Find all Reservation for the current Customer ordered from most recent to oldest.
      SELECT reservationID FROM Reservation WHERE customerUserID = $userID
      ORDER BY start_date DESC;
    - For each Reservation for the Customer:
        - Find the: *reservation ID, tools, start date, end date, pick-up clerk, drop-off clerk, number of days, total deposit price*, and *total rental price*
          SELECT Reservation.reservationID, Tool.toolNumber, start_date,
          end_date, pickUpUserID, dropOffUserID,
          ROUND((purchase_price * 0.15), 2) AS rental_price,
          ROUND((purchase_price * 0.40), 2) AS deposit_price
          FROM Reservation
          INNER JOIN IsOf ON Reservation.reservationID=IsOf.reservationID
          INNER JOIN Tool ON IsOf.toolNumber=Tool.toolNumber
          WHERE customerUserID = '$userID';
        - Total rental price is a summation of 15% of purchase price of each Tool in the reservation.
        - Total deposit price is a summation of 40% of purchase price of each Tool in the reservation
        - Days rented is a subtraction of end_date from start_date of Reservation
- When ready, customer selects next action from **Customer Main Menu / Nav Bar**.

Table of Contents

Check Tool Availability

Abstract Code

- User clicked on ***Check Tool Availability*** link from **Customer Main Menu / Nav Bar** form.
- Run the **Check Tool Availability** task: query the Tool, Type, Sub-Type, Power Source, Sale Order, Reservation, and Service Order to find the available tools based on a search criteria from the customer.
    - Type is populated
        - SELECT DISTINCT tt_name
          FROM ToolTypeOption;
    - Power Source is populated
        - SELECT DISTINCT ps_name
          FROM ToolTypeOption
          WHERE tt_name = '$tt_name';
    - Sub Type is populated:
        - SELECT DISTINCT tst_name
          FROM ToolTypeOption
          WHERE tt_name = '$tt_name'
          AND ps_name = '$ps_name';
    - Customer fill in *start date, end date, custom search, type, sub-type* and *power source* input fields for the search criteria
    - Click the ***Search*** button.
    - Find all Tool that are available
        - Get all tools based on the search criteria and not “for-sale” and not “reserved” and not “in-repair” and not rented 50 times.
            - SELECT DISTINCT Tool.toolNumber, type, sub_type, sub_option, power_source,
              ROUND((purchase_price * 0.15), 2) AS rental_price,
              ROUND((purchase_price * 0.40), 2) AS deposit_price
              FROM Tool
              LEFT JOIN SaleOrder
              ON SaleOrder.toolNumber = Tool.toolNumber
              LEFT JOIN ServiceOrderRequest
              ON ServiceOrderRequest.toolNumber = Tool.toolNumber
              LEFT JOIN IsOf
              ON IsOf.toolNumber = Tool.toolNumber
              LEFT JOIN Reservation
              ON Reservation.reservationID = IsOf.reservationID
              WHERE SaleOrder.saleOrderID IS NULL
              AND Tool.toolNumber NOT IN

Table of Contents

```
(
        SELECT Tool.toolNumber
        FROM Tool
        LEFT JOIN ServiceOrderRequest
        ON ServiceOrderRequest.toolNumber = Tool.toolNumber
        WHERE
        (
                (STR_TO_DATE('$searchStartDate', '%Y-%m-%d
%H:%i:%s') < ServiceOrderRequest.end_date)
                AND
                (STR_TO_DATE('$searchEndDate', '%Y-%m-%d
%H:%i:%s') > ServiceOrderRequest.start_date)
        )
)
AND Tool.toolNumber NOT IN
(
        SELECT Tool.toolNumber
        FROM Tool
        LEFT JOIN IsOf
        ON IsOf.toolNumber = Tool.toolNumber
        LEFT JOIN Reservation
        ON Reservation.reservationID = IsOf.reservationID
        WHERE
        (
                (STR_TO_DATE('$searchStartDate', '%Y-%m-%d
%H:%i:%s') < Reservation.end_date)
                AND
                (STR_TO_DATE('$searchEndDate', '%Y-%m-%d
%H:%i:%s') > Reservation.start_date)
        )
)
AND Tool.toolNumber NOT IN (
        SELECT Tool.toolNumber
        FROM Tool
        LEFT JOIN IsOf
        ON IsOf.toolNumber = Tool.toolNumber
        LEFT JOIN Reservation
        ON Reservation.reservationID = IsOf.reservationID
        WHERE Reservation.pickUpUserID IS NOT NULL
        GROUP BY Tool.toolNumber
        HAVING COUNT(IsOf.reservationID) >= 50
)
AND Tool.type = '$typeName'
```

Table of Contents

```
AND Tool.sub_type = '$subTypeName'
AND Tool.power_source = '$powerSourceName'
AND Tool.sub_option LIKE '$customSearch%';
```

- If more than 10 tools retrieved on the query, then:
  - Prompt user to specify a more unique search by entering more criteria
- Else:
  - Create short-description:
    - If power_source == Manual → Description is a string concatenation of [sub_option] + [sub_type]
    - Else → Description is a string concatenation of [power_source] + [sub_option] + [and sub_type]
  - Display the *toolNumber*, *short-description*, *rental price*, *deposit price*
  - The *short-description* is a link for the customer to pop-up the **Tool Details** form.

- When ready, customer selects next action from **Customer Main Menu / Nav Bar**.

Table of Contents

Make Reservation

Abstract Code

- User clicked on ***Make Reservation*** link from **Customer Main Menu / Nav Bar** form.
- Run the **Check Tool Availability** task to load the tools
    - To each tool loaded, add a checkbox (unchecked by default), which is going to be used to add a tool to the reservation.
- User add tools to the reservation by checking the box on the tool
    - When the tool is checked,
    - If tool is due to return within the next 24 hours
        - Display notification with date and time when that tool is expected to become available.

```
SELECT Tool.toolNumber, Reservation.end_date
FROM Tool
INNER JOIN IsOf
ON IsOf.toolNumber = Tool.toolNumber
INNER JOIN Reservation
ON Reservation.reservationID = IsOf.reservationID
WHERE DATEDIFF(Reservation.end_date, curdate()) =
'$toolNumber'
```

            - "You have the option to rent the tool later or make an alternate selection."
    - Else If tools added to reservation is greater than 10, then:
        - Display error message prompting the user to reduce the number of tools in the current reservation to 10.
    - Else
        - The tool is moved to the "Tools Added to Reservation" area.
        - Display the *tool ID*, *description*, *rental price*, *deposit price* and a check box.
            - The checkbox (checked by default) is to remove the tool from the reservation.
- User can repeat the previous steps to search for other tools and keep modifying the reservation.
- When the ***Calculate*** button is pressed, the **Summarize Reservation** task is executed.
- The **Reservation Summary** form is loaded with the following 2 section details, which are filled out by executing the **Summarize Reservation** task (handle with concurrency in mind):
    - The **View Price/Date Summary** task will be executed in the application, no need to query database, to fill out the top section of the Reservation Summary, which includes Reservation Dates, Number of days rented, Total Deposit Price, and Total Rental Price

Table of Contents

        - The Number of days rented is calculated based on the Reservation start and end dates.
        - The Total Deposit is calculated as the sum of the deposit price for each individual tool.
        - The Total Rental Price is calculated as the sum of the rental price for each individual tool, multiplied by the Number of days over which they are rented.
    - The **View Tool Summary** task will be executed to fill out the bottom section of the Tools details for the reservation which includes the *ToolID*, *description*, *deposit price* and *rental price*
        - The rental price is calculated as the sum of the rental price multiplied by the number of days.
    - When the **Submit** button is pressed, then insert the reservation information to the database, which includes a verification of all the tools to make sure they are still available:
        - Insert the reservation information to the database, which includes: List of *ToolIDs*, *confirmationNum*, *reservationID*, *number of days rent*, *total deposit price*, *total rental price*, *reservationDate* (start date and end date).
            - START TRANSACTION;
            - INSERT INTO Reservation (customerUserID, start_date, end_date, pickUpUserID, dropOffUserID)
              VALUES ('$userID', STR_TO_DATE('$startDate', '%Y-%m-%d %H:%i:%s'), STR_TO_DATE('$endDate', '%Y-%m-%d %H:%i:%s'), NULL, NULL);
            - For each tool, check if the tool is still available and insert. If the tool is not available anymore, an SQL Exception will be thrown by the Database
                - SET @tn = NULL;
                - SELECT @tn := Tool.toolNumber
                  FROM Tool
                  LEFT JOIN SaleOrder
                  ON SaleOrder.toolNumber = Tool.toolNumber
                  LEFT JOIN ServiceOrderRequest
                  ON ServiceOrderRequest.toolNumber = Tool.toolNumber
                  LEFT JOIN IsOf
                  ON IsOf.toolNumber = Tool.toolNumber
                  LEFT JOIN Reservation
                  ON Reservation.reservationID = IsOf.reservationID
                  WHERE SaleOrder.saleOrderID IS NULL
                  AND Tool.toolNumber NOT IN
                  (
                  SELECT Tool.toolNumber
                  FROM Tool

Table of Contents

```
                    LEFT JOIN ServiceOrderRequest
                    ON ServiceOrderRequest.toolNumber =
                        Tool.toolNumber
                    WHERE
                    (
                    (STR_TO_DATE('$startDate', '%Y-%m-%d
            %H:%i:%s') < ServiceOrderRequest.end_date)
                    AND
                    (STR_TO_DATE('$endDate', '%Y-%m-%d
        %H:%i:%s') > ServiceOrderRequest.start_date)
                    )
            )
            AND Tool.toolNumber NOT IN
            (
                    SELECT Tool.toolNumber
                    FROM Tool
                    LEFT JOIN IsOf
                    ON IsOf.toolNumber = Tool.toolNumber
                    LEFT JOIN Reservation
                    ON Reservation.reservationID =
                        IsOf.reservationID
                    WHERE
                    (
                    (STR_TO_DATE('$startDate', '%Y-%m-%d
        %H:%i:%s') < Reservation.end_date)
                    AND
                    (STR_TO_DATE('$endDate', '%Y-%m-%d
        %H:%i:%s') > Reservation.start_date)
                    )
            )
            AND Tool.toolNumber = $toolNumber;

            INSERT INTO IsOf (reservationID, toolNumber)
                    VALUES (LAST_INSERT_ID(), @tn);
```

- If everything was Ok, then Commit transaction.
    - COMMIT;
    - The **Reservation Confirmation** form is displayed.
- Else if an SQL exception is thrown
    - Display error message specifying the reservation was unsuccessful because a tool is not available.
        - Include tool ID and description on error message
        - Go back to **Make Reservation** form.

Table of Contents

        - When the **Reset** button is pressed the reservation is deleted from the local cache.
            - The **Main Menu / Nav Bar** is displayed.
            - When ready, customer selects next action from **Customer Main Menu / Nav Bar**.
- When the **Reservation Confirmation** form is loaded, the **Confirm Reservation** task is executed.
    - The **Reservation Confirmation** form is the same as the **Reservation Summary** but the title is “Reservation Confirmation” plus it includes the *Reservation ID* underneath the title.
- When ready, customer selects next action from **Customer Main Menu / Nav Bar**.

Table of Contents

Purchase Tool

Abstract Code

- User clicked on ***Purchase Tool*** link from **Customer Main Menu / Nav Bar** form.
- Run the **Purchase Too**l task: query the Tool, Type, Sub-Type, Power Source, and Sale Order to find the available tools based on a search criteria from the customer.
    - Clerk picks the power source (ps_name):
        - SELECT DISTINCT * FROM ToolTypeOption where ps_name = '$ps_name';
    - Clerk picks the power source (ps_name) and the tool-type (tt_name):
        - SELECT DISTINCT * FROM ToolTypeOption where ps_name = '$ps_name' AND tt_name= '$tt_name'
    - Clerk picks the power source (ps_name) AND the tool-type (tt_name) AND the tool sub-type (tst_name):
        - SELECT DISTINCT * FROM ToolTypeOption where ps_name = '$ps_name' AND tt_name='$tt_name' AND tst_name = '$tst_name';
    - Customer fill in *custom search, type, sub-type* and *power source* input fields for the search criteria
    - Click the ***Search*** button.
    - Find all Tool that are available for-sale by executing the **Search For-Sale Inventory** task:
        - Get all tools marked for-sale on the Sale Order table based on the search criteria
          SELECT saleOrderID FROM SaleOrder WHERE sold_date = NULL;
        - If more than 10 tools retrieved on the query, then:
            - Prompt user to specify a more unique search by entering more criteria
        - Else:
            - Display the *tool ID*, *current status*, *short-description*, *purchase* price, and a purchase tool button.
              SELECT Tool.toolNumber, sub_type, sub_option, power_source,
              ROUND((purchase_price * 0.50), 2) AS sale_price
              FROM SaleOrder
              INNER JOIN Tool
              ON Tool.toolNumber = SaleOrder.toolNumber
              WHERE sold_date IS NULL;
                - If power_source == Manual → Description is a string concatenation of [sub_option] + [sub_type]
                - Else → Description is a string concatenation of [power_source] + [sub_option] + [and sub_type]

Table of Contents

- The *description* is a link for the customer to pop-up the **Tool Details** form.
- The purchase tool button is to add a tool to the purchase order.

- User add tools to the purchase order by pressing the button on the tool
  - When the **Purchase Tool** button is pressed,
    - The tool is moved to the “Tools Added to Purchase” area.
    - Display the tool ID, description, purchase price, and a remove tool button.
      SELECT toolNumber, sub_option, sub_type, power_source, purchase_price
      FROM Tool
      WHERE toolNumber = '$toolID';
      - If power_source == Manual → Description is a string concatenation of [sub_option] + [sub_type]
      - Else → Description is a string concatenation of [power_source] + [sub_option] + [and sub_type]
      - The **Remove Tool** button is to remove the tool from the reservation.
- User can repeat the previous steps to search for other tools and keep modifying the purchase order.
- When the ***Calculate*** button is pressed, the **Summarize Purchase** task is executed.

- The **Purchase Summary** form is loaded with the following, which are filled out by executing the **Summarize Purchase** task:
  - The **View Tool Summary** task will be executed to fill out the bottom section of the Tools details for the reservation which includes the *ToolID*, *description*, and *sale price,* as well as the total sale price.
    SELECT Tool.toolNumber, sub_option, sub_type, power_source,
    ROUND((purchase_price * 0.50), 2) AS sale_price
    FROM SaleOrder
    INNER JOIN Tool
    ON Tool.toolNumber = SaleOrder.toolNumber
    WHERE Tool.toolNumber = '$toolID';
    - If power_source == Manual → Description is a string concatenation of [sub_option] + [sub_type]
    - Else → Description is a string concatenation of [power_source] + [sub_option] + [and sub_type]
    - Total Sale Price is calculated by the sum of all the tools sale price.
  - When the **Submit** button is pressed,
    - Execute query to verify the tools has not been purchased by another customer.

Table of Contents

- Query the database to verify tools are available for purchase
  SELECT Tool.toolNumber, type, sub_type, sub_option, power_source, purchase_price
  FROM Tool
  JOIN SaleOrder
  ON SaleOrder.toolNumber != Tool.toolNumber
  JOIN ServiceOrderRequest
  ON ServiceOrderRequest.toolNumber != Tool.toolNumber
  LEFT OUTER JOIN IsOf
  ON IsOf.toolNumber = Tool.toolNumber
  LEFT OUTER JOIN Reservation
  ON Reservation.reservationID = IsOf.reservationID
  WHERE
  Reservation.end_date < STR_TO_DATE('$searchStartDate', '%Y-%m-%d %H:%i:%s')
  OR Reservation.start_date > STR_TO_DATE('$searchEndDate', '%Y-%m-%d %H:%i:%s')
  OR Reservation.start_date IS NULL
  AND Tool.toolNumber = '$toolID';
- If all tools available, then for each tool:
  - then insert the ***customer ID*** and ***soldDate*** to the Sale Order instance in the database.
    INSERT INTO SaleOrder (saleOrderID, for_sale_date)
    VALUES ('$saleOrderID', '$for_sale_date')
  - The **Purchase Confirmation** form is displayed.
- Else
  - Display error message specifying the reservation was unsuccessful because a tool is not available.
    - Include tool ID and description on error message
      SELECT toolNumber, sub_option, sub_type, power_source
      FROM Tool
      WHERE toolNumber = '$toolID';
      - If power_source == Manual → Description is a string concatenation of [sub_option] + [sub_type]
      - Else → Description is a string concatenation of [power_source] + [sub_option] + [and sub_type]
    - Go back to **Purchase Tool** form.
- When the **Reset** button is pressed the purchase is deleted from the local cache.
  - The **Customer Main Menu / Nav Bar** is displayed.

Table of Contents

        - When ready, customer selects next action from **Customer Main Menu / Nav Bar**.
- When the **Purchase Confirmation** form is loaded, the **Confirm Purchase** task is executed.
    - The **Purchase Confirmation** task have 2 sub-tasks that can run in parallel:
        - **View Purchase Information** task is executed to display the top section of the Purchase Confirmation form, which includes:
            - Date, Total Sale Price, and Credit Card Number

            ```
            SELECT Tool.toolNumber, sub_option, sub_type, power_source,
            ROUND((purchase_price * 0.50), 2) AS sale_price,
            FROM SaleOrder
            INNER Join Tool
            WHERE Tool.toolNumber = '$toolID';
            ```

                - Total Sale Price is calculated by the sum of all the tools sale price.
        - The bottom section is loaded by the **View Tool Summary** task, which is described under the **Summarize Purchase** task.
    - When the **Print Receipt** button is pressed, the host operating system's print dialog box will open.
- When ready, customer selects next action from **Customer Main Menu / Nav Bar**.

Table of Contents

Clerk Main Menu / Navigation

Abstract Code

- Show Pick-Up Reservation, Drop-Off Reservation, Add New Tool, Service Order, Service Status, Sell Tool, Sale Status, Generate Reports and Logout links.
- Get the user firstname and display “Welcome firstname!” in the navigation bar.
    - SELECT first_name, userID FROM User WHERE userID = '$userID';
- Upon:
    - Click ***Pick-Up Reservation*** link - Jump to the **Pick-Up Reservation** task.
    - Click ***Drop-Off Reservation*** link - Jump to the **Drop-Off Reservation** task.
    - Click ***Add New Tool*** link - Jump to the **Add/View Tool Details** task.
    - Click ***Service Order*** link - Jump to the **Request Service Order** task.
    - Click ***Service Status*** link - Jump to the **View Service Status** task.
    - Click ***Sell Tool*** link - Jump to the **Sell Tool** task
    - Click ***Sale Status*** link - Jump to the **View Sell Status** task
    - Click ***Generate Reports*** link - Jump to the **Generate Reports** task.
    - Click ***Logout*** - Logout from the system.

Table of Contents

Pick Up Reservation

Abstract Code

- Clerk clicked ***Pick Up Reservation*** from the **Clerk Main Menu/Navigation**.
    - Run **Search Outstanding Reservations** to query all outstanding reservations.
        - For each reservation in the database, query and display the *Customer, C*ustomerID, Reservation Start Date and Reservation End Date.
          SELECT User.username, customerUserID, end_date, start_date, reservationID
          FROM Reservation INNER JOIN User
          ON Reservation.customerUserID = User.userId
          WHERE pickUpUserID IS NULL AND dropOffUserID IS NULL;
    - If a specific Reservation is clicked
        - Display the **Reservation Details**
        - For the reservation, query and display *Customer Name, Total Deposit and* Total Rental Price.
          SELECT Reservation.reservationID, User.first_name, User.last_name,
          ROUND((Tool.purchase_price * 0.15), 2) AS rental_price,
          ROUND((Tool.purchase_price * 0.40), 2) AS deposit_price
          FROM Reservation INNER JOIN User
          ON Reservation.customerUserID = User.userId
          INNER JOIN IsOf ON IsOf.reservationID = Reservation.reservationID
          INNER JOIN Tool ON IsOf.toolNumber = Tool.toolNumber
          WHERE Reservation.reservationID = '$reservationID';
        - Total rental price is a summation of 15% of purchase price of each Tool in the reservation.
        - Total deposit price is a summation of 40% of purchase price of each Tool in the reservation
    - Specific reservation is entered and the button ***Pick Up*** is clicked
        - The **Rental Pick-Up Summary** is displayed
- Clerk entered a specific reservation number and clicked the button ***Pick Up*** on **Pick Up Reservations** page
    - Load **Rental Pick-Up Summary** page to query the selected reservation
        - For the specific reservation, populate text fields with *Reservation ID, Customer Name, Total Deposit, Total Rent Price and the Customer Credit Card Information.*
          SELECT Reservation.reservationID, User.first_name, User.last_name, cred_number, name, cvc, exp_month, exp_year,
          ROUND((Tool.purchase_price * 0.15), 2) AS rental_price,
          ROUND((Tool.purchase_price * 0.40), 2) AS deposit_price
          FROM Reservation

Table of Contents

INNER JOIN User ON Reservation.customerUserID = User.userID
INNER JOIN IsOf ON IsOf.reservationID = Reservation.reservationID
INNER JOIN Tool ON IsOf.toolNumber = Tool.toolNumber
INNER JOIN CreditCard ON CreditCard.userID = User.userID
WHERE Reservation.reservationID = '$reservationID';

        - Total rental price is a summation of 15% of purchase price of each Tool in the reservation.
        - Total deposit price is a summation of 40% of purchase price of each Tool in the reservation
    - If the Customer wants to use the Existing Credit Card
        - Clerk selected the Existing radio button.
        - Query and display the Credit Card fields *Name on Credit Card, Credit* Card #, CVC, Expiration Month and Expiration Year.
            - SELECT cred_number, name, cvc, exp_month, exp_year FROM CreditCard WHERE userID = '$userID';

    - If the Customer wants to use a New Credit Card
        - Clerk selected the “New” radio button.
        - Expiration Month and Expiration Year are auto-populated with available DateTime values.
        - Clerk inputs the *Name on Credit Card, Credit Card #, CVC, Expiration Month and Expiration Year* of the Customer Credit Card
          UPDATE CreditCard SET name='$name', cred_number = '$cred_number', cvc = '$cvc', exp_month = '$exp_month', exp_year = '$exp_year' WHERE userID = '$userID';
    - ***Confirm Pick Up*** is clicked when all information is entered
        - If New Credit Card information was entered, it is stored to the database.
- Clerk clicked the button ***Confirm Pick Up*** on the **Rental Pick-Up Summary** page
    - Display **Rental Contract Summary** page
        - Query and populate *Pick-Up Clerk, Customer Name, Credit Card #, Start Date, End Date, Tool IDs* being reserved, *Tool Names* being reserved, *Deposit Price* for each tool and *Rental Price* for each tool.
          SELECT Reservation.reservationID, User.first_name, User.last_name, cred_number, name, cvc, exp_month, exp_year, pickUpUserID, dropOffUserID, start_date, end_date,
          ROUND((Tool.purchase_price * 0.15), 2) AS rental_price,
          ROUND((Tool.purchase_price * 0.40), 2) AS deposit_price
          FROM Reservation
          INNER JOIN User ON Reservation.customerUserID = User.userID
          INNER JOIN IsOf ON IsOf.reservationID = Reservation.reservationID
          INNER JOIN Tool ON IsOf.toolNumber = Tool.toolNumber
          INNER JOIN CreditCard ON CreditCard.userID = User.userID
          WHERE Reservation.reservationID = '$reservationID';

Table of Contents

        - Total rental price is a summation of 15% of purchase price of each Tool in the reservation.
        - Total deposit price is a summation of 40% of purchase price of each Tool in the reservation
    - If a Tool Name is clicked, the **Tool Details** are displayed for the tool
        - Call the Helper Tool Details Helper Task
    - Upon clicking ***Print Contract***
        - OS Print dialog is opened.
- Upon printing summary, return to **Clerk Main Menu/Navigation**

Table of Contents

Drop Off Reservation

Abstract Code

- Clerk clicked ***Drop Off Reservation*** from the **Clerk Main Menu/Navigation**.
    - **Search Outstanding Drop-Offs** to query all reservations that having been picked-up, waiting to be dropped-off.
        - For each reservation in the database, query and display the Customer, CustomerID, Reservation Start Date and Reservation End Date.
        - SELECT User.username, customerUserID, end_date, start_date, reservationID FROM Reservation INNER JOIN User ON Reservation.customerUserID = User.userId WHERE pickUpUserID IS NOT NULL AND dropOffUserID IS NULL;
    - If a specific Reservation is clicked
        - For the reservation, query and display *Customer Name, Total Deposit and* Total Rental Price.
            - SELECT Tool.purchase_price, Reservation.reservationID, User.first_name, User.last_name FROM Reservation INNER JOIN User ON Reservation.customerUserID = User.userId INNER JOIN IsOf ON IsOf.reservationID = Reservation.reservationID INNER JOIN Tool ON IsOf.toolNumber = Tool.toolNumber WHERE Reservation.reservationID = '$reservationID';
    - Specific reservation is entered and the button **Drop Off** is clicked
        - The **Rental Drop-Off Summary** is displayed
- Clerk entered a specific reservation number and clicked the button ***Confirm Drop Off*** on **Drop Off Reservations** page
    - Load **Rental Drop-Off Summary** page to query the selected reservation
        - For the specific reservation, populate text fields with *Reservation ID, Customer Name, Total Deposit, Total Rent Price and the Total Due, Tool IDs* being dropped-off, *Tool Names* being dropped-off, *Deposit Price* for each tool and *Rental Price* for each tool.
            - SELECT Tool.purchase_price, Reservation.reservationID, User.first_name, User.last_name FROM Reservation INNER JOIN User ON Reservation.customerUserID = User.userId INNER JOIN IsOf ON IsOf.reservationID = Reservation.reservationID INNER JOIN Tool ON IsOf.toolNumber = Tool.toolNumber WHERE Reservation.reservationID = '$reservationID';
    - If a ***Tool Name*** is clicked, the **Tool Details** are displayed for the tool
        - Call the Helper Tool Details Helper Task
    - Upon clicking ***Drop Off***
        - Update reservation record in the database

Table of Contents

        - UPDATE Reservation SET dropOffUserID='$userID' WHERE `reservationID` = '$reservationID'
    - OS Print dialog is opened.
    - Check if this is the 50th time the tool has been rented.
        - SELECT COUNT(Reservation.reservationID) FROM Reservation INNER JOIN IsOf ON Reservation.reservationID = IsOf.reservationID INNER JOIN Tool ON IsOf.toolNumber = Tool.toolNumber WHERE Tool.toolNumber = $toolNumber
        - If the count is 50, then mark it as for sale with clerkUserID as 1.
            - INSERT INTO SaleOrder (saleOrderID, clerkUserID, customerUserID, toolNumber, for_sale_date, sold_date) VALUES (NULL,1,NULL,$toolNumber,$for_sale_date,NULL);
- Upon printing summary, return to **Clerk Main Menu/Navigation**

Table of Contents

View Tool Details

Abstract Code

- Clerk or Customer clicked on View Tool Details**.**
    - Query the *Tool ID, Tool Type,* and *Accessories* for the selected tool.

```
SELECT Tool.toolNumber, type, sub_type, sub_option, manufacturer, width,
purchase_price, material, weight, power_source, length,
LadderTool.weight_capacity, LadderTool.step_count, Straight.rubber_feet,
Step.pail_shelf,Gun.capacity, Gun.gauge_rating, Socket.drive_size,
Socket.sae_size, Socket.deep_socket, Screwdriver.screw_size,
Hammer.anti_vibration, Plier.adjustable,
Ratchet.drive_size,WheelBarrow.bin_material, WheelBarrow.bin_volume,
WheelBarrow.wheel_count, Digging.blade_width, Digging.blade_length,
Prunning.blade_material,  Prunning.blade_length, Striking.head_weight,
Rake.tine_count,PowerTool.volt_rating, PowerTool.amp_rating,
PowerTool.max_rpm_rating,  PowerTool.min_rpm_rating,
Generator.power_rating, Saw.blade_size, Sander.dust_bag,
AirCompressor.tank_size, AirCompressor.pressure_rating,
Drill.min_torque_rating, Drill.max_torque_rating, Drill.adjustable_clutch,
Mixer.drum_size, Mixer.motor_rating,Accessory.description, Accessory.quantity
FROM Tool
LEFT OUTER JOIN LadderTool ON Tool.toolNumber = LadderTool.toolNumber
LEFT OUTER JOIN Straight ON LadderTool.toolNumber = Straight.toolNumber
LEFT OUTER JOIN Step ON LadderTool.toolNumber = Step.toolNumber
LEFT OUTER JOIN HandTool ON Tool.toolNumber = HandTool.toolNumber
LEFT OUTER JOIN Gun ON HandTool.toolNumber = Gun.toolNumber
LEFT OUTER JOIN Socket ON HandTool.toolNumber = Socket.toolNumber
LEFT OUTER JOIN ScrewDriver ON HandTool.toolNumber =
ScrewDriver.toolNumber
LEFT OUTER JOIN Hammer ON HandTool.toolNumber = Hammer.toolNumber
LEFT OUTER JOIN Plier ON HandTool.toolNumber = Plier.toolNumber
LEFT OUTER JOIN Ratchet ON HandTool.toolNumber = Ratchet.toolNumber
LEFT OUTER JOIN GardenTool ON Tool.toolNumber = GardenTool.toolNumber
LEFT OUTER JOIN WheelBarrow ON GardenTool.toolNumber =
WheelBarrow.toolNumber
LEFT OUTER JOIN Digging ON GardenTool.toolNumber = Digging.toolNumber
LEFT OUTER JOIN Prunning ON GardenTool.toolNumber =
Prunning.toolNumber
LEFT OUTER JOIN Striking ON GardenTool.toolNumber = Striking.toolNumber
LEFT OUTER JOIN Rake ON GardenTool.toolNumber = Rake.toolNumber
```

Table of Contents

```
LEFT OUTER JOIN PowerTool ON Tool.toolNumber = PowerTool.toolNumber
LEFT OUTER JOIN Generator ON PowerTool.toolNumber =
Generator.toolNumber
LEFT OUTER JOIN Saw ON PowerTool.toolNumber = Saw.toolNumber
LEFT OUTER JOIN Sander ON PowerTool.toolNumber = Sander.toolNumber
LEFT OUTER JOIN AirCompressor ON PowerTool.toolNumber =
AirCompressor.toolNumber
LEFT OUTER JOIN Drill ON PowerTool.toolNumber = Drill.toolNumber
LEFT OUTER JOIN Mixer ON PowerTool.toolNumber = Mixer.toolNumber
LEFT OUTER JOIN Accessory ON PowerTool.toolNumber =
Accessory.toolNumber
WHERE Tool.toolNumber = $toolNumber
```

- Total rental price is a summation of 15% of purchase price of each Tool in the reservation.
- Total deposit price is a summation of 40% of purchase price of each Tool in the reservation
- For All Tools
    - If [material] IS NOT “”, set the substring [material] = [material].
    - If [material] IS null or “”, set the substring [material] = “”.
- If Tool [type] is Ladder Tool
    - Short-Description = [sub_option] + [sub_type]
    - If Tool [sub-type] IS Straight
        - If [rubber-feet] IS true, set the substring [rubber-feet] = “ with rubber feet ”.
        - If [rubber-feet] IS false or null, set the substring [rubber-feet] = “”.

      Full-Description = [width] + “ in. W. ”+ [length] + “ in. L. ” + [weight] + “ lb. ” + [sub-option] + [sub-type] + [material] + [weight capacity] + “ lb. capacity “ + [step-count] + “ -step ” + [rubber feet] + “ by ” + [Manufacturer]
    - If Tool [sub-type] IS Step
        - If [pail-shelf] IS true, set the substring [rubber-feet] = “ with pail shelf ”.
        - If [pail-shelf] IS false or null, set the substring [rubber-feet] = “”.

      Full-Description = [width] + “ in. W. ”+ [length] + “ in. L. ” + [weight] + “ lb. ” + [sub-option] + [sub-type] + [material] + [weight capacity] + “ lb. capacity “ + [step-count] + “ -step ” + [pail-shelf] + “ by ” + [Manufacturer]
- If Tool [type] IS Hand Tool
    - Short-Description = [sub_option] + [sub_type]
    - If Tool [sub-type] IS Gun

      Full-Description = [width] + “ in. W. ”+ [length] + “ in. L. ” + [weight] + “ lb. ” + [sub-option] + [sub-type] + [material] + [gauge-rating] + “ G ” + [capacity] + “ by ” + [Manufacturer]
    - If Tool [sub-type] IS Socket
        - If [deep-socket] IS true, set substring [deep-socket] = “ deep-socket “
        - If [deep-socket] IS false or null, set substring [deep-socket] = ““

Table of Contents

Full-Description = [width] + " in. W. "+ [length] + " in. L. " + [weight] + " lb. " + [sub-option] + [sub-type] + [material] + [drive-size] + " in. " + [sae-size] + " in. "+ [deep-socket] + " by " + [Manufacturer]

- If Tool [sub-type] IS ScrewDriver
  Full-Description = [width] + " in. W. "+ [length] + " in. L. " + [weight] + " lb. " + [sub-option] + [sub-type] + [material] + [screw-size] + " # " + " by " + [Manufacturer]
- If Tool [sub-type] IS Hammer
  - If [anti-vibration] IS true, set substring [anti-vibration] = " anti-vibration "
  - If [anti-vibration] IS false or null, set substring [anti-vibration] = ""

  Full-Description = [width] + " in. W. "+ [length] + " in. L. " + [weight] + " lb. " + [sub-option] + [sub-type] + [material] + [anti-vibration] + " by " + [Manufacturer]
- If Tool [sub-type] IS Plier
  - If [adjustable] IS true, set substring [adjustable] = " adjustable "
  - If [adjustable] IS false, set substring [adjustable] = " non-adjustable "

  Full-Description = [width] + " in. W. "+ [length] + " in. L. " + [weight] + " lb. " + [sub-option] + [sub-type] + [material] + [adjustable] + " by " + [Manufacturer]
- If Tool [sub-type] IS Ratchet
  Full-Description = [width] + " in. W. "+ [length] + " in. L. " + [weight] + " lb. " + [sub-option] + [sub-type] + [material] + [drive-size] + " in. " + " by " + [Manufacturer]
- If Tool [sub-type] IS Wrench
  - If [drive-size] IS NOT null, set substring [drive-size] = [drive-size] + " in."
  - If [drive-size] IS null, set substring [drive-size] = ""

  Full-Description = [width] + " in. W. "+ [length] + " in. L. " + [weight] + " lb. " + [sub-option] + [sub-type] + [material] + [drive-size] + " by " + [Manufacturer]

- If Tool [type] is Garden Tool
  - Short-Description = [sub_option] + [sub_type]
  - If Tool [sub-type] IS Wheelbarrow
    - If [bin-volume] IS NOT null, set substring [bin-volume] = [bin-volume] + " cu ft."
    - If [bin-volume] IS null, set substring [blade-material] = ""

    Full-Description = [width] + " in. W. "+ [length] + " in. L. " + [weight] + " lb." + [sub-option] + [sub-type] + [material] + [handle-material] + " handle " + [wheel-count] + " wheeled" + [bin-material] + [bin-volume] + " by " + [Manufacturer]
  - If Tool [sub-type] IS Digging
    - If [blade-width] IS NOT null, set substring [blade-width] = [blade-width] + "in."
    - If [blade-width] IS null, set substring [blade-width] = ""

    Full-Description = [width] + " in. W. "+ [length] + " in. L. " + [weight] + " lb." + [sub-option] + [sub-type] + [material] + [handle-material] + " handle " + [blade-width] + [blade-length] + " in." + " by " + [Manufacturer]

Table of Contents

- If Tool [sub-type] IS Prunning
  - If [blade-material] IS NOT null, set substring [blade-material] = [blade-material] + " blade "
  - If [blade-material] IS null, set substring [blade-material] = ""

  Full-Description = [width] + " in. W. "+ [length] + " in. L. " + [weight] + " lb." + [sub-option] + [sub-type] + [material] + [handle-material] + " handle " + [blade-material] + [blade-length] + " in. " + " by " + [Manufacturer]
- If Tool [sub-type] IS Striking

  Full-Description = [width] + " in. W."+ [length] + " in. L." + [weight] + " lb." + [sub-option] + [sub-type] + [material] + [handle-material] + " handle" + [head-weight] + " lb. head weight " + " by " + [Manufacturer]
- If Tool [sub-type] IS Rake

  Full-Description = [width] + " in. W. "+ [length] + " in. L. " + [weight] + " lb." + [sub-option] + [sub-type] + [material] + [handle-material] + " handle " + [tine-count] + " tine" + " by " + [Manufacturer]

- If Tool [type] is Power Tool
  - Short-Description = [power_source] + [sub_option] + [and sub_type]
  - For all Power Tools
    - If [max-rpm-rating] IS NOT null, set substring [max-rpm-rating] = "/" + [max-rpm-rating]
    - If [max-rpm-rating] IS null, set substring [max-rpm-rating] = "".
  - If Tool [sub-type] IS Drill
    - If [adjustable-clutch] IS true, set substring [adjustable-clutch] = " adjustable clutch"
    - If [adjustable-clutch] IS false or null, set substring [adjustable-clutch] = "".
    - If [max-torque-rating] IS NOT null, set substring = "/" + [max-torque-rating]
    - If [max-torque-rating] IS null, set substring [max-torque-rating] = "".

    Full-Description = [width] + " in. W. "+ [length] + " in. L. " + [weight] + " lb. " + [power-source] + [sub-option] + [sub-type] + [volt] + " Volt" + [Amp] + " Amp" + [min-rpm-rating][max-rpm-rating] + " RPM" + [material] + [adjustable-clutch] + [min-torque-rating][max-torque-rating] + " ft-lb" + " by " + [Manufacturer]
  - If Tool [sub-type] IS Saw

    Full-Description = [width] + " in. W. "+ [length] + " in. L. " + [weight] + " lb." + [power-source] + [sub-option] + [sub-type] + [volt] + " Volt" + [Amp] + " Amp" + [min-rpm-rating][max-rpm-rating] + " RPM" + [material] + [blade-size] + " in." + " by " + [Manufacturer]
  - If Tool [sub-type] IS Sander
    - If [dust-bag] IS true, set substring [dust-bag] = " dust-bag".
    - If [dust-bag] IS false, set substring [dust-bag] = " no dust-bag".

    Full-Description = [width] + " in. W. "+ [length] + " in. L. " + [weight] + " lb." + [power-source] + [sub-option] + [sub-type] + [volt] + " Volt" + [Amp] + " Amp" +

[min-rpm-rating][max-rpm-rating] + " RPM" + [material] + [dust-bag] + " by " + [Manufacturer]

- If Tool [sub-type] IS Mixer
  Full-Description = [width] + " in. W. "+ [length] + " in. L. " + [weight] + " lb." + [power-source] + [sub-option] + [sub-type] + [volt] + " Volt" + [Amp] + " Amp" + [min-rpm-rating][max-rpm-rating] + " RPM" + [material] + [motor-rating] + " HP" + [drum-size] + " cu.ft." + " by " + [Manufacturer]
- If Tool [sub-type] IS Air-Compressor
  - If [pressure-rating] IS NOT null, set substring [pressure-rating] = [pressure-rating] + " psi"
  - If [pressure-rating] IS null, set substring [pressure-rating] = "".

  Full-Description = [width] + " in. W. "+ [length] + " in. L. " + [weight] + " lb." + [power-source] + [sub-option] + [sub-type] + [volt] + " Volt" + [Amp] + " Amp" + [min-rpm-rating][max-rpm-rating] + " RPM" + [material] + [tank-size] + " gal" + [pressure-rating] + " by " + [Manufacturer]
- If Tool [sub-type] IS Generator
  Full-Description = [width] + " in. W. "+ [length] + " in. L. " + [weight] + " lb." + [power-source] + [sub-option] + [sub-type] + [volt] + " Volt" + [Amp] + " Amp" + [min-rpm-rating][max-rpm-rating] + " RPM" + [material] + [power-rating] + "Watt" + " by " + [Manufacturer]
- If Accessory is associated with Tool [toolNumber], list all accessories as
  - If [acc-desc] IS " Batteries "
    Accessory-Description = [acc-quantity] + [volt-rating] + " Volt " + [amp-rating] + " Amp " + [battery-type] + [acc-desc]
  - If [acc-desc] IS NOT " Batteries "
    Accessory-Description = [acc-quantity] + [acc-desc]

Table of Contents

Add Tool Details

Abstract Code

- Clerk clicked ***Add New Tool*** from the **Clerk Main Menu/Navigation**.
    - **Tool Types** are displayed.
    - Clerk is prompted with a dropdown based UI in which he sets specifications for the tools that need to be added to the inventory. Each subsequent drop down results in querying data yielded from the query from the previous drop down.
    - Clerk picks the power source (ps_name):
        - SELECT DISTINCT * from ToolTypeOption where ps_name = $ps_name;
    - Clerk picks the power source (ps_name) and the tool-type (tt_name):
        - SELECT DISTINCT * from ToolTypeOption where ps_name = $ps_name AND tt_name=$tt_name'
    - Clerk picks the power source (ps_name) AND the tool-type (tt_name) AND the tool sub-type (tst_name):
        - SELECT DISTINCT * from ToolTypeOption where ps_name = $ps_name AND tt_name=$tt_name' AND tst_name = $tst_name;
    - Clerk picks the power source (ps_name) AND the tool-type (tt_name) AND the tool sub-type (tst_name) AND the tool sub-option (tst_name):
        - SELECT DISTINCT * from ToolTypeOption where ps_name = $ps_name AND tt_name=$tt_name' AND tst_name = $tst_name AND tso_name = tso_name;
    - Clerk additionally must enter the corresponding *Purchase Price, Manufacturer, Width, width Fraction, Width Unit, Length, Length Fraction, Length Unit, Weight* and optionally *Material.* *** ***THE FOLLOWING QUERY WILL BE REPEATED PER TOOL TO BE ADDED.****
        - START TRANSACTION;
          INSERT INTO Tool (type, sub_type, sub_option, manufacturer, width, purchase_price, material, weight, power_source, length)
          VALUES($tt_name, $tst_name, $tso_name, $manufacturer, $width, $purchase_price, $material, $weight, $ps_name, $length);
    - If *Hand Tool* is selected
        - Power-Source is set to *Manual*
        - Sub-types drop-down is loaded with *Screwdriver, Socket, Ratchet, Wrench, Pliers, Gun* and *Hammer.* *** ***THE FOLLOWING QUERY WILL BE REPEATED PER HAND TOOL TO BE ADDED.****
            - INSERT INTO HandTool(toolNumber) VALUES (LAST_INSERT_ID());
        - If Screwdriver Sub-Type is selected

Table of Contents

        - Sub-options drop-down for the Screwdriver are loaded with *phillips (cross), hex, torx* and *slotted(flat).*
        - Clerk must select sub-option and fill in *screw-size*.
            - INSERT INTO ScrewDriver(toolNumber, screw_size) VALUES(LAST_INSERT_ID(), $screw_size);
    - If Socket Sub-Type is selected
        - Sub-options drop-down for the Socket are loaded with *deep* and *standard.*
        - Clerk must select sub-option and fill in *drive-size, sae-size* and optionally *deep socket*.
            - INSERT INTO Socket(toolNumber,drive_size, sea_size,deep_socket) VALUES(LAST_INSERT_ID(),$drive_size , $sea_size, $deep_socket);
    - If Ratchet Sub-Type is selected
        - Sub-options drop-down for the Ratchet are loaded with *adjustable* and *fixed.*
        - Clerk must select sub-option and fill in *drive-size*.
            - INSERT INTO Ratchet(toolNumber,drive_size) VALUES(LAST_INSERT_ID(),$drive_size);
    - If Wrench Sub-Type is selected
        - Sub-options drop-down for the Wrench are loaded with *crescent, torque* and *pipe.*
        - Clerk must select sub-option and fill in drive_size
            - INSERT INTO Wrench(toolNumber,drive_size) VALUES(LAST_INSERT_ID(),$drive_size);
    - If Pliers Sub-Type is selected
        - Sub-options drop-down for the Pliers are loaded with *needle nose, cutting* and *crimper.*
        - Clerk must select sub-option and optionally fill in *adjustable*.
            - INSERT INTO Plier(toolNumber,adjustable) VALUES(LAST_INSERT_ID(),true);
    - If Gun Sub-Type is selected
        - Sub-options drop-down for the Gun are loaded with *nail* and *staple.*
        - Clerk must select sub-option and fill in *gauge-rating* and *capacity*.
            - INSERT INTO Gun(toolNumber,gauge_rating, capacity) VALUES(LAST_INSERT_ID(),$gauge_rating , $capacity);
    - If Hammer Sub-Type is selected
        - Sub-options drop-down for the Hammer are loaded with *claw, sledge* and *framing.*
        - Clerk must select sub-option and optionally fill in *anti-vibration*.

Table of Contents

        - INSERT INTO Hammer(toolNumber,anti_vibration) VALUES(LAST_INSERT_ID(),$anti_vibration);
- If *Garden Tools is* selected
    - Power-Source is set to *Manual*
    - Sub-types drop-down is loaded with *Digger, Pruner, Rakes, Wheelbarrows* and *Striking.*
        - *** ***THE FOLLOWING QUERY WILL BE REPEATED PER GARDEN TOOL TO BE ADDED.****
            - INSERT INTO GardenTool(toolNumber) VALUES (LAST_INSERT_ID());
    - If Digging Sub-Type is selected
        - Sub-options drop-down for the Digger is loaded with *pointed shovel, flat shovel, scoop shovel* and *edger.*
        - Clerk must select sub-option and fill in *handle-material, blade-length* and optionally *blade-width*.
            - INSERT INTO Digging(toolNumber,blade_width, blade_length) VALUES(LAST_INSERT_ID(), $blade_width, $blade_length);
    - If Prunning Sub-Type is selected
        - Sub-options drop-down for the Pruner are loaded with *sheer, loppers* and *hedge.*
        - Clerk must select sub-option and fill in *handle-material, blade-length* and optionally *blade-material*.
            - INSERT INTO Prunning(toolNumber,bin_material, blade_length) VALUES(LAST_INSERT_ID(), $blade_material, $blade_length);
    - If Rakes Sub-Type is selected
        - Sub-options drop-down for the Rakes are loaded with *leaf, landscaping* and *rock.*
        - Clerk must select sub-option and fill in *handle-material* and optionally *tine-count*.
            - INSERT INTO Rake(toolNumber, tine_count) VALUES(LAST_INSERT_ID(), $tine_count);
    - If Wheelbarrows Sub-Type is selected
        - Sub-options drop-down for the Wheelbarrows are loaded with *1-wheel* and *2-wheel.*
        - Clerk must select sub-option and fill in *handle-material, bin-material, wheel-count* and optionally *bin-volume*.
            - INSERT INTO WheelBarrow(toolNumber,bin_material, bin_volume, bin_count) VALUES(LAST_INSERT_ID(), $bin_material, $bin_volume, $wheel_count);
    - If Striking Sub-Type is selected

Table of Contents

- Sub-options drop-down for the Striking are loaded with *bar pry, rubber mallet, tamper, pick axe* and *single bit axe.*
- Clerk must select sub-option and fill in *handle-material* and *head-weight*.
    - INSERT INTO Striking(toolNumber, head_weight) VALUES(LAST_INSERT_ID(), $head_weight);

- If Ladder Tool is selected
    - Power-Source is set to *Manual*
    - *Clerk picks the step-c*ount ($step_count) and weight-capacity($weight_capacity) ****** THIS QUERY IS REPEATED PER LADDER TOOL ADDED***:***
        - INSERT INTO LadderTool (toolNumber, weight_capacity, step_count) VALUES (LAST_INSERT_ID(), $weight_capacity, $step_count);
        - *If Straight ladder is picked, and clerk optionally picks rubber-fee*t($rubber-feet):
            - INSERT INTO Straight (toolNumber, rubber_feet) VALUES (LAST_INSERT_ID(), $rubber_feet);
        - If Step ladder is picked, and clerk optionally picks pail_shelf($pail_shelf):
            - INSERT INTO Step(toolNumber, pail_shelf) VALUES (LAST_INSERT_ID(), $pail_shelf);

- If *Power Tool* is selected
    - Clerk is prompted to input volt_rating($volt_rating), amp_rating($amp_rating), max_rpm_rating($max_rpm_rating), min_rpm_rating($min_rpm_rating), battery_type ($battery_type) ***** THE FOLLOWING QUERY IS REPEATED PER POWER TOOL ADDED***:**
        - INSERT INTO PowerTool (toolNumber, volt_rating, amp_rating, max_rpm_rating, min_rpm_rating, battery_type) VALUES(LAST_INSERT_ID(), $volt_rating, $amp_rating, $max_rpm_rating, $min_rpm_rating);
        - Power-Source drop-down is loaded with *A/C, D/C* and *Gas*.
        - If Power-Source A/C is selected
            - Sub-types drop-down is loaded with *Drill, Saw, Sander, Air-Compressor* and *Mixer*.
            - If Drill Sub-Type is selected
                - Sub-options drop-down for the Drill is loaded with *driver* and *hammer, the clerk is prompted to input min_torque_rating and optionally input* max_torque_rating*, adjustable_clutch*

Table of Contents

- INSERT INTO Drill (toolNumber, min_torque_rating, max_torque_rating, adjustable_clutch) VALUES (LAST_INSERT_ID(), $min_torque_rating, $max_torque_rating, $adjustable_clutch);
- If Drill has accessories:
  - INSERT INTO Accessory(toolNumber, description, quantity) VALUES (LAST_INSERT_ID(), $description, $quantity);

- If Saw Sub-Type is selected
  - Sub-options drop-down for the Saw is loaded with *circular, reciprocating* and *jig.*
  - Clerk must select sub-option and fill *blade-size:*
  - INSERT INTO Saw (toolNumber, blade_size) VALUES (LAST_INSERT_ID(), $blade_size);
  - If Saw has accessories:
    - INSERT INTO Accessory(toolNumber, description, quantity) VALUES (LAST_INSERT_ID(), $description, $quantity);
- If Sander Sub-Type is selected
  - Sub-options drop-down for the Sander is loaded with *finish, sheet, belt* and *random orbital.*
  - Clerk optionally selects *dust-bag*.
    - INSERT INTO Sander(toolNumber, dust_bag) VALUES(LAST_INSERT_ID(), $dust_bag);
    - If Sander has accessories:
      - INSERT INTO Accessory(toolNumber, description, quantity) VALUES (LAST_INSERT_ID(), $description, $quantity);
- If Air-Compressor Sub-Type is selected
  - Sub-options drop-down for the Air-compressor is loaded with *reciprocating.*
  - Clerk must select tank_size and optionally select pressure_rating.
    - INSERT INTO AirCompressor(toolNumber, tank_size, pressure_rating) VALUES

Table of Contents

(LAST_INSERT_ID(), $tank_size, $pressure_rating);
- If AirCompressor has accessories:
  - INSERT INTO Accessory(toolNumber, description, quantity) VALUES (LAST_INSERT_ID(), $description, $quantity);

- If Mixer Sub-Type is selected
  - Sub-options drop-down for the Mixer is loaded with *concrete.*
  - Clerk must select motor-rating and *drum-size*.
    - INSERT INTO Mixer(toolNumber, motor_rating, drum_size) VALUES (LAST_INSERT_ID(), $motor_rating, $drum_size);
    - If Mixer has accessories:
      - INSERT INTO Accessory(toolNumber, description, quantity) VALUES (LAST_INSERT_ID(), $description, $quantity);

- If Power-Source D/C is selected,
  - Sub-types drop-down is loaded with *Drill, Saw* and *Sander*
  - If Drill Sub-Type is selected
    - Sub-options drop-down for the Drill is loaded with *driver* and *hammer, the clerk is prompted to input min_torque_rating and optionally input max_torque_rating, adjustable_clutch*
      - INSERT INTO Drill (toolNumber, min_torque_rating, max_torque_rating, adjustable_clutch) VALUES (LAST_INSERT_ID(), $min_torque_rating, $max_torque_rating, $adjustable_clutch);
      - Add battery as accessory, specify either ‘’Li-Ion” OR ‘NiCd’ OR ‘NiMh’
        - INSERT INTO Accessory(toolNumber, description, quantity) VALUES (LAST_INSERT_ID(), $battery_type, $quantity);
      - *If Drill has other accessories:*

Table of Contents

        - INSERT INTO Accessory(toolNumber, description, quantity) VALUES (LAST_INSERT_ID(), $description, $quantity);
    - If Saw Sub-Type is selected
        - Sub-options drop-down for the Saw is loaded with circular, reciprocating and jig.
        - Clerk must select sub-option and fill blade-size:
        - INSERT INTO Saw (toolNumber, blade_size) VALUES (LAST_INSERT_ID(), $blade_size);
        - Add battery as accessory, specify either ‘’Li-Ion” OR ‘NiCd’ OR ‘NiMh’
            - INSERT INTO Accessory(toolNumber, description, quantity) VALUES (LAST_INSERT_ID(), $battery_type, $quantity);
        - If Saw has other accessories:
            - INSERT INTO Accessory(toolNumber, description, quantity) VALUES (LAST_INSERT_ID(), $description, $quantity);
    - If Sander Sub-Type is selected
        - Sub-options drop-down for the Sander is loaded with *finish, sheet, belt* and *random orbital.*
        - Clerk optionally selects *dust-bag*.
            - INSERT INTO Sander(toolNumber, dust_bag) VALUES(LAST_INSERT_ID(), $dust_bag);
            - Add battery as accessory, specify either ‘’Li-Ion” OR ‘NiCd’ OR ‘NiMh’
                - INSERT INTO Accessory(toolNumber, description, quantity) VALUES (LAST_INSERT_ID(), $battery_type, $quantity);
            - If Sander has accessories:
                - INSERT INTO Accessory(toolNumber, description, quantity) VALUES (LAST_INSERT_ID(), $description, $quantity);
- If Power-Source Gas is selected

Table of Contents

- Sub-types drop-down is loaded with *Air-Compressor, Mixer* and *Generator.*
- If Air-Compressor Sub-Type is selected
  - Sub-options drop-down for the Air-compressor is loaded with *reciprocating.*
  - Clerk must select tank_size and optionally select *pressure_rating*.
    - INSERT INTO AirCompressor(toolNumber, tank_size, pressure_rating) VALUES (LAST_INSERT_ID(), $tank_size, $pressure_rating);
    - If AirCompressor has accessories:
      - INSERT INTO Accessory(toolNumber, description, quantity) VALUES (LAST_INSERT_ID(), $description, $quantity);
- If Mixer Sub-Type is selected
  - Sub-options drop-down for the Mixer is loaded with *concrete.*
  - Clerk must select motor-rating and drum-size.
    - INSERT INTO Mixer(toolNumber, motor_rating, drum_size) VALUES (LAST_INSERT_ID(), $motor_rating, $drum_size);
    - If Mixer has accessories:
      - INSERT INTO Accessory(toolNumber, description, quantity) VALUES (LAST_INSERT_ID(), $description, $quantity);
- If Generator Sub-Type is selected
  - Sub-options drop-down for the Generator is loaded with *electric.*
  - Clerk must select *power-rating*.
    - INSERT INTO Generator (toolNumber, power_rating) VALUES (LAST_INSERT_ID(), $power_rating);
    - If Generator has accessories:
      - INSERT INTO Accessory(toolNumber, description, quantity) VALUES

Table of Contents

(LAST_INSERT_ID(), $description, $quantity);

- Accessories are optional to be paired with power-tool:
  - Available accessories are *Drill bits, Saw Blade, Soft Case, Hard Case, D/C Batteries 7.2-80V, D/C Battery Charger, Safety Hat, Safety Pants, Safety Goggles, Safety Vest, Hose* and *Gas Tank*.
  - Clerk must enter *accessory-description* and *quantity* if an accessory is desired to be added to the power tool.

- If a Tool Name is clicked, the **Tool Details** are displayed for the tool
  - Execute the Tool Details Helper task
- Clerk additionally must enter the corresponding *Purchase Price, Manufacturer, Width, Width Fraction, Width Unit, Length, Length Fraction, Length Unit, Weight* and optionally *Material,* then clicks ***Confirm***.
  - The queries above are executed only after the confirm button is hit.
  - Clerk is returned to **Clerk Main Menu/Navigation**

Table of Contents

Request Service Order

Abstract Code

- Find all serviceable tools.
- query the Tool type, Sub-Type, Power Source, Sale Order, Reservation, and Service Order to find the available tools based on a search criteria from the customer.
    - Type is populated
        - SELECT DISTINCT tt_name
          FROM ToolTypeOption;
    - Power Source is populated
        - SELECT DISTINCT ps_name
          FROM ToolTypeOption
          WHERE tt_name = '$tt_name';
    - Sub Type is populated:
        - SELECT DISTINCT tst_name
          FROM ToolTypeOption
          WHERE tt_name = '$tt_name'
          AND ps_name = '$ps_name';
    - Customer fill in *start date, end date, custom search, type, sub-type* and *power source* input fields for the search criteria
    - Click the ***Search*** button
        - Display the *Tool ID, Description, Rental Price* and *Deposit Price* for all the tools that are available within the date range.
        - SELECT DISTINCT Tool.toolNumber, type, sub_type, sub_option, power_source,
          ROUND((purchase_price * 0.15), 2) AS rental_price,
          ROUND((purchase_price * 0.40), 2) AS deposit_price
          FROM Tool
          LEFT JOIN SaleOrder
          ON SaleOrder.toolNumber = Tool.toolNumber
          LEFT JOIN ServiceOrderRequest
          ON ServiceOrderRequest.toolNumber = Tool.toolNumber
          LEFT JOIN IsOf
          ON IsOf.toolNumber = Tool.toolNumber
          LEFT JOIN Reservation
          ON Reservation.reservationID = IsOf.reservationID
          WHERE SaleOrder.saleOrderID IS NULL
          AND Tool.toolNumber NOT IN
          (
                  SELECT Tool.toolNumber
                  FROM Tool

Table of Contents

```
        LEFT JOIN ServiceOrderRequest
        ON ServiceOrderRequest.toolNumber = Tool.toolNumber
        WHERE
        (
                (STR_TO_DATE('$searchStartDate', '%Y-%m-%d
%H:%i:%s') < ServiceOrderRequest.end_date)
                AND
                (STR_TO_DATE('$searchEndDate', '%Y-%m-%d
%H:%i:%s') > ServiceOrderRequest.start_date)
        )
)
AND Tool.toolNumber NOT IN
(
        SELECT Tool.toolNumber
        FROM Tool
        LEFT JOIN IsOf
        ON IsOf.toolNumber = Tool.toolNumber
        LEFT JOIN Reservation
        ON Reservation.reservationID = IsOf.reservationID
        WHERE
        (
                (STR_TO_DATE('$searchStartDate', '%Y-%m-%d
%H:%i:%s') < Reservation.end_date)
                AND
                (STR_TO_DATE('$searchEndDate', '%Y-%m-%d
%H:%i:%s') > Reservation.start_date)
        )
)
AND Tool.toolNumber NOT IN (
        SELECT Tool.toolNumber
        FROM Tool
        LEFT JOIN IsOf
        ON IsOf.toolNumber = Tool.toolNumber
        LEFT JOIN Reservation
        ON Reservation.reservationID = IsOf.reservationID
        WHERE Reservation.pickUpUserID IS NOT NULL
        GROUP BY Tool.toolNumber
        HAVING COUNT(IsOf.reservationID) >= 50
)
AND Tool.type = '$typeName'
AND Tool.sub_type = '$subTypeName'
AND Tool.power_source = '$powerSourceName'
AND Tool.sub_option LIKE '$customSearch%';
```

Table of Contents

- Clerk can click on “Service Tool” button to auto-populate Tool ID in the Tool ID field.
- If a Tool Name is clicked, the **Tool Details** are displayed for the tool.
    - Execute the Tool Detail Helper task
- Upon click “Confirm”
    - Insert a service order request record into the database
        - INSERT INTO ServiceOrderRequest (serviceOrderID, userID, toolNumber, cost, start_date, end_date) VALUES (NULL, $userID, $toolNumber, $cost, $start_date, $end_date);
    - Clerk is returned to **Clerk Main Menu/Navigation**

Table of Contents

View Service Status

Abstract Code

- Load all tools currently in Service.
  For each serviceable tool:
    - *serviceOrderID, toolNumber,* sub_option, sub_type, power_source*, Start Date, End Date, Repair Cost and* Clerk ID is displayed.
      SELECT ServiceOrderRequest.ServiceOrderID, Tool.toolNumber, sub_option, sub_type, power_source, start_date, end_date, cost, userID
      FROM ServiceOrderRequest
      INNER JOIN Tool ON Tool.toolNumber = ServiceOrderRequest.toolNumber
      WHERE (start_date < NOW() && NOW()> end_date);
        - If power_source == Manual → Description is a string concatenation of [sub_option] + [sub_type]
        - Else → Description is a string concatenation of [power_source] + [sub_option] + [and sub_type]
- If a Tool Name is clicked, the **Tool Details** task is executed.
- If Clerk clicks Fix-Now button
    - UPDATE ServiceOrderRequest SET end_date=NOW() WHERE serviceOrderID = $serviceOrderID
- Upon:
    - Closing of window, Clerk is returned to **Clerk Main Menu/Navigation.**

Table of Contents

Sell Tool

Abstract Code

- Load all available tools for sale
- query the Tool type, Sub-Type, Power Source, Sale Order, Reservation, and Service Order to find the available tools based on a search criteria from the customer.
    - Type is populated
        - SELECT DISTINCT tt_name
          FROM ToolTypeOption;
    - Power Source is populated
        - SELECT DISTINCT ps_name
          FROM ToolTypeOption
          WHERE tt_name = '$tt_name';
    - Sub Type is populated:
        - SELECT DISTINCT tst_name
          FROM ToolTypeOption
          WHERE tt_name = '$tt_name'
          AND ps_name = '$ps_name';
    - Customer fill in *custom search, type, sub-type* and *power source* input fields for the search criteria
    - Click the ***Search*** button
        - SELECT DISTINCT Tool.toolNumber, type, sub_type, sub_option, power_source,
          ROUND((purchase_price * 0.15), 2) AS rental_price,
          ROUND((purchase_price * 0.40), 2) AS deposit_price
          FROM Tool
          LEFT JOIN SaleOrder
          ON SaleOrder.toolNumber = Tool.toolNumber
          LEFT JOIN ServiceOrderRequest
          ON ServiceOrderRequest.toolNumber = Tool.toolNumber
          LEFT JOIN IsOf
          ON IsOf.toolNumber = Tool.toolNumber
          LEFT JOIN Reservation
          ON Reservation.reservationID = IsOf.reservationID
          WHERE SaleOrder.saleOrderID IS NULL
          AND Tool.toolNumber NOT IN
          (
                  SELECT Tool.toolNumber
                  FROM Tool
                  LEFT JOIN ServiceOrderRequest
                  ON ServiceOrderRequest.toolNumber = Tool.toolNumber

Table of Contents

```
        WHERE
        (
                (STR_TO_DATE('$searchStartDate', '%Y-%m-%d
%H:%i:%s') < ServiceOrderRequest.end_date)
                AND
                (STR_TO_DATE('$searchEndDate', '%Y-%m-%d
%H:%i:%s') > ServiceOrderRequest.start_date)
        )
)
AND Tool.toolNumber NOT IN
(
        SELECT Tool.toolNumber
        FROM Tool
        LEFT JOIN IsOf
        ON IsOf.toolNumber = Tool.toolNumber
        LEFT JOIN Reservation
        ON Reservation.reservationID = IsOf.reservationID
        WHERE
        (
                (STR_TO_DATE('$searchStartDate', '%Y-%m-%d
%H:%i:%s') < Reservation.end_date)
                AND
                (STR_TO_DATE('$searchEndDate', '%Y-%m-%d
%H:%i:%s') > Reservation.start_date)
        )
)
AND Tool.toolNumber NOT IN (
        SELECT Tool.toolNumber
        FROM Tool
        LEFT JOIN IsOf
        ON IsOf.toolNumber = Tool.toolNumber
        LEFT JOIN Reservation
        ON Reservation.reservationID = IsOf.reservationID
        WHERE Reservation.pickUpUserID IS NOT NULL
        GROUP BY Tool.toolNumber
        HAVING COUNT(IsOf.reservationID) >= 50
)
AND Tool.type = '$typeName'
AND Tool.sub_type = '$subTypeName'
AND Tool.power_source = '$powerSourceName'
AND Tool.sub_option LIKE '$customSearch%';
```

- If a Tool Name is clicked, the **Tool Details** are displayed for the tool
    - Execute the Helper Tool Details task

Table of Contents

- Upon click “Sell Tool”
    - INSERT INTO SaleOrder (saleOrderID, clerkUserID, customerUserID, toolNumber, for_sale_date, sold_date) VALUES (NULL,$userID,NULL,$toolNumber,$for_sale_date,NULL);
- Upon:
    - Closing of window, Clerk is returned to **Main Menu/Navigation.**

Table of Contents

View Sell Status

Abstract Code

- Load all items for sale or already sold
- For each tool for sale or sold:
    - *Sale ID, Current Status, Tool ID, Description, Customer ID, Sale Price, Sale Date* and Clerk ID is displayed.
      SELECT SaleOrder.saleOrderID, Tool.toolNumber, sub_option, sub_type, power_source, for_sale_date, clerkUserID,
      ROUND((purchase_price * 0.50), 2) AS sale_price
      FROM SaleOrder
      INNER JOIN Tool ON Tool.toolNumber = SaleOrder.toolNumber
        - If power_source == Manual → Description is a string concatenation of [sub_option] + [sub_type]
        - Else → Description is a string concatenation of [power_source] + [sub_option] + [and sub_type]
- If a Tool Name is clicked, the **Tool Details** are displayed for the tool.
- Upon:
    - Closing of window, Clerk is returned to **Main Menu/Navigation.**

Table of Contents

Generate Clerk Report

Abstract Code

- User clicked on ***Generate Reports*** link from **Clerk Main Menu**.
- If **Clerk Report** is selected all clerks' information is queried
    - Each clerk's *Clerk ID, First Name, Middle Name, Last Name, E-mail, Hire Date, Number of Pick-Ups, Number of Drop-Offs* and *Combined Total* are displayed.
      SELECT Clerk.userID, first_name, middle_name, last_name, email, date_hired, pickUpUserID, dropOffUserID
      FROM Clerk
      INNER JOIN User ON User.userID = Clerk.userID
      INNER JOIN Reservation ON Reservation.pickUpUserID = Clerk.userID
      OR Reservation.dropOffUserID = Clerk.userID
      WHERE Clerk.userID = '$userID';
        - Number of PickUps is calculated by counting all reservations where the Clerk is set as pickUpUserID.
        - Number of DropOffs is calculated by counting all reservations where the Clerk is set as dropOffUserID.
    - Upon clicking ***Back to Report Menu***
        - Clerk is returned to **Select a Report** menu
- Upon:
    - Closing of window, Clerk is returned to **Main Menu/Navigation.**

Table of Contents

Generate Customer Report

Abstract Code

- User clicked on ***Generate Reports*** link from **Clerk Main Menu**.
- If **Customer Report** is selected, all customers' information is queried
    - Each customer's *Customer ID, View Profile, First Name, Middle Name, Last Name, E-mail, Total # Reservations* and *Total # Tools Rented* are displayed.
      SELECT Customer.userID, first_name, middle_name, last_name, email,
      Reservation.customerUserID, Tool.toolNumber
      FROM Customer
      INNER JOIN User ON User.userID = Customer.userID
      INNER JOIN Reservation ON Reservation.customerUserID = Customer.userID
      INNER JOIN IsOf ON Reservation.reservationID=IsOf.reservationID
      INNER JOIN Tool ON IsOf.toolNumber=Tool.toolNumber
      WHERE Customer.userID = '$userID';
        - Number of Reservations is calculated by counting all reservations where the Customer is set as customerUserID.
        - Number of Tools Rented is calculated by counting all tools in the reservations where the toolNumber is contained inside the Reservation where the customer userId is set as customerUserID.
    - If *View Profile* is clicked, the customers **View Profile** if queried.
    - Upon clicking ***Back to Report Menu***
        - Clerk is returned to **Select a Report** menu
- Upon:
    - Closing of window, Clerk is returned to **Main Menu/Navigation.**

Table of Contents

Generate Tool Inventory Report

Abstract Code

- User clicked on ***Generate Reports*** link from **Clerk Main Menu**.
- If **Tool Inventory Report** is selected, all tools in the inventory are displayed
    - Create a hash table with the tool number as the key and an Object as the value which will contain the Current Status, Date, Short Description, Rental Profit, and Total Cost.
        - -- Select the tools that are AVAILABLE
        - SELECT DISTINCT Tool.toolNumber, type, sub_type, sub_option, power_source,
          FROM Tool
          LEFT JOIN SaleOrder
          ON SaleOrder.toolNumber = Tool.toolNumber
          LEFT JOIN ServiceOrderRequest
          ON ServiceOrderRequest.toolNumber = Tool.toolNumber
          LEFT JOIN IsOf
          ON IsOf.toolNumber = Tool.toolNumber
          LEFT JOIN Reservation
          ON Reservation.reservationID = IsOf.reservationID
          WHERE SaleOrder.saleOrderID IS NULL
          AND Tool.toolNumber NOT IN
          (
                  SELECT Tool.toolNumber
                  FROM Tool
                  LEFT JOIN ServiceOrderRequest
                  ON ServiceOrderRequest.toolNumber = Tool.toolNumber
                  WHERE
                  (
                          (STR_TO_DATE('$lastDayOfMonth, '%Y-%m-%d %H:%i:%s') < ServiceOrderRequest.end_date)
                          AND
                          (STR_TO_DATE('$lastDayOfMonth, '%Y-%m-%d %H:%i:%s') > ServiceOrderRequest.start_date)
)
)
          AND Tool.toolNumber NOT IN
          (
                  SELECT Tool.toolNumber
                  FROM Tool
                  LEFT JOIN IsOf

Table of Contents

```
        ON IsOf.toolNumber = Tool.toolNumber
        LEFT JOIN Reservation
        ON Reservation.reservationID = IsOf.reservationID
        WHERE
        (
                (STR_TO_DATE('$lastDayOfMonth, '%Y-%m-%d
%H:%i:%s') < Reservation.end_date)
                AND
                (STR_TO_DATE('$lastDayOfMonth, '%Y-%m-%d
%H:%i:%s') > Reservation.start_date)
        )
)
```

- Store the returned tool numbers from the query into the hash table as the key, the status set to Available, an empty date, and the short-description.
    - Create short-description:
        - If power_source == Manual → Description is a string concatenation of [sub_option] + [sub_type]
        - Else → Description is a string concatenation of [power_source] + [sub_option] + [and sub_type]
- -- Select the tools that are FOR-SALE
- SELECT Tool.toolNumber, type, sub_type, sub_option, power_source, for_sale_date
  FROM Tool
  JOIN SaleOrder
  ON SaleOrder.toolNumber = Tool.toolNumber
  WHERE SaleOrder.sold_date IS NULL;
- Add the tool to the hash table with the toolNumber as the key, a status of For-Sale, a date of for_sale_date, the short-description
    - Create short-description:
        - If power_source == Manual → Description is a string concatenation of [sub_option] + [sub_type]
        - Else → Description is a string concatenation of [power_source] + [sub_option] + [and sub_type]
- -- Select the tools that are SOLD
- SELECT Tool.toolNumber, type, sub_type, sub_option, power_source, sold_date
  FROM Tool
  JOIN SaleOrder
  ON SaleOrder.toolNumber = Tool.toolNumber
  WHERE SaleOrder.sold_date IS NOT NULL;
- Add the tool to the hash table with the toolNumber as the key, a status of Sold, a date of for_sale_date, the short-description
    - Create short-description:

Table of Contents

        - If power_source == Manual → Description is a string concatenation of [sub_option] + [sub_type]
        - Else → Description is a string concatenation of [power_source] + [sub_option] + [and sub_type]
    - -- Select the tools that are IN-REPAIR
    - SELECT Tool.toolNumber, type, sub_type, sub_option, Power_source, end_date
      FROM Tool
      INNER JOIN ServiceOrderRequest
      ON ServiceOrderRequest.toolNumber = Tool.toolNumber
      WHERE
      (
      (STR_TO_DATE('2017-10-31 00:00:00', '$lastDayOfMonth') < ServiceOrderRequest.end_date)
      AND
      (STR_TO_DATE('2017-10-31 00:00:00', '$lastDayOfMonth') > ServiceOrderRequest.start_date)
)
    - Add the tool to the hash table with the toolNumber as the key, a status of In-Repair, a date of end_date, the short-description
        - Create short-description:
            - If power_source == Manual → Description is a string concatenation of [sub_option] + [sub_type]
            - Else → Description is a string concatenation of [power_source] + [sub_option] + [and sub_type]
    - -- Select the tools that are RENTED
    - SELECT Tool.toolNumber, type, sub_type, sub_option, power_source, end_date
      FROM Tool
      INNER JOIN IsOf
      ON IsOf.toolNumber = Tool.toolNumber
      INNER JOIN Reservation
      ON Reservation.reservationID = IsOf.reservationID
      WHERE
      (
      (STR_TO_DATE('$lastDayOfMonth', '%Y-%m-%d %H:%i:%s') < Reservation.end_date)
      AND
      (STR_TO_DATE('$lastDayOfMonth', '%Y-%m-%d %H:%i:%s') > Reservation.start_date)
)
    - Add the tool to the hash table with the toolNumber as the key, a status of Rented, a date of end_date, the short-description

Table of Contents

- Create short-description:
  - If power_source == Manual → Description is a string concatenation of [sub_option] + [sub_type]
  - Else → Description is a string concatenation of [power_source] + [sub_option] + [and sub_type]
- -- Select the Total Cost per tool
- SELECT Tool.toolNumber,
  (SUM(COALESCE(ServiceOrderRequest.cost,0)) + Tool.purchase_price)
  AS total_cost
  FROM Tool
  LEFT OUTER JOIN ServiceOrderRequest
  ON ServiceOrderRequest.toolNumber = Tool.toolNumber
  GROUP BY Tool.toolNumber
- Insert the Total Cost per tool to the hash table
- -- Select the Rental Profit per tool
- SELECT Tool.toolNumber,
  (SUM(DATEDIFF(Reservation.end_date, Reservation.start_date)) *
  ROUND((purchase_price * 0.15), 2)) AS rental_profit
  FROM Tool
  INNER JOIN IsOf
  ON IsOf.toolNumber = Tool.toolNumber
  INNER JOIN Reservation
  ON Reservation.reservationID = IsOf.reservationID
  GROUP BY Tool.toolNumber
- Insert the Rental Profit per tool to the hash table
- Calculate the Total Profit per tool and add it to the hash table
  - Total Profit = Rental Profit - Total Cost

- Each Tool's *Tool ID, Current Status, Date, Description, Rental Profit, Total Cost,* and *Total Profit* are displayed in descending order by the total profit.
- If a Tool Name is clicked, the **Tool Details** are displayed for the tool.
- Upon clicking ***Back to Report Menu***
  - Clerk is returned to **Select a Report** menu

- Upon:
  - Closing of window, Clerk is returned to **Main Menu/Navigation.**

Table of Contents