Table of Contents

Table of Contents **1**

Abstract Code (AC) 2

Main Menu 2

Login 3

Registration 4

View Profile 4

Check Tool Availability 6

Tool Search 7

Full Tool Details 8

Make Reservation 8

Purchase Tool 10

Pick-Up Reservation 11

Drop-Off Reservation 12

Add New Tool 13

Service Order / Repair Tool 17

Service Status 18

Sell Tool 19

Sale Status 20

Generate Report 20

Abstract Code (AC)

Note: All variables listed in the code are in the MySql format of @variable name.

Main Menu

Abstract Code

- Customer
  - Show "Login", "View Profile", "Check Tool Availability", "Make Reservation", "Purchase Tool", and "Logout" link
- Clerk
  - "Login", "Pick-Up", "Drop-Off", "Add New Tool", "Service Tool", “Service Status”, "Sell Tool", "Sale Status", "Reports" and "Logout"
- Upon
  - User
    - Click ***Login*** button - Jump to the **Login** task
    - Click ***Logout*** button - Invalidate login session and go back to the **Login** form
  - Customer
    - Click ***View Profile*** button - Jump to the **View Profile** task
    - Click ***Check Tool Availability*** button - Jump to the **Check Tool Availability** task
    - Click ***Make Reservation*** button - Jump to the **Make Reservation** task
    - Click ***Purchase Tool*** button - Jump to the **Purchase Tool** task
    - Click ***Logout*** button - Jump to **Login** form
  - Clerk
    - Click ***Pick-up Reservation*** button - Jump to the **Pick-Up Reservation** task
    - Click ***Drop-off Reservation*** button - Jump to the **Drop-Off Reservation** task
    - Click ***Add New tool*** button - Jump to the **Add New Tool** task
    - Click ***Service Order*** button - Jump to the **Service Order / Repair Tool** task
    - Click ***Service Status*** button - Jump to the **Service Status** task
    - Click ***Sell Tool*** button - Jump to the **Sell Tool** task
    - Click ***Sale Status*** button - Jump to the **Sale Status** task
    - Click ***Reports*** button - Jump to the **Generate Report** task
    - Click ***Logout*** button - Jump to **Login** form

Table of Contents

Login

Abstract Code

- User clicks ***Login*** button from **Main Menu**
- Requires user to enter both *username* and *password*, before submission; Displays error if either field is empty.
- User select ***Customer*** radio button
    - User enters *username* and *password*
    - When User clicks ***Enter*** button
        - Find Username and verify password

```
SELECT password FROM Customer WHERE user_name=@username;
```

        - If Username does not exist
            - Go to **Registration** form
        - If Password does not match,
            - Redirect back to **Login**, Display prompt for user to enter password again
        - If Password matches
            - Go to **Main Menu**
- User selects ***Clerk radio button***
    - User enters *username* and *password*
    - When User clicks ***Enter*** button
    - Find Username in Clerk.Username

```
SELECT password FROM Clerk WHERE user_name=@clerkUsername;
```

    - Find Username in Customer.Username
        - If found
            - Display error message
    - If Username not found in either
        - Display error message
    - If Password does not match,
        - Redirect back to **Login**, Display prompt for user to enter password again
    - If Password matches
        - If TempPassword is not NULL
            - Display **Change Password** form
            - User clicks **Save button**
            - Update Password in Client

```
UPDATE Clerk SET password = @newpassword WHERE user_name = @clerkUsername;
UPDATE Clerk SET temp_password = NULL WHERE user_name = @clerkUsername;
```

        - Go to **Main Menu**

Table of Contents

Registration

Abstract Code

- User clicks ***Login*** on the **Customer Login** form with incorrect user information or clicks ***Create Account*** button
- User enters *username, email address, full name (first, middle, last), home phone, work phone, cell phone, address (City, State, Zipcode), password* (twice)
- User enters Credit Card details, *credit card name, credit card number, expiration month, expiration year, CVC*
- Validate that at least one primary phone number is entered.

```
INSERT INTO Address (street, city, state, zip) VALUES(@street, @city,@state, @zip);

INSERT INTO CreditCard (name, card_number, cvc, expiration_month, expiration_year)
VALUES (@cardname, @card_number, @cvc, @expiration_month, @expiration_year);
SET @ccId = last_insert_id();

INSERT INTO Customer (user_name, first_name, middle_name, last_name, email, password,
Address_Id, CreditCard_Id)
VALUES (@user_name, @first_name, @middle_name, @last_name, @email, @password,
@addressId, @ccId);

INSERT INTO PhoneNumber (area_code, number, extension, type, `primary`,
Customer_UserName) VALUES(@areacode, @pnum, @pext, @pType, @primPhone,
@user_name);
```

- User clicks ***Register*** button
    - If the username exists
        - Prompt user to pick another username
    - Else
        - Update Customer, Phone, and CreditCard
        - Go to **Login** form

View Profile

Abstract Code

- User clicks on ***View Profile*** button from the **Main Menu**
- Run the **View Profile** task: query for information about the user and his/her profile where $username is the current user using the HTTP Session/Cookie
    - **View Customer** task
        - Find the User using the Username; Display the email, full name
        - For each Phone under this Customer.Username

Table of Contents

        - Display the phone number
    - Find Address using Customer.Username; Display address (city, street, state, zip)

```
SELECT email, first_name, middle_name, last_name, p.area_code AS cellAc, p.extension AS cellExt, p.number AS cellNumber,
q.area_code AS workAc, q.extension AS workExt, q.number AS workNumber,
r.area_code AS homeAc, r.extension AS homeExt, r.number AS homeNumber,
city, street, zip, state
FROM Customer AS c
JOIN Address AS a ON a.id = c.Address_Id
LEFT OUTER JOIN PhoneNumber AS p ON c.CellPhoneNumber_Id = p.id
LEFT OUTER JOIN PhoneNumber AS q ON c.WorkPhoneNumber_Id = q.id
LEFT OUTER JOIN PhoneNumber AS r ON c.HomePhoneNumber_Id = r.id
WHERE user_name=@username;
```

- **View Reservations** task
    - Find rental history
        - For each Reservation for the Customer.Username
            - Display Reservation Number, StartDate, EndDate,
            - Calculate Number of Days from StartDate/EndDate
            - Find Tool based ON Tool.Number; Display Name, Deposit Price, Rental Price
            - Find Clerk based on Drop-off Clerk.Number, Pick-up Clerk.Number; Display Clerk.Name for both Pick-Up and Drop-Off clerk

```
SELECT r.id AS reservationId, start_date, end_date, DropOffClerk_UserName,
PickupClerk_UserName,
(DATEDIFF(end_date, start_date)) AS numDays,
sum(deposit_price) AS TotalDeposit,
sum(rental_price) AS TotalRental
FROM Tool
JOIN
(SELECT Tool_id,Reservations_Id FROM ToolReservations AS tr
WHERE tr.Reservations_Id in (SELECT id FROM Reservation AS r WHERE
r.Customer_UserName = @username)) AS rid
on rid.Tool_id = id
JOIN Reservation AS r ON r.id = Reservations_Id
ORDER BY booking_date;

# Get Tools associated with this reservation
SELECT cat.name AS category, ps.name AS powersource, st.name AS subtype, so.name AS suboption
FROM Tool AS t
JOIN
```

Table of Contents

```
(SELECT Tool_id, Reservations_Id FROM ToolReservations AS tr WHERE
tr.Reservations_Id = @ourReservationId) AS trr ON trr.Tool_id = t.id
JOIN SubOption AS so ON so.id = t.SubOption_Id
JOIN SubType AS st ON st.id = t.SubType_Id
JOIN Category AS cat ON cat.id = t.Category_Id
JOIN PowerSource AS ps ON ps.id = t.PowerSource_Id;
```

- Users clicks ***Close*** button, go to **Main Menu** form

Check Tool Availability

Abstract Code

- User clicks ***Check Availability*** link
- User inputs *Start Date, End Date, Keywords, Power Source, Sub-Type* and/or *Type*
- Run **Tool Search** task where Reservation.ToolNumber equals ToolNumbers from search and Reservation.StartDate...EndDate not equal to *Start Date, End Date*
    - If more than 10 tools are returned
        - Display prompt for user to specify more criteria
    - Else
        - For each Tool; Display Tool Number, Description (aggregate), Rental Price, Deposit Price
- User clicks ***Tool*** details link,
    - Find Tool using Tool.Number; Display Description, Deposit Price, Rental Price
    - Run **Full Tool Details** task; Display Tool ID, Tool Type, Short Description, Full Description (concatenated) , Deposit Price, Rental Price.
- Users clicks ***Close*** button, go to **Main Menu** form
- Check tool availability is done in 3 ways:

1. Summarized check tool availability with type, powersource, sub type, rental and deposit prices returned.
2. With full description and short description of tool returned.
3. With full description and short description of tool returned + ability to search using a keyword.

All of them, returning the tools that are available which are not in rent, service, or sale.

```
  -- Check Tool Availability
-- without keyword without details (full description, short)

SET @powersource :='Manual';
SET @subtype :='ScrewDriver';
```

Table of Contents

```
SET @category :='Hand';
SET @startdate :='2017-10-02 00:00:00';
SET @enddate :='2017-10-12 00:00:00';
SET @saledate :='2017-10-02 00:00:00';
SET @solddate :='2017-10-12 00:00:00';

SELECT
        tool.id AS toolId,
   category.name AS category,
   powersource.name AS powersource,
   subtype.name AS subtype, suboption.name AS suboption,
   rental_price,
   deposit_price
FROM Tool AS tool
JOIN SubOption AS suboption ON suboption.id = tool.SubOption_Id
JOIN SubType AS subtype ON subtype.id = tool.SubType_Id
JOIN PowerSource AS powersource ON powersource.id = tool.PowerSource_Id
JOIN Category AS category ON category.id = tool.Category_Id
WHERE subtype.name = @subtype
        AND powersource.name = @powersource
   AND category.name = @category
   AND tool.id NOT IN (
                SELECT Tool_Id AS toolId
                FROM ToolReservations AS toolreserv
                JOIN Reservation AS reservation ON reservation.id =
toolreserv.Reservations_Id
                WHERE reservation.start_date >= @startdate AND reservation.end_date <=
@enddate
                )
        AND tool.id NOT IN (
                SELECT Tool_Id AS toolId
      FROM SaleOrder AS saleorder
      WHERE saleorder.for_sale_date >= @saledate AND saleorder.sold_date is NOT NULL
AND saleorder.sold_date <= @solddate
   )
   AND tool.id NOT IN (
                SELECT Tool_Id AS toolId
                FROM ServiceOrder AS serviceorder
                WHERE serviceorder.start_date >= @startdate AND serviceorder.end_date <=
@enddate
                )

-- tool check availability with tool details and without keyword search

SET @powersource :='Manual';
SET @subtype :='ScrewDriver';
SET @category :='Hand';
```

Table of Contents

```
SET @startdate :='2017-10-02 00:00:00';
SET @enddate :='2017-10-12 00:00:00';
SET @saledate :='2017-10-02 00:00:00';
SET @solddate :='2017-10-12 00:00:00';

SELECT
        toolId,
   full_desc,
   short_desc,
   other_desc,
        powersource,
        subtype,
   suboption,
        rental_price,
        deposit_price,
        material,
        width,
        weight,
        length,
        manufacturer
FROM(
        SELECT
                tool.id AS toolId,
                category.name AS category,
                CONCAT(
                        COALESCE(IF(
                                        powersource.name='A/C','electric',
                                        IF(
                                                powersource.name='D/C', 'cordless',
                                                IF(powersource.name = 'Manual', '',
                                                powersource.name)
                                        )
                                ), ''), ' ',
                        COALESCE(suboption.name, ''), ' ',
                        COALESCE(subtype.name, '')
                ) AS short_desc,
                CONCAT(
                        COALESCE(width, ''), ' in. W x ',
                        COALESCE(length, ''), 'in. L ',
        CONCAT(COALESCE(weight, ''), 'lb'), ' ',
                        COALESCE(other_desc.other_desc, ''), ' ',
                        COALESCE(IF(
                                        powersource.name='A/C','electric',
                                        IF(
                                                powersource.name='D/C', 'cordless',
                                                IF(powersource.name = 'Manual', '',
                                                powersource.name)
```

Table of Contents

```
                                        )
                                ), ''), ' ',
                        COALESCE(suboption.name, ''), ' ',
                        COALESCE(subtype.name, ''), ' ',
                        CONCAT('By ', COALESCE(manufacturer, '')), ' '
                ) AS full_desc,
        other_desc.other_desc,
                powersource.name AS powersource,
                subtype.name AS subtype, suboption.name AS suboption,
                rental_price,
                deposit_price,
                material,
                width,
                weight,
                length,
                manufacturer
        FROM Tool AS tool
        JOIN SubOption AS suboption ON suboption.id = tool.SubOption_Id
        JOIN SubType AS subtype ON subtype.id = tool.SubType_Id
        JOIN PowerSource AS powersource ON powersource.id = tool.PowerSource_Id
        JOIN Category AS category ON category.id = tool.Category_Id
        JOIN (
                SELECT
                        id AS toolId,
                        CONCAT(
                                COALESCE(CONCAT(gauge_rating, 'G'), ''), ' ',
                                COALESCE(capacity, '')
                        ) AS other_desc
                FROM HandGun
                UNION
                SELECT
                        id AS toolId,
                        anti_vibration AS other_desc
                FROM HandHammer
                UNION
                SELECT
                        id AS toolId,
                        adjustable AS other_desc
                FROM HandPlier
                UNION
                SELECT
                        id AS toolId,
                        CONCAT(drive_size, 'in') AS other_desc
                FROM HandRatchet
                UNION
                SELECT
                        id AS toolId,
```

Table of Contents

```
        CONCAT('#', screw_size) AS other_desc
                FROM ScrewDriver
                UNION
                SELECT
                        id AS toolId,
                        CONCAT(
                                COALESCE(CONCAT(drive_size, 'in'), ''), ' ',
                                COALESCE(CONCAT(sae_size, 'in'), ''), ' ',
                                COALESCE(deep_socket, '')
                        ) AS other_desc
                FROM HandSocket
                UNION
                SELECT
                        pt.id AS toolId,
                        CONCAT(
                                COALESCE(CONCAT(tank_size, 'gal.'), ''), ' ',
                                COALESCE(CONCAT(pressure_rating, 'psi'), ''), ' ',
                                COALESCE(CONCAT(volt_rating, 'V'), ''), ' ',
                                COALESCE(CONCAT(amp_rating, 'Amp'), ''), ' ',
                                COALESCE(CONCAT(min_rpm_rating, 'min rpm'), ''), ' ',
                                COALESCE(CONCAT(max_rpm_rating, 'max rpm'), '')
                        ) AS other_desc
                FROM PowerAirCompressor pa
                JOIN PowerTool pt ON pt.id = pa.id
                UNION
                SELECT
                        pt.id AS toolId,
                        CONCAT(
                                COALESCE(adjustable_clutch, ''), ' ',
                                COALESCE(CONCAT(min_torque_rating, 'ft-lb'), ''), ' ',
                                COALESCE(CONCAT(max_torque_rating, 'ft-lb'), ''), ' ',
                                COALESCE(CONCAT(volt_rating, 'V'), ''), ' ',
                                COALESCE(CONCAT(amp_rating, 'Amp'), ''), ' ',
                                COALESCE(CONCAT(min_rpm_rating, 'min rpm'), ''), ' ',
                                COALESCE(CONCAT(max_rpm_rating, 'max rpm'), '')
                        ) AS other_desc
                FROM PowerDrill pa
                JOIN PowerTool pt ON pt.id = pa.id
                UNION
                SELECT
                        pt.id AS toolId,
        CONCAT(
                                COALESCE(CONCAT(power_rating, 'Watts'), ''), ' ',
                                COALESCE(CONCAT(volt_rating, 'V'), ''), ' ',
                                COALESCE(CONCAT(amp_rating, 'Amp'), ''), ' ',
                                COALESCE(CONCAT(min_rpm_rating, 'min rpm'), ''), ' ',
                                COALESCE(CONCAT(max_rpm_rating, 'max rpm'), '')
```

Table of Contents

```
                ) AS other_desc
        FROM PowerGenerator pa
        JOIN PowerTool pt ON pt.id = pa.id
        UNION
        SELECT
                pt.id AS toolId,
                CONCAT(
                        COALESCE(CONCAT(motor_rating, 'HP'), ''), ' ',
                        COALESCE(CONCAT(drum_size, 'cu ft.'), ''), ' ',
                        COALESCE(CONCAT(volt_rating, 'V'), ''), ' ',
                        COALESCE(CONCAT(amp_rating, 'Amp'), ''), ' ',
                        COALESCE(CONCAT(min_rpm_rating, 'min rpm'), ''), ' ',
                        COALESCE(CONCAT(max_rpm_rating, 'max rpm'), '')
                ) AS other_desc
        FROM PowerMixer pa
        JOIN PowerTool pt ON pt.id = pa.id
        UNION
        SELECT
                pt.id AS toolId,
CONCAT(
                        COALESCE(dust_bag, ''), ' ',
                        COALESCE(CONCAT(volt_rating, 'V'), ''), ' ',
                        COALESCE(CONCAT(amp_rating, 'Amp'), ''), ' ',
                        COALESCE(CONCAT(min_rpm_rating, 'min rpm'), ''), ' ',
                        COALESCE(CONCAT(max_rpm_rating, 'max rpm'), '')
                ) AS other_desc
        FROM PowerSander pa
        JOIN PowerTool pt ON pt.id = pa.id
        UNION
        SELECT
                pt.id AS toolId,
                CONCAT(
                        COALESCE(CONCAT(blade_size, 'in.'), ''), ' ',
                        COALESCE(CONCAT(volt_rating, 'V'), ''), ' ',
                        COALESCE(CONCAT(amp_rating, 'Amp'), ''), ' ',
                        COALESCE(CONCAT(min_rpm_rating, 'min rpm'), ''), ' ',
                        COALESCE(CONCAT(max_rpm_rating, 'max rpm'), '')
                ) AS other_desc
        FROM PowerSaw pa
        JOIN PowerTool pt ON pt.id = pa.id
        UNION
        SELECT
                ga.id AS toolId,
                CONCAT(
                        COALESCE(handle_material, ''), ' ',
                        COALESCE(CONCAT(blade_length, 'in.'), ''), ' ',
                        COALESCE(blade_material, '')
```

Table of Contents

```
                ) AS other_desc
        FROM PruningTool ga
        JOIN GardenTool gt ON gt.id = ga.id
        UNION
        SELECT
                ga.id AS toolId,
CONCAT(
                        COALESCE(handle_material, ''), ' ',
                        COALESCE(CONCAT(tine_count, 'tine'), '') ' '
                ) AS other_desc
        FROM RakeTool ga
        JOIN GardenTool gt ON gt.id = ga.id
        UNION
        SELECT
                ga.id AS toolId,
                CONCAT(
                        COALESCE(CONCAT(head_weight, 'lb.'), ''), ' ',
                        COALESCE(handle_material, '')
                ) AS other_desc
        FROM StrikingTool ga
        JOIN GardenTool gt ON gt.id = ga.id
        UNION
        SELECT
                ga.id AS toolId,
                CONCAT(
                        COALESCE(bin_material, ''), ' ',
                        COALESCE(CONCAT(wheel_count, 'wheeled'), ''), ' ',
                        COALESCE(CONCAT(bin_volume, 'cu ft.'), ''), ' ',
                        COALESCE(handle_material, '')
                ) AS other_desc
        FROM WheelBarrowTool ga
        JOIN GardenTool gt ON gt.id = ga.id
        UNION
        SELECT
                ga.id AS toolId,
                CONCAT(
                        COALESCE(CONCAT(blade_length, 'in.'), ''), ' ',
                        COALESCE(CONCAT(blade_width, 'in.'), ''), ' ',
                        COALESCE(handle_material, '')
                ) AS other_desc
        FROM DiggingTool ga
        JOIN GardenTool gt ON gt.id = ga.id
        UNION
        SELECT
                la.id AS toolId,
                CONCAT(
                        COALESCE(pail_shelf, ''), ' ',
```

Table of Contents

```
                        COALESCE(step_count, ''), ' ',
                        COALESCE(CONCAT(weight_capacity, 'Lb.'), '')
                ) AS other_desc
        FROM StepLadder la
        JOIN LadderTool lt ON lt.id = la.id
        UNION
        SELECT
                la.id AS toolId,
    CONCAT(
                        COALESCE(rubber_feet, ''), ' ',
                        COALESCE(step_count, ''), ' ',
                        COALESCE(CONCAT(weight_capacity, 'Lb.'), '')
                ) AS other_desc
        FROM StraightLadder la
        JOIN LadderTool lt ON lt.id = la.id
    ) other_desc
    ON other_desc.toolId = tool.id
) tools
WHERE
    subtype = @subtype
    AND powersource = @powersource
  AND category = @category
  AND toolId NOT IN (
        SELECT Tool_Id AS toolId
        FROM ToolReservations AS toolreserv
        JOIN Reservation AS reservation ON reservation.id =
toolreserv.Reservations_Id
        WHERE reservation.start_date >= @startdate AND reservation.end_date <=
@enddate
    )
    AND toolId NOT IN (
        SELECT Tool_Id AS toolId
    FROM SaleOrder AS saleorder
    WHERE saleorder.for_sale_date >= @saledate AND saleorder.sold_date is NOT NULL
AND saleorder.sold_date <= @solddate
  )
  AND toolId NOT IN (
        SELECT Tool_Id AS toolId
        FROM ServiceOrder AS serviceorder
        WHERE serviceorder.start_date >= @startdate AND serviceorder.end_date <=
@enddate
    )


-- Tool check availability with keyword Search and Full & short description

SET @powersource :='Manual';
```

Table of Contents

```
SET @subtype :='ScrewDriver';
SET @category :='Hand';
SET @startdate :='2017-10-02 00:00:00';
SET @enddate :='2017-10-12 00:00:00';
SET @saledate :='2017-10-02 00:00:00';
SET @solddate :='2017-10-12 00:00:00';

SET @PartialName :='Screwdriver';
SELECT
        toolId,
   full_desc,
   short_desc,
   other_desc,
        powersource,
        subtype,
   suboption,
        rental_price,
        deposit_price,
        material,
        width,
        weight,
        length,
        manufacturer
FROM(
        SELECT
                tool.id AS toolId,
                category.name AS category,
                CONCAT(
                        COALESCE(IF(
                                        powersource.name='A/C','electric',
                                        IF(
                                                powersource.name='D/C', 'cordless',
                                                IF(powersource.name = 'Manual', '',
                                                powersource.name)
                                        )
                                ), ''), ' ',
                        COALESCE(suboption.name, ''), ' ',
                        COALESCE(subtype.name, '')
                ) AS short_desc,
                CONCAT(
                        COALESCE(width, ''), ' in. W x',
                        COALESCE(length, ''), 'in. L ',
                        CONCAT(COALESCE(weight, ''), 'lb'), ' ',
                        COALESCE(other_desc.other_desc, ''), ' ',
                        COALESCE(IF(
                                        powersource.name='A/C','electric',
                                        IF(
```

Table of Contents

```
                                        powersource.name='D/C', 'cordless',
                                        IF(powersource.name = 'Manual', '',
                                        powersource.name)
                                    )
                                ), ''), ' ',
                        COALESCE(suboption.name, ''), ' ',
                        COALESCE(subtype.name, ''), ' ',
                        CONCAT('By ', COALESCE(manufacturer, '')), ' '
                ) AS full_desc,
        other_desc.other_desc,
                powersource.name AS powersource,
                subtype.name AS subtype, suboption.name AS suboption,
                rental_price,
                deposit_price,
                material,
                width,
                weight,
                length,
                manufacturer
        FROM Tool AS tool
        JOIN SubOption AS suboption ON suboption.id = tool.SubOption_Id
        JOIN SubType AS subtype ON subtype.id = tool.SubType_Id
        JOIN PowerSource AS powersource ON powersource.id = tool.PowerSource_Id
        JOIN Category AS category ON category.id = tool.Category_Id
        JOIN (
                SELECT
                        id AS toolId,
                        CONCAT(
                                COALESCE(CONCAT(gauge_rating, 'G'), ''), ' ',
                                COALESCE(capacity, '')
                        ) AS other_desc
                FROM HandGun
                UNION
                SELECT
                        id AS toolId,
                        anti_vibration AS other_desc
                FROM HandHammer
                UNION
                SELECT
                        id AS toolId,
                        adjustable AS other_desc
                FROM HandPlier
                UNION
                SELECT
                        id AS toolId,
                        CONCAT(drive_size, 'in') AS other_desc
                FROM HandRatchet
```

Table of Contents

```
        WHERE id = @toolid
        UNION
        SELECT
                id AS toolId,
CONCAT('#',screw_size) AS other_desc
        FROM ScrewDriver
        UNION
        SELECT
                id AS toolId,
                CONCAT(
                        COALESCE(CONCAT(drive_size, 'in'), ''), ' ',
                        COALESCE(CONCAT(sae_size, 'in'), ''), ' ',
                        COALESCE(deep_socket, '')
                ) AS other_desc
        FROM HandSocket
        UNION
        SELECT
                pt.id AS toolId,
                CONCAT(
                        COALESCE(CONCAT(tank_size, 'gal.'), ''), ' ',
                        COALESCE(CONCAT(pressure_rating, 'psi'), ''), ' ',
                        COALESCE(CONCAT(volt_rating, 'V'), ''), ' ',
                        COALESCE(CONCAT(amp_rating, 'Amp'), ''), ' ',
                        COALESCE(CONCAT(min_rpm_rating, 'min rpm'), ''), ' ',
                        COALESCE(CONCAT(max_rpm_rating, 'max rpm'), '')
                ) AS other_desc
        FROM PowerAirCompressor pa
        JOIN PowerTool pt ON pt.id = pa.id
        UNION
        SELECT
                pt.id AS toolId,
                CONCAT(
                        COALESCE(adjustable_clutch, ''), ' ',
                        COALESCE(CONCAT(min_torque_rating, 'ft-lb'), ''), ' ',
                        COALESCE(CONCAT(max_torque_rating, 'ft-lb'), ''), ' ',
                        COALESCE(CONCAT(volt_rating, 'V'), ''), ' ',
                        COALESCE(CONCAT(amp_rating, 'Amp'), ''), ' ',
                        COALESCE(CONCAT(min_rpm_rating, 'min rpm'), ''), ' ',
                        COALESCE(CONCAT(max_rpm_rating, 'max rpm'), '')
                ) AS other_desc
        FROM PowerDrill pa
        JOIN PowerTool pt ON pt.id = pa.id
        UNION
        SELECT
                pt.id AS toolId,
CONCAT(
                        COALESCE(CONCAT(power_rating, 'Watts'), ''), ' ',
```

Table of Contents

```
                        COALESCE(CONCAT(volt_rating, 'V'), ''), ' ',
                        COALESCE(CONCAT(amp_rating, 'Amp'), ''), ' ',
                        COALESCE(CONCAT(min_rpm_rating, 'min rpm'), ''), ' ',
                        COALESCE(CONCAT(max_rpm_rating, 'max rpm'), '')
                ) AS other_desc
        FROM PowerGenerator pa
        JOIN PowerTool pt ON pt.id = pa.id
        UNION
        SELECT
                pt.id AS toolId,
                CONCAT(
                        COALESCE(CONCAT(motor_rating, 'HP'), ''), ' ',
                        COALESCE(CONCAT(drum_size, 'cu ft.'), ''), ' ',
                        COALESCE(CONCAT(volt_rating, 'V'), ''), ' ',
                        COALESCE(CONCAT(amp_rating, 'Amp'), ''), ' ',
                        COALESCE(CONCAT(min_rpm_rating, 'min rpm'), ''), ' ',
                        COALESCE(CONCAT(max_rpm_rating, 'max rpm'), '')
                ) AS other_desc
        FROM PowerMixer pa
        JOIN PowerTool pt ON pt.id = pa.id
        UNION
        SELECT
                pt.id AS toolId,
CONCAT(
                        COALESCE(dust_bag, ''), ' ',
                        COALESCE(CONCAT(volt_rating, 'V'), ''), ' ',
                        COALESCE(CONCAT(amp_rating, 'Amp'), ''), ' ',
                        COALESCE(CONCAT(min_rpm_rating, 'min rpm'), ''), ' ',
                        COALESCE(CONCAT(max_rpm_rating, 'max rpm'), '')
                ) AS other_desc
        FROM PowerSander pa
        JOIN PowerTool pt ON pt.id = pa.id
        UNION
        SELECT
                pt.id AS toolId,
                CONCAT(
                        COALESCE(CONCAT(blade_size, 'in'), ''), ' ',
                        COALESCE(CONCAT(volt_rating, 'V'), ''), ' ',
                        COALESCE(CONCAT(amp_rating, 'Amp'), ''), ' ',
                        COALESCE(CONCAT(min_rpm_rating, 'min rpm'), ''), ' ',
                        COALESCE(CONCAT(max_rpm_rating, 'max rpm'), '')
                ) AS other_desc
        FROM PowerSaw pa
        JOIN PowerTool pt ON pt.id = pa.id
        UNION
        SELECT
                ga.id AS toolId,
```

Table of Contents

```
                CONCAT(
                        COALESCE(handle_material, ''), ' ',
                        COALESCE(CONCAT(blade_length, 'in.'), ''), ' ',
                        COALESCE(blade_material, '')
                ) AS other_desc
        FROM PruningTool ga
        JOIN GardenTool gt ON gt.id = ga.id
        UNION
        SELECT
                ga.id AS toolId,
CONCAT(
                        COALESCE(handle_material, ''), ' ',
                        COALESCE(CONCAT(tine_count, 'tine'), ''), ' '
                ) AS other_desc
        FROM RakeTool ga
        JOIN GardenTool gt ON gt.id = ga.id
        UNION
        SELECT
                ga.id AS toolId,
                CONCAT(
                        COALESCE(CONCAT(head_weight, 'lb.'), ''), ' ',
                        COALESCE(handle_material, '')
                ) AS other_desc
        FROM StrikingTool ga
        JOIN GardenTool gt ON gt.id = ga.id
        UNION
        SELECT
                ga.id AS toolId,
                CONCAT(
                        COALESCE(bin_material, ''), ' ',
                        COALESCE(CONCAT(wheel_count, 'wheeled'), ''), ' ',
                        COALESCE(CONCAT(bin_volume, 'cu ft.'), ''), ' ',
                        COALESCE(handle_material, '')
                ) AS other_desc
        FROM WheelBarrowTool ga
        JOIN GardenTool gt ON gt.id = ga.id
        UNION
        SELECT
                ga.id AS toolId,
                CONCAT(
                        COALESCE(CONCAT(blade_length, 'in'), ''), ' ',
                        COALESCE(CONCAT(blade_width, 'in') , ''), ' ',
                        COALESCE(handle_material, '')
                ) AS other_desc
        FROM DiggingTool ga
        JOIN GardenTool gt ON gt.id = ga.id
        UNION
```

Table of Contents

```
                SELECT
                        la.id AS toolId,
                        CONCAT(
                                COALESCE(pail_shelf, ''), ' ',
                                COALESCE(step_count, ''), ' ',
                                COALESCE(CONCAT(weight_capacity, 'Lb.'), '')
                        ) AS other_desc
                FROM StepLadder la
                JOIN LadderTool lt ON lt.id = la.id
                UNION
                SELECT
                        la.id AS toolId,
        CONCAT(
                                COALESCE(rubber_feet, ''), ' ',
                                COALESCE(step_count, ''), ' ',
                                COALESCE(CONCAT(weight_capacity, 'Lb.'), '')
                        ) AS other_desc
                FROM StraightLadder la
                JOIN LadderTool lt ON lt.id = la.id
        ) other_desc
        ON other_desc.toolId = tool.id
) tools
WHERE
        (
                other_desc LIKE CONCAT('%', @PartialName,'%') OR
                full_desc LIKE CONCAT('%', @PartialName,'%') OR
                short_desc LIKE CONCAT('%', @PartialName,'%')
        )
        AND subtype = @subtype
        AND powersource = @powersource
   AND category = @category
   AND toolId NOT IN (
                SELECT Tool_Id AS toolId
                FROM ToolReservations AS toolreserv
                JOIN Reservation AS reservation ON reservation.id =
toolreserv.Reservations_Id
                WHERE reservation.start_date >= @startdate AND reservation.end_date <=
@enddate
        )
        AND toolId NOT IN (
                SELECT Tool_Id AS toolId
      FROM SaleOrder AS saleorder
      WHERE saleorder.for_sale_date >= @saledate AND saleorder.sold_date is NOT NULL
AND saleorder.sold_date <= @solddate
   )
   AND toolId NOT IN (
                SELECT Tool_Id AS toolId
```

Table of Contents

```
                FROM ServiceOrder AS serviceorder
                WHERE serviceorder.start_date >= @startdate AND serviceorder.end_date <=
@enddate
        )
```

Tool Search

Abstract Code

- ** This task encompasses the Search logic included in many other task*
- Search is done with keyword search or without keyword search. For example: keyword "hex" should return all tools that has "hex" in its data.
- User clicks ***Search*** button
    - For each Tool that matches the ToolNumber.ToolType, PowerSource, SubTypes, and/or keyword search.
        - Return Tool.Number, Tool.Name, RentalPrice, and DepositPrice
        - Return full tool details description and short description also, with each tool type and unit and manufacturer.

```
-- tool search with getting all description full and short -- without keyword search

SET @powersource :='Manual';
SET @subtype :='ScrewDriver';
SET @category :='Hand';

SELECT
        toolId,
   full_desc,
   short_desc,
   other_desc,
        powersource,
        subtype,
   suboption,
        rental_price,
        deposit_price,
        material,
        width,
        weight,
        length,
        manufacturer
FROM(
        SELECT
                tool.id AS toolId,
```

Table of Contents

```
            category.name AS category,
            CONCAT(
                    COALESCE(IF(
                            powersource.name='A/C','electric',
                            IF(
                                    powersource.name='D/C', 'cordless',
                                    IF(powersource.name = 'Manual', '',
        powersource.name)
                            )
                    ), ''), ' ',
                    COALESCE(suboption.name, ''), ' ',
                    COALESCE(subtype.name, '')
            ) AS short_desc,
            CONCAT(
                    COALESCE(width, ''), ' in. W x',
                    COALESCE(length, ''), 'in. L ',
    CONCAT(COALESCE(weight, ''), 'lb'), ' ',
                    COALESCE(other_desc.other_desc, ''), ' ',
                    COALESCE(IF(
                            powersource.name='A/C','electric',
                            IF(
                                    powersource.name='D/C', 'cordless',
                                    IF(powersource.name = 'Manual', '',
        powersource.name)
                            )
                    ), ''), ' ',
                    COALESCE(suboption.name, ''), ' ',
                    COALESCE(subtype.name, ''), ' ',
                    CONCAT('By ', COALESCE(manufacturer, '')), ' '
            ) AS full_desc,
other_desc.other_desc,
            powersource.name AS powersource,
            subtype.name AS subtype, suboption.name AS suboption,
            rental_price,
            deposit_price,
            material,
            width,
            weight,
            length,
            manufacturer
    FROM Tool AS tool
    JOIN SubOption AS suboption ON suboption.id = tool.SubOption_Id
    JOIN SubType AS subtype ON subtype.id = tool.SubType_Id
    JOIN PowerSource AS powersource ON powersource.id = tool.PowerSource_Id
    JOIN Category AS category ON category.id = tool.Category_Id
    JOIN (
            SELECT
```

Table of Contents

```
                id AS toolId,
                CONCAT(
                        COALESCE(CONCAT(gauge_rating, 'G'), ''), ' ',
                        COALESCE(capacity, '')
                ) AS other_desc
        FROM HandGun
        UNION
        SELECT
                id AS toolId,
                anti_vibration AS other_desc
        FROM HandHammer
        UNION
        SELECT
                id AS toolId,
                adjustable AS other_desc
        FROM HandPlier
        UNION
        SELECT
                id AS toolId,
                CONCAT(drive_size, 'in') AS other_desc
        FROM HandRatchet
        UNION
        SELECT
                id AS toolId,
CONCAT('#', screw_size) AS other_desc
        FROM ScrewDriver
        UNION
        SELECT
                id AS toolId,
                CONCAT(
                        COALESCE(CONCAT(drive_size, 'in'), ''), ' ',
                        COALESCE(CONCAT(sae_size, 'in'), ''), ' ',
                        COALESCE(deep_socket, '')
                ) AS other_desc
        FROM HandSocket
        UNION
        SELECT
                pt.id AS toolId,
                CONCAT(
                        COALESCE(CONCAT(tank_size, 'gal'), ''), ' ',
                        COALESCE(CONCAT(pressure_rating, 'psi'), ''), ' ',
                        COALESCE(CONCAT(volt_rating, 'V'), ''), ' ',
                        COALESCE(CONCAT(amp_rating, 'Amp'), ''), ' ',
                        COALESCE(CONCAT(min_rpm_rating, 'min rpm'), ''), ' ',
                        COALESCE(CONCAT(max_rpm_rating, 'max rpm'), '')
                ) AS other_desc
        FROM PowerAirCompressor pa
```

Table of Contents

```
        JOIN PowerTool pt ON pt.id = pa.id
        UNION
        SELECT
            pt.id AS toolId,
            CONCAT(
                COALESCE(adjustable_clutch, ''), ' ',
                COALESCE(CONCAT(min_torque_rating, 'min ft-lb'), ''), ' ',
                COALESCE(CONCAT(max_torque_rating, 'max ft-lb'), ''), ' ',
                COALESCE(CONCAT(volt_rating, 'V'), ''), ' ',
                COALESCE(CONCAT(amp_rating, 'Amp'), ''), ' ',
                COALESCE(CONCAT(min_rpm_rating, 'min rpm'), ''), ' ',
                COALESCE(CONCAT(max_rpm_rating, 'max rpm'), '')
            ) AS other_desc
        FROM PowerDrill pa
        JOIN PowerTool pt ON pt.id = pa.id
        UNION
        SELECT
            pt.id AS toolId,
CONCAT(
                COALESCE(CONCAT(power_rating, 'Watts'), ''), ' ',
                COALESCE(CONCAT(volt_rating, 'V'), ''), ' ',
                COALESCE(CONCAT(amp_rating, 'Amp'), ''), ' ',
                COALESCE(CONCAT(min_rpm_rating, 'min rpm'), ''), ' ',
                COALESCE(CONCAT(max_rpm_rating, 'max rpm'), '')
            ) AS other_desc
        FROM PowerGenerator pa
        JOIN PowerTool pt ON pt.id = pa.id
        UNION
        SELECT
            pt.id AS toolId,
            CONCAT(
                COALESCE(CONCAT(motor_rating, 'HP'), ''), ' ',
                COALESCE(CONCAT(drum_size, 'cu ft.'), ''), ' ',
                COALESCE(CONCAT(volt_rating, 'V'), ''), ' ',
                COALESCE(CONCAT(amp_rating, 'Amp'), ''), ' ',
                COALESCE(CONCAT(min_rpm_rating, 'min rpm'), ''), ' ',
                COALESCE(CONCAT(max_rpm_rating, 'max rpm'), '')
            ) AS other_desc
        FROM PowerMixer pa
        JOIN PowerTool pt ON pt.id = pa.id
        UNION
        SELECT
            pt.id AS toolId,
CONCAT(
                COALESCE(dust_bag, ''), ' ',
                COALESCE(CONCAT(volt_rating, 'V'), ''), ' ',
                COALESCE(CONCAT(amp_rating, 'Amp'), ''), ' ',
```

Table of Contents

```
                        COALESCE(CONCAT(min_rpm_rating, 'min rpm'), ''), ' ',
                        COALESCE(CONCAT(max_rpm_rating, 'max rpm'), '')
                ) AS other_desc
        FROM PowerSander pa
        JOIN PowerTool pt ON pt.id = pa.id
        UNION
        SELECT
                pt.id AS toolId,
                CONCAT(
                        COALESCE(CONCAT(blade_size, 'in'), ''), ' ',
                        COALESCE(CONCAT(volt_rating, 'V'), ''), ' ',
                        COALESCE(CONCAT(amp_rating, 'Amp'), ''), ' ',
                        COALESCE(CONCAT(min_rpm_rating, 'min rpm'), ''), ' ',
                        COALESCE(CONCAT(max_rpm_rating, 'max rpm'), '')
                ) AS other_desc
        FROM PowerSaw pa
        JOIN PowerTool pt ON pt.id = pa.id
        UNION
        SELECT
                ga.id AS toolId,
                CONCAT(
                        COALESCE(handle_material, ''), ' ',
                        COALESCE(CONCAT(blade_length, 'in.'), ''), ' ',
                        COALESCE(blade_material, '')
                ) AS other_desc
        FROM PruningTool ga
        JOIN GardenTool gt ON gt.id = ga.id
        UNION
        SELECT
                ga.id AS toolId,
CONCAT(
                        COALESCE(handle_material, ''), ' ',
                        COALESCE(CONCAT(tine_count, 'tine'), ''), ' '
                ) AS other_desc
        FROM RakeTool ga
        JOIN GardenTool gt ON gt.id = ga.id
        UNION
        SELECT
                ga.id AS toolId,
                CONCAT(
                        COALESCE(CONCAT(head_weight, 'lb.'), ''), ' ',
                        COALESCE(handle_material, '')
                ) AS other_desc
        FROM StrikingTool ga
        JOIN GardenTool gt ON gt.id = ga.id
        UNION
        SELECT
```

Table of Contents

```
                ga.id AS toolId,
                CONCAT(
                        COALESCE(bin_material, ''), ' ',
                        COALESCE(CONCAT(wheel_count, 'wheeled'), ''), ' ',
                        COALESCE(CONCAT(bin_volume, 'cu ft.'), ''), ' ',
                        COALESCE(handle_material, '')
                ) AS other_desc
        FROM WheelBarrowTool ga
        JOIN GardenTool gt ON gt.id = ga.id
        UNION
        SELECT
                ga.id AS toolId,
                CONCAT(
                        COALESCE(CONCAT(blade_length, 'in.'), ''), ' ',
                        COALESCE(CONCAT(blade_width, 'in.'), ''), ' ',
                        COALESCE(handle_material, '')
                ) AS other_desc
        FROM DiggingTool ga
        JOIN GardenTool gt ON gt.id = ga.id
        UNION
        SELECT
                la.id AS toolId,
                CONCAT(
                        COALESCE(pail_shelf, ''), ' ',
                        COALESCE(step_count, ''), ' ',
                        COALESCE(CONCAT(weight_capacity, 'Lb.'), '')
                ) AS other_desc
        FROM StepLadder la
        JOIN LadderTool lt ON lt.id = la.id
        UNION
        SELECT
                la.id AS toolId,
    CONCAT(
                        COALESCE(rubber_feet, ''), ' ',
                        COALESCE(step_count, ''), ' ',
                        COALESCE(CONCAT(weight_capacity, 'Lb.'), '')
                ) AS other_desc
        FROM StraightLadder la
        JOIN LadderTool lt ON lt.id = la.id
    ) other_desc
    ON other_desc.toolId = tool.id
) tools
WHERE
    subtype = @subtype
    AND powersource = @powersource
  AND category = @category
```

Table of Contents

```
-- tool search with getting all description full and short -- WITH keyword search

SET @powersource :='Manual';
SET @subtype :='ScrewDriver';
SET @category :='Hand';
SET @PartialName :='Hex';

SELECT
        toolId,
   full_desc,
   short_desc,
   other_desc,
        powersource,
        subtype,
   suboption,
        rental_price,
        deposit_price,
        material,
        width,
        weight,
        length,
        manufacturer
FROM(
        SELECT
                tool.id AS toolId,
                category.name AS category,
                CONCAT(
                        COALESCE(
                                IF(
                                        powersource.name='A/C','electric',
                                        IF(
                                                powersource.name='D/C', 'cordless',
                                                IF(powersource.name = 'Manual', '',
                                                powersource.name)
                                        )
                                ), ''), ' ',
                        COALESCE(suboption.name, ''), ' ',
                        COALESCE(subtype.name, '')
                ) AS short_desc,
                CONCAT(
                        COALESCE(width, ''), ' in. W x',
                        COALESCE(length, ''), 'in. L ',
        CONCAT(COALESCE(weight, ''), 'lb'), ' ',
                        COALESCE(other_desc.other_desc, ''), ' ',
                        COALESCE(IF(
                                        powersource.name='A/C','electric',
```

Table of Contents

```
                                IF(
                                        powersource.name='D/C', 'cordless',
                                        IF(powersource.name = 'Manual', '',
                                        powersource.name)
                                )
                        ), ''), ' ',
                COALESCE(suboption.name, ''), ' ',
                COALESCE(subtype.name, ''), ' ',
                CONCAT('By ', COALESCE(manufacturer, '')), ' '
        ) AS full_desc,
other_desc.other_desc,
        powersource.name AS powersource,
        subtype.name AS subtype, suboption.name AS suboption,
        rental_price,
        deposit_price,
        material,
        width,
        weight,
        length,
        manufacturer
  FROM Tool AS tool
  JOIN SubOption AS suboption ON suboption.id = tool.SubOption_Id
  JOIN SubType AS subtype ON subtype.id = tool.SubType_Id
  JOIN PowerSource AS powersource ON powersource.id = tool.PowerSource_Id
  JOIN Category AS category ON category.id = tool.Category_Id
  JOIN (
        SELECT
                id AS toolId,
                CONCAT(
                        COALESCE(CONCAT(gauge_rating, 'G'), ''), ' ',
                        COALESCE(capacity, '')
                ) AS other_desc
        FROM HandGun
        UNION
        SELECT
                id AS toolId,
                anti_vibration AS other_desc
        FROM HandHammer
        UNION
        SELECT
                id AS toolId,
                adjustable AS other_desc
        FROM HandPlier
        UNION
        SELECT
                id AS toolId,
                CONCAT(drive_size, 'in') AS other_desc
```

Table of Contents

```
        FROM HandRatchet
        UNION
        SELECT
                id AS toolId,
CONCAT('#', screw_size) AS other_desc
        FROM ScrewDriver
        UNION
        SELECT
                id AS toolId,
                CONCAT(
                        COALESCE(CONCAT(drive_size, 'in'), ''), ' ',
                        COALESCE(CONCAT(sae_size, 'in'), ''), ' ',
                        COALESCE(deep_socket, '')
                ) AS other_desc
        FROM HandSocket
        UNION
        SELECT
                pt.id AS toolId,
                CONCAT(
                        COALESCE(CONCAT(tank_size, 'gal'), ''), ' ',
                        COALESCE(CONCAT(pressure_rating, 'psi'), ''), ' ',
                        COALESCE(CONCAT(volt_rating, 'V'), ''), ' ',
                        COALESCE(CONCAT(amp_rating, 'Amp'), ''), ' ',
                        COALESCE(CONCAT(min_rpm_rating, 'min rpm'), ''), ' ',
                        COALESCE(CONCAT(max_rpm_rating, 'max rpm'), '')
                ) AS other_desc
        FROM PowerAirCompressor pa
        JOIN PowerTool pt ON pt.id = pa.id
        UNION
        SELECT
                pt.id AS toolId,
                CONCAT(
                        COALESCE(adjustable_clutch, ''), ' ',
                        COALESCE(CONCAT(min_torque_rating, 'ft-lb'), ''), ' ',
                        COALESCE(CONCAT(max_torque_rating, 'ft-lb'), ''), ' ',
                        COALESCE(CONCAT(volt_rating, 'V'), ''), ' ',
                        COALESCE(CONCAT(amp_rating, 'Amp'), ''), ' ',
                        COALESCE(CONCAT(min_rpm_rating, 'min rpm'), ''), ' ',
                        COALESCE(CONCAT(max_rpm_rating, 'max rpm'), '')
                ) AS other_desc
        FROM PowerDrill pa
        JOIN PowerTool pt ON pt.id = pa.id
        UNION
        SELECT
                pt.id AS toolId,
CONCAT(
                        COALESCE(CONCAT(power_rating, 'Watts'), ''), ' ',
```

Table of Contents

```
                        COALESCE(CONCAT(volt_rating, 'V'), ''), ' ',
                        COALESCE(CONCAT(amp_rating, 'Amp'), ''), ' ',
                        COALESCE(CONCAT(min_rpm_rating, 'min rpm'), ''), ' ',
                        COALESCE(CONCAT(max_rpm_rating, 'max rpm'), '')
                ) AS other_desc
        FROM PowerGenerator pa
        JOIN PowerTool pt ON pt.id = pa.id
        UNION
        SELECT
                pt.id AS toolId,
                CONCAT(
                        COALESCE(CONCAT(motor_rating, 'HP'), ''), ' ',
                        COALESCE(CONCAT(drum_size, 'cu ft.'), ''), ' ',
                        COALESCE(CONCAT(volt_rating, 'V'), ''), ' ',
                        COALESCE(CONCAT(amp_rating, 'Amp'), ''), ' ',
                        COALESCE(CONCAT(min_rpm_rating, 'min rpm'), ''), ' ',
                        COALESCE(CONCAT(max_rpm_rating, 'max rpm'), '')
                ) AS other_desc
        FROM PowerMixer pa
        JOIN PowerTool pt ON pt.id = pa.id
        UNION
        SELECT
                pt.id AS toolId,
CONCAT(
                        COALESCE(dust_bag, ''), ' ',
                        COALESCE(CONCAT(volt_rating, 'V'), ''), ' ',
                        COALESCE(CONCAT(amp_rating, 'Amp'), ''), ' ',
                        COALESCE(CONCAT(min_rpm_rating, 'min rpm'), ''), ' ',
                        COALESCE(CONCAT(max_rpm_rating, 'max rpm'), '')
                ) AS other_desc
        FROM PowerSander pa
        JOIN PowerTool pt ON pt.id = pa.id
        UNION
        SELECT
                pt.id AS toolId,
                CONCAT(
                        COALESCE(CONCAT(blade_size, 'in'), ''), ' ',
                        COALESCE(CONCAT(volt_rating, 'V'), ''), ' ',
                        COALESCE(CONCAT(amp_rating, 'Amp'), ''), ' ',
                        COALESCE(CONCAT(min_rpm_rating, 'min rpm'), ''), ' ',
                        COALESCE(CONCAT(max_rpm_rating, 'max rpm'), '')
                ) AS other_desc
        FROM PowerSaw pa
        JOIN PowerTool pt ON pt.id = pa.id
        UNION
        SELECT
                ga.id AS toolId,
```

Table of Contents

```
                CONCAT(
                        COALESCE(handle_material, ''), ' ',
                        COALESCE(CONCAT(blade_length, 'in.'), ''), ' ',
                        COALESCE(blade_material, '')
                ) AS other_desc
        FROM PruningTool ga
        JOIN GardenTool gt ON gt.id = ga.id
        UNION
        SELECT
                ga.id AS toolId,
CONCAT(
                        COALESCE(handle_material, ''), ' ',
                        COALESCE(CONCAT(tine_count, 'tine'), ''), ' '
                ) AS other_desc
        FROM RakeTool ga
        JOIN GardenTool gt ON gt.id = ga.id
        UNION
        SELECT
                ga.id AS toolId,
                CONCAT(
                        COALESCE(CONCAT(head_weight, 'lb.'), ''), ' ',
                        COALESCE(handle_material, '')
                ) AS other_desc
        FROM StrikingTool ga
        JOIN GardenTool gt ON gt.id = ga.id
        UNION
        SELECT
                ga.id AS toolId,
                CONCAT(
                        COALESCE(bin_material, ''), ' ',
                        COALESCE(CONCAT(wheel_count, 'wheeled'), ''), ' ',
                        COALESCE(CONCAT(bin_volume, 'cu ft.'), ''), ' ',
                        COALESCE(handle_material, '')
                ) AS other_desc
        FROM WheelBarrowTool ga
        JOIN GardenTool gt ON gt.id = ga.id
        UNION
        SELECT
                ga.id AS toolId,
                CONCAT(
                        COALESCE(CONCAT(blade_length, 'in.'), ''), ' ',
                        COALESCE(CONCAT(blade_width, 'in.'), ''), ' ',
                        COALESCE(handle_material, '')
                ) AS other_desc
        FROM DiggingTool ga
        JOIN GardenTool gt ON gt.id = ga.id
        UNION
```

Table of Contents

```
                SELECT
                        la.id AS toolId,
                        CONCAT(
                                COALESCE(pail_shelf, ''), ' ',
                                COALESCE(step_count, ''), ' ',
                                COALESCE(CONCAT(weight_capacity, 'Lb.'), '')
                        ) AS other_desc
                FROM StepLadder la
                JOIN LadderTool lt ON lt.id = la.id
                UNION
                SELECT
                        la.id AS toolId,
        CONCAT(
                                COALESCE(rubber_feet, ''), ' ',
                                COALESCE(step_count, ''), ' ',
                                COALESCE(CONCAT(weight_capacity, 'Lb.'), '')
                        ) AS other_desc
                FROM StraightLadder la
                JOIN LadderTool lt ON lt.id = la.id
        ) other_desc
        ON other_desc.toolId = tool.id
) tools
WHERE
        (
                other_desc LIKE CONCAT('%', @PartialName,'%') OR
                full_desc LIKE CONCAT('%', @PartialName,'%') OR
                short_desc LIKE CONCAT('%', @PartialName,'%')
        )
        AND subtype = @subtype
        AND powersource = @powersource
   AND category = @category
```

Full Tool Details

Abstract Code

User clicked button that requires detailed description using the tool id.

- Summarized: Find Hand, Garden, Ladder, Power, etc. (including accessories, materials, etc) based on Tool.Number; Display full description
- Find the tool with tool number/id.
- Returns: sub-type, sub-option, power-source, width, length, weight, manufacturer.
- Also return: Short description [power-source][suboption][subtype] with the units of power-source whether electric, cordless, or gas. (manual excluded from units)
- Also return: Full description [dimensions][short description][other descriptors][manufacturer]

Table of Contents

- In addition to rental price, deposit price, and description of accessories.
- Every tool whether hand tool, garden tool, power tool, or ladder tool is described within the other descriptors with each tool defined units.

```
-- Full Tool Details
-- including all the details of full description and short description

SET @toolid :='8';
SELECT
        tool.id AS toolId,
   Category.name AS category,
        CONCAT(
                COALESCE(powersource.name, ''), ' ',
                COALESCE(suboption.name, ''), ' ',
                COALESCE(subtype.name, '')
        ) AS short_desc,
        CONCAT(
                COALESCE(width, ''), ' in. W x',
                COALESCE(length, ''), 'in. L ',
                CONCAT(COALESCE(weight, ''), 'lb'), ' ',
                COALESCE(other_desc.other_desc, ''), ' ',
                COALESCE(
                        IF(
                                powersource.name='A/C','electric',
                                IF(
                                        powersource.name='D/C', 'cordless',
                                        powersource.name
                                )
                        ),
        ''), ' ',
                COALESCE(suboption.name, ''), ' ',
                COALESCE(subtype.name, ''), ' ',
                CONCAT('By ', COALESCE(manufacturer, '')), ' '
        ) AS full_desc,
   Powersource.name AS powersource,
   Subtype.name AS subtype, suboption.name AS suboption,
   rental_price,
   deposit_price,
   material,
   width,
   weight,
   length,
   manufacturer,
   accessory.description AS acc_description
FROM Tool AS tool
JOIN SubOption AS suboption ON suboption.id = tool.SubOption_Id
```

Table of Contents

```
JOIN SubType AS subtype ON subtype.id = tool.SubType_Id
JOIN PowerSource AS powersource ON powersource.id = tool.PowerSource_Id
JOIN Category AS category ON category.id = tool.Category_Id
JOIN Accessory AS accessory ON accessory.PowerTool_Id = tool.id
JOIN (

	SELECT
		CONCAT(
			COALESCE(CONCAT(gauge_rating, 'G'), ''), ' ',
			COALESCE(capacity, '')
		) AS other_desc
	FROM HandGun
	WHERE id = @toolid
	UNION
	SELECT
		anti_vibration AS other_desc
	FROM HandHammer
	WHERE id = @toolid
	UNION
	SELECT
		adjustable AS other_desc
	FROM HandPlier
	WHERE id = @toolid
	UNION
	SELECT
		CONCAT(drive_size, 'in') AS other_desc
	FROM HandRatchet
	WHERE id = @toolid
	UNION
	SELECT
		CONCAT('#', screw_size)AS other_desc
	FROM ScrewDriver
	WHERE id = @toolid
	UNION
	SELECT
		CONCAT(
			COALESCE(CONCAT(drive_size, 'in'), ''), ' ',
			COALESCE(CONCAT(sae_size, 'in'), ''), ' ',
			COALESCE(deep_socket, '')
		) AS other_desc
	FROM HandSocket
	WHERE id = @toolid
	UNION
	SELECT
		CONCAT(
			COALESCE(CONCAT(tank_size, 'gal'), ''), ' ',
			COALESCE(CONCAT(pressure_rating, 'psi'), ''), ' ',
```

Table of Contents

```
                COALESCE(CONCAT(volt_rating, 'V'), ''), ' ',
                COALESCE(CONCAT(amp_rating, 'Amp'), ''), ' ',
                COALESCE(CONCAT(min_rpm_rating, 'min rpm'), ''), ' ',
                COALESCE(CONCAT(max_rpm_rating, 'max rpm'), '')
        ) AS other_desc
FROM PowerAirCompressor pa
JOIN PowerTool pt ON pt.id = pa.id
WHERE pt.id = @toolid
UNION
SELECT
        CONCAT(
                COALESCE(adjustable_clutch, ''), ' ',
                COALESCE(CONCAT(min_torque_rating, 'ft-lb'), ''), ' ',
                COALESCE(CONCAT(max_torque_rating, 'ft-lb'), ''), ' ',
                COALESCE(CONCAT(volt_rating, 'V'), ''), ' ',
                COALESCE(CONCAT(amp_rating, 'Amp'), ''), ' ',
                COALESCE(CONCAT(min_rpm_rating, 'min rpm'), ''), ' ',
                COALESCE(CONCAT(max_rpm_rating, 'max rpm'), '')
        ) AS other_desc
FROM PowerDrill pa
JOIN PowerTool pt ON pt.id = pa.id
WHERE pt.id = @toolid
UNION
SELECT
        CONCAT(
                COALESCE(CONCAT(power_rating, 'Watts'), ''), ' ',
                COALESCE(CONCAT(volt_rating, 'V'), ''), ' ',
                COALESCE(CONCAT(amp_rating, 'Amp'), ''), ' ',
                COALESCE(CONCAT(min_rpm_rating, 'min rpm'), ''), ' ',
                COALESCE(CONCAT(max_rpm_rating, 'max rpm'), '')
        ) AS other_desc
FROM PowerGenerator pa
JOIN PowerTool pt ON pt.id = pa.id
WHERE pt.id = @toolid
UNION
SELECT
        CONCAT(
                COALESCE(CONCAT(motor_rating, 'HP'), ''), ' ',
                COALESCE(CONCAT(drum_size, 'cu ft.'), ''), ' ',
                COALESCE(CONCAT(volt_rating, 'V'), ''), ' ',
                COALESCE(CONCAT(amp_rating, 'Amp'), ''), ' ',
                COALESCE(CONCAT(min_rpm_rating, 'min rpm'), ''), ' ',
                COALESCE(CONCAT(max_rpm_rating, 'max rpm'), '')
        ) AS other_desc
FROM PowerMixer pa
JOIN PowerTool pt ON pt.id = pa.id
WHERE pt.id = @toolid
```

Table of Contents

```
UNION
SELECT
	CONCAT(
		COALESCE(dust_bag, ''), ' ',
		COALESCE(CONCAT(volt_rating, 'V'), ''), ' ',
		COALESCE(CONCAT(amp_rating, 'Amp'), ''), ' ',
		COALESCE(CONCAT(min_rpm_rating, 'min rpm'), ''), ' ',
		COALESCE(CONCAT(max_rpm_rating, 'max rpm'), '')
	) AS other_desc
FROM PowerSander pa
JOIN PowerTool pt ON pt.id = pa.id
WHERE pt.id = @toolid

UNION
SELECT
	CONCAT(
		COALESCE(CONCAT(blade_size, 'in'), ''), ' ',
		COALESCE(CONCAT(volt_rating, 'V'), ''), ' ',
		COALESCE(CONCAT(amp_rating, 'Amp'), ''), ' ',
		COALESCE(CONCAT(min_rpm_rating, 'min rpm'), ''), ' ',
		COALESCE(CONCAT(max_rpm_rating, 'max rpm'), '')
	) AS other_desc
FROM PowerSaw pa
JOIN PowerTool pt ON pt.id = pa.id
WHERE pt.id = @toolid

UNION
SELECT
	CONCAT(
		COALESCE(handle_material, ''), ' ',
		COALESCE(CONCAT(blade_length, 'in.'), ''), ' ',
		COALESCE(blade_material, '')
	) AS other_desc
FROM PruningTool ga
JOIN GardenTool gt ON gt.id = ga.id
WHERE gt.id = @toolid

UNION
SELECT
	CONCAT(
		COALESCE(handle_material, ''), ' ',
		COALESCE(CONCAT(tine_count, 'tine'), '')
	) AS other_desc
FROM RakeTool ga
JOIN GardenTool gt ON gt.id = ga.id
WHERE gt.id = @toolid
```

Table of Contents

```
UNION
SELECT
        CONCAT(
                COALESCE(CONCAT(head_weight, 'lb.'), ''), ' ',
                COALESCE(handle_material, '')
        ) AS other_desc
FROM StrikingTool ga
JOIN GardenTool gt ON gt.id = ga.id
WHERE gt.id = @toolid

UNION
SELECT
        CONCAT(
                COALESCE(bin_material, ''), ' ',
                COALESCE(CONCAT(wheel_count, 'wheeled'), ''), ' ',
                COALESCE(CONCAT(bin_volume, 'cu ft.'), ''), ' ',
                COALESCE(handle_material, '')
        ) AS other_desc
FROM WheelBarrowTool ga
JOIN GardenTool gt ON gt.id = ga.id
WHERE gt.id = @toolid

UNION
SELECT
        CONCAT(
                COALESCE(CONCAT(blade_length, 'in.'), ''), ' ',
                COALESCE(CONCAT(blade_width, 'in.'), ''), ' ',
                COALESCE(handle_material, '')
        ) AS other_desc
FROM DiggingTool ga
JOIN GardenTool gt ON gt.id = ga.id
WHERE gt.id = @toolid

UNION
SELECT
        CONCAT(
                COALESCE(pail_shelf, ''), ' ',
                COALESCE(step_count, ''), ' ',
                COALESCE(CONCAT(weight_capacity, 'Lb.'), '')
        ) AS other_desc
FROM StepLadder la
JOIN LadderTool lt ON lt.id = la.id
WHERE la.id = @toolid

UNION
SELECT
        CONCAT(
```

Table of Contents

```
                        COALESCE(rubber_feet, ''), ' ',
                        COALESCE(step_count, ''), ' ',
                        COALESCE(CONCAT(weight_capacity, 'Lb.'), '')
                ) AS other_desc
        FROM StraightLadder la
        JOIN LadderTool lt ON lt.id = la.id
        WHERE la.id = @toolid
) other_desc
WHERE tool.id = @toolid;
```

Make Reservation

Abstract Code

- User($customer_username) clicks ***Make Reservation*** button from **Main Menu**
- User enters *Start Date*($start_date)*, End Date*($end_date)*, Keywords*($keywords)*, Power Source*($power_source)*, Sub-Type*($sub_type) and/or *Type*($type)
- Run **Tool Search** task where Reservation.ToolNumber equals ToolNumbers from search and Reservation.EndDate not greater than *Start Date*
    - For each Tool; Display Tool Number, Short Description, Rental Price, Deposit Price

```
SELECT Tool_Id FROM ToolReservations WHERE Tool_Id  NOT IN (SELECT TR.Tool_Id
FROM Reservation AS R INNER JOIN ToolReservations AS TR ON
TR.Reservations_Id=R.id WHERE DropOffClerk_UserName IS NULL or end_date >
@start_date)
UNION SELECT id FROM Tool WHERE id NOT IN (SELECT Tool_Id FROM
ToolReservations);
```

- User clicks ***Add*** button
    - If less than 11 tools already added
        - If tool count reaches 0
            - Display error message
        - Else
            - Add tool to reservation list section
    - Else
        - Display error message for user to reduce the number of tools in the current reservation to 10.
    - If the tool is being returned within next 24 hours
        - Display info message
- User clicks ***Remove*** *button*
    - Remove item from reservation list section

Table of Contents

- User clicks ***Calculate Total*** button
    - Run **Reservation Summary** task
        - Displays Reservation Dates, Number of Days
        - Calculate total deposits, and rental price; Display Total Deposit, Rental Price
        - User clicks ***Submit*** button
            - Write new Reservation record, return ReservationNumber($reservation_id)

```
INSERT INTO Reservation (booking_date, start_date, end_date, Customer_UserName)
VALUES (NOW(), @start_date, @end_date, @customer_username);
```

            - For each tool($tool_id) to be reserved, find ToolNumber
                - Check if tool is already reserved by checking for ToolNumber isin Reservation with an EndDate greater than the StartDate or the DropOffClerk_UserName is NULL.

```
SELECT R.id FROM Reservation AS R INNER JOIN ToolReservations AS TR ON
TR.Reservations_Id=R.id WHERE DropOffClerk_UserName IS NULL or end_date >
@start_date AND TR.tool_id=@tool_id;
```

                - If tool is not reserved,

```
INSERT INTO ToolReservations VALUES (@tool_id, @reservation_id);
```

                    - Update Reservation Summary to be Reservation Confirmation
                        - Display ReservationNumber($reservation_id)
                - Else
                    - Redirect User to **Make Reservation** form
                    - Remove item from Tools Added to Reservation list
                    - Update Available Tools For Rent list
        - User clicks ***Reset*** *button*
            - Reset values in **Make Reservation** form to initial state value
- Go to **Main Menu** form

Purchase Tool

Abstract Code

- User($customer_username) clicks ***Purchase Tool*** button from **Main Menu**
- User enters *Keyword*($keywords), *Type*($type), *Sub-Type*($sub_type), and/or *Power Source*($power_source)

Table of Contents

- User clicks ***Search*** *button*
    - Run **Tool Search** task**:** find tools w/ no SoldDate in SaleOrder

```
(SELECT Tool_Id FROM SaleOrder WHERE sold_date IS NULL);
```

    - User clicks ***Purchase Tool*** *button*
        - Add tool to purchase list
    - User clicks ***Submit*** button
        - Find Credit Card using Customer.Username($customer_username)

```
SELECT card_number, expiration_month, expiration_year, cvc FROM CreditCard AS CC
INNER JOIN Customer AS C ON CC.id=C.CreditCard_Id WHERE
C.user_name=@customer_username;
```

        - Process CreditCard.Number for Customer.Username
    - Run **Purchase tool task**
        - For each tool($tool_id) in Tools
            - Update SoldDate -> now() and Customer.Number for SaleOrder

```
UPDATE SaleOrder SET sold_date=NOW(), Customer_UserName=@customer_username
WHERE tool_id=@tool_id;
```

        - Run **Purchase Confirmation task**
            - Display ConfirmationNumber
- Go to **Main Menu** form

Pick-Up Reservation

Abstract Code

- User($clerk_username) clicks ***Pick-Up button*** from **Main Menu**
- Run **Pick-Up Reservation** task
    - For each Reservation WHERE Reservation.DropoffOffClerk_UserName is NULL
        - Display ReservationNumber, CustomerNumber, CustomerName, StartDate, and EndDate

```
SELECT R.id AS ReservationNumber, C.user_name AS CustomerUsername, c.id AS
customerId, concat(C.first_name, " ", C.last_name) AS CustomerName, R.start_date,
R.end_date FROM Customer AS C INNER JOIN Reservation AS R ON
```

Table of Contents

```
C.user_name=R.Customer_UserName WHERE R.DropOffClerk_UserName IS NULL;
```

- User clicks ***ID link*** or enters ID and clicks ***Pick-Up*** button
- Run **Pick-Up Reservation Summary task**
    - Display TotalDeposit, CustomerName, TotalRentalRrice
    - Find CreditCard with Reservation.CustomerNumber

```
SELECT card_number, expiration_month, expiration_year, cvc FROM CreditCard AS CC
INNER JOIN Customer AS C ON CC.id=C.CreditCard_Id INNER JOIN Reservation AS R ON
R.Customer_UserName=C.user_name WHERE R.id=@reservation_id;
```

    - If User clicks ***New*** button
        - Run **Credit Card Information** task
            - User enters *credit card name*($cardname)*, credit card number*($card_number)*, CVC*($cvc)*, expiration month*($expiration_month), *and expiration year*($expiration_year)
            - If user clicks ***Confirm Pick Up*** button
                - Insert CreditCardName, CreditCardNumber, CVC, ExpirationMonth, ExpirationDate in CreditCard
                - Insert Customer.Number, CreditCard.ID($credit_card_id) in CustomerCreditCard

```
INSERT INTO CreditCard (name, card_number, cvc, expiration_month, expiration_year)
VALUES (@cardname, @card_number, @cvc, @expiration_month, @expiration_year);

UPDATE Customer SET CreditCard_Id=@credit_card_id WHERE
user_name=@customer_username;
```

- User clicks ***Confirm Pick Up*** button
- Run **Rental Contract task**
    - Update Reservation with Pick-Up clerk's Clerk.Number, BookingDate -> now()

```
UPDATE Reservation SET PickupClerk_UserName=@clerk_username,
booking_date=NOW() WHERE id=@reservation_id;
```

    - Display Pick-UpClerkName, CustomerName, CreditCard XXXX, Start Date, End Date
    - For each Tool.Number in Reservation
        - Update Rentals

Table of Contents

```
INSERT INTO Rentals (Tool_Id, start_date, number_of_rentals) VALUE (@tool_id, NOW(), 1)
ON DUPLICATE KEY UPDATE start_date=NOW(), end_date=NULL,
number_of_rentals=number_of_rentals + 1;
```

        - Find Tool.Name, DepositPrice, RentalPrice

```
SELECT deposit_price, rental_price FROM Tool WHERE id=@tool_id;
```

    - User clicks ***Print Contract*** button
        - Go to **Main Menu** form

Drop-Off Reservation

Abstract Code

- User($clerk_username) clicks ***Drop-Off button*** from **Main Menu**
- Run **Drop-Off Reservation** task
    - For each Reservation with Reservation.DropOffClerk is NULL
        - Find the Customer.Name from Reservation.CustomerNumber
        - Display, ReservationNumber, CustomerNumber, CustomerName, StartDate, and EndDate

```
SELECT R.id AS ReservationNumber, C.user_name AS CustomerUsername,
concat(C.first_name, " ", C.last_name) AS CustomerName, R.start_date, R.end_date FROM
Customer AS C INNER JOIN Reservation AS R ON C.user_name=R.Customer_UserName
WHERE R.DropOffClerk_UserName IS NULL;
```

- User clicks ***ID link*** or enters ID and clicks ***Drop-Off*** *button*
- Run **Drop-Off Reservation Summary** task
    - Find the Customer.Name from Reservation.CustomerNumber
    - Display TotalDeposit, CustomerName, TotalRentalRrice
    - Calculate TotalDue; Display TotalDue
- Run **Rental Contract task**
    - Update Reservation with Drop-Off clerk's Clerk.Number

```
UPDATE Reservation SET DropOffClerk_UserName=@clerk_username WHERE
id=@reservation_id;
```

    - Display Drop-OffClerk.Name, CustomerName, CreditCard XXXX, Start Date, End Date
    - For each Tool.Number in Reservation

Table of Contents

- Update Rentals

```
UPDATE Rentals SET end_date=NOW() WHERE Tool_Id=@tool_id;
```

- Check number of rentals, if equals 50, create Sales Order.

```
SELECT number_of_rentals FROM Rentals WHERE number_of_rentals >= 50;

# If greater than 50
INSERT INTO SaleOrder (Tool_Id, for_sale_date, purchase_price, Clerk_UserName)
VALUES (@tool_id, NOW(), (SELECT original_price FROM Tool WHERE id=@tool_id) * 0.5,
'jwatson@tools4rent.com');
```

- Find Tool.Name, DepositPrice, RentalPrice

```
SELECT deposit_price, rental_price FROM Tool WHERE id=@tool_id;
```

- User clicks ***Tool Name*** link
  - Run **Full Tool Details** task
- User can clicks ***Print Contract*** button or ***Close*** button
  - Go to **Main Menu** form

Add New Tool

Abstract Code

- User clicks ***Add Tool*** from **Main Menu**
- User selects *Type*
- Run **Sub-Type**($Sub_Type)**/Sub-Option**($Sub_Option) task

  - Find Sub-Type based on Type; Populate Sub-Type menu

```
SET @catId = 1;
SET @powersource = 1;
SET @subtypeId = 2;

SELECT name FROM Category;

SELECT name FROM PowerSourceCategory AS psc
JOIN PowerSource AS ps ON ps.id = psc.PowerSource_Id
WHERE psc.Category_Id = @catId;

SELECT name
```

Table of Contents

```
FROM SubTypePowerSource AS stps
JOIN SubType AS st ON st.id = stps.SubType_Id
WHERE stps.Category_Id = @catId AND stps.PowerSource_Id =
@powersource;

SELECT so.name FROM SubOption AS so
JOIN SubType AS st ON st.id = so.SubType_Id
JOIN SubTypePowerSource AS stps ON stps.SubType_Id = st.id
WHERE PowerSource_Id = @powersource AND so.SubType_Id =
@subtypeId AND Category_Id = @catId;
```

    - Validate that tool type and power-source are selected first before allowing sub-type and sub-options
    - Find Sub-Option based on Type, Sub-Type; Populate Sub-Option menu
- User enters *Purchase Price(*$Purchase_Price*)*

Calculate the *Deposit Price and Rental Prices*

```
SET @Purchase_Price = 20,
@Tool_id = 2;
UPDATE Tool
SET deposit_price = @Purchase_Price * 0.4, rental_price = @Purchase_Price
* 0.5
WHERE Tool.id = @Tool_id;
```

- User select ***Hand Tool*** radio button
    - Update menu options
    - User enters *Manufacturer, Width, Width Fraction, Width Unit, Length, Weight, Length Fraction, Length Unit, Drive/Chuck Size (if applicable)*
- User select ***Power Tool*** radio button
    - Find Power Source based on Type, Sub-Type; Populate Power Source menu
    - Update menu options
    - User enters *Power :Source, Gauge Unit, Capacity Unit, A/C Volt Rating, Power Generated, Power Fraction, Power Unit, Torque Min/Max, Pressure Min/Max, Speed Min/Max*
    - User enters *Power Tool Accessory Quantity, Description*
    - User clicks ***Add Accessory*** button
        - Add another accessory input fields
    - Validate User enters multiple speed for variable speed devices and one speed for single speed devices
- User select ***Cordless Tool*** option
    - Update menu options
    - User enters *Battery Type, Quantity, D/C Volt Rating*
- Convert measurements
- Validate all fields have values based on type

Table of Contents

- User clicks ***Add Tool*** button
    - Insert Tool price, weight, etc. into Tool

```
# This insert tool is used AS the base for all tool insertions then we have another insert for each specific Tool Category.

INSERT INTO Tool (width, weight, length, manufacturer, material, deposit_price, rental_price, original_price, Category_Id, PowerSource_Id, SubType_Id, SubOption_Id)
VALUES (@width, @weight, @length, @manufacturer, @material, @deposit_price, @original_price, @rental_price, @Category_Id, @PowerSource_Id, @SubType_Id, @SubOption_Id);

SET @lastTool:= last_insert_id();
```

-

```
# We also need a query for each specific subtype

#ScrewDriver
INSERT INTO HandTool VALUES (@lastTool);
SET @lastTool:= last_INSERT_id();
INSERT INTO ScrewDriver (id, screw_size) VALUES (@lastTool, @screwsize);

# Socket
SET @lastTool:= last_insert_id();
INSERT INTO HandTool VALUES (@lastTool);
SET @lastTool:= last_insert_id();
INSERT INTO HandSocket (id, drive_size, sae_size, deep_socket) VALUES (@lastTool, @drivesize,@saeSize, @deepsocket);

#Ratchet
INSERT INTO HandTool VALUES (@lastTool);
SET @lastTool:= last_insert_id();
INSERT INTO HandRatchet (id, drive_size) VALUES (@lastTool,@drivesize);

#Plier
INSERT INTO HandTool VALUES (@lastTool);
SET @lastTool:= last_insert_id();
INSERT INTO HandPlier(id, adjustable) VALUES (@lastTool,@adjustable);

#Gun
INSERT INTO HandTool VALUES (@lastTool);
SET @lastTool:= last_insert_id();
INSERT INTO HandGun (id, gauge_rating, capacity) VALUES (@lastTool,@gaugerating, @capacity);

#Hammer
```

Table of Contents

```
INSERT INTO HandTool VALUES (@lastTool);
SET @lastTool:= last_insert_id();
INSERT INTO HandHammer (id, anti_vibration) VALUES (@lastTool,@antivib);

# Garden, use this AS a base for each Garden tool below
INSERT INTO GardenTool (id, handle_material) VALUES (@lastTool, @handlematerial);

# Pruning
INSERT INTO PruningTool (id, blade_material, blade_length) VALUES (@lastTool,
@blademateriał, @bladelength);

#Striking
INSERT INTO StrikingTool (id, head_weight) VALUES (@lastTool, @headweight);

#Digging
INSERT INTO DiggingTool (id, blade_width, blade_length) VALUES (@lastTool,
@bladewidth, @bladelength);

#Rake
INSERT INTO RakeTool (id, tine_count) VALUES (@lastTool, @tinecount);

#Wheelbarrow
INSERT INTO WheelBarrowTool (id, bin_material,wheel_count,bin_volume) VALUES
(@lastTool, @binmaterial, @wheelcount, @binvolume);

#PowerTool - base for each power tool below
INSERT INTO PowerTool (id, volt_rating, amp_rating, min_rpm_rating, max_rpm_rating)
VALUES (@lastTool, @voltrating, @amprating, @minrpm, @maxrpm);

#PowerDrill
INSERT INTO PowerDrill (id, adjustable_clutch, min_torque_rating, max_torque_rating)
VALUES (@lastTool, @adjustableclutch, @mintorquerating, @maxtorquerating);

#Saw
INSERT INTO PowerSaw (id, blade_size) VALUES (@lastTool, @bladesize);

#Sander
INSERT INTO PowerSander (id, dust_bag) VALUES (@lastTool, @dustbag);

#AirCompressor
INSERT INTO PowerAirCompressor (id, tank_size, pressure_Rating) VALUES (@lastTool,
@tanksize, @pressureRating);

#PowerMixer
INSERT INTO PowerMixer (id, motor_rating, drum_size) VALUES (@lastTool, @motorrating,
@drumsize);
```

Table of Contents

```
#PowerGenerator
INSERT INTO PowerGenerator (id, power_rating) VALUES (@lastTool, @powerrating);

#Accessory
INSERT INTO Accessory (description, quantity, PowerTool_Id) VALUES (@description,
@qty, @lastTool);
Set @lastAccessory = last_insert_id();

#Cordless Accessory
INSERT INTO CordlessAccessory (id, volt_rating, amp_rating) VALUES (@lastAccessory,
@voltrating, @amprating);

#Ladder - Base use for each ladder subtype below
SET @lastTool:= last_insert_id();
INSERT INTO LadderTool (id, step_count, weight_capacity) VALUES (@lastTool,
@stepcount, @weightcapacity);

#Straight Ladder
INSERT INTO StrightLadder (id, rubber_feet) VALUES (@lastTool, @rubberfeet);

#Step Ladder
INSERT INTO StepLadder (id, pail_shelf) VALUES (@lastTool, @pailshelf);
```

- Go to **Main Menu** form

Service Order / Repair Tool

Abstract Code

- User clicks ***Repair Tool*** from **Main Menu**
- User enters *StartDate, EndDate, Keyword, Type, Power Source, Sub-Type*

```
SET @StartDate = '2017-10-02', @EndDate = '2017-10-09', @Service_Cost = 47.88,
@Category_Id = (SELECT id FROM Category WHERE name = 'Hand'),
@Tool_Id = @toolid,
@Clerk_ID = (SELECT id FROM Clerk WHERE user_name = 'admin@gatech.edu');
@PowerSource_Id, @SubType_Id, @SubOption_Id, @deposit_price, @rental_price);
```

- Run **Tool Search** task: where tools not in ServiceOrder.ToolNumber <> Tool.Number or ServiceOrder.ToolNumber == Tool.Number and EndDate <> Null
    - For each Tool
        - Display ToolNumber, Description, Rental Price, and Deposit Price

```
SET @tool_id = 1;
```

Table of Contents

```
SELECT t.id, concat(c.name, " ", ps.name, " ", st.name) AS Description, t.rental_price,
t.deposit_price
FROM Tool AS t
LEFT JOIN Category AS c ON c.id = t.Category_Id
LEFT JOIN PowerSourceCategory AS psc ON psc.Category_Id = t.Category_Id
LEFT JOIN PowerSource AS ps ON ps.id = psc.PowerSource_Id
LEFT JOIN SubType AS st ON st.id = t.SubType_Id
WHERE t.id = @tool_id
```

- User clicks ***Service Tool*** button
    - Update list of tools to service
- User enters *Tool ID*
    - Validate that tool does not have a service order
- User enters *Service Cost*
- User clicks ***Confirm*** button
    - Run **Service Tool** task
        - Create ServiceOrder record with ClerkNumber, StartDate, Cost

```
INSERT INTO ServiceOrder (start_date, end_date, clerk_username, service_cost, toolid)
VALUES (@StartDate, @EndDate, @Clerk_ID, @Service_Cost, @Tool_Id);
```

- Go to **Main Menu** form

Service Status

Abstract Code

- User clicks ***Service Status*** button from **Main Menu**
- User enters *StartDate, EndDate, Keyword, Type, Power Source, Sub-Type*
- Run **Tool Search** task: where ServiceOrder.EndDate is NULL
    - Display ServiceNumber, Status, ToolNumber, Description, StartDate, EndDate
    -

```
SELECT so.id AS ServiceOrder_ID, EXISTS(SELECT id FROM Tool t2 WHERE t.id = t2.id
and t2.id not in (SELECT Tool_Id FROM ToolReservations AS tr WHERE tr.Tool_Id = t2.id)
and t2.id not in (SELECT id FROM ServiceOrder AS so WHERE so.Tool_Id = t2.id AND
so.end_date > NOW() )
and t2.id not in (SELECT id FROM SaleOrder AS so WHERE so.sold_date is NULL AND
so.Tool_Id = t2.id)
and t2.id not in (SELECT id FROM SaleOrder AS so WHERE so.sold_date is not NULL AND
so.Tool_Id = t2.id)
) AS available,
EXISTS(SELECT Tool_Id FROM Rentals WHERE start_date <= NOW() AND end_date >=
NOW()) AS rented,
EXISTS(SELECT id FROM ServiceOrder AS so WHERE so.Tool_Id = t.id AND so.end_date
```

Table of Contents

```
> NOW() ) AS inrepair,
EXISTS(SELECT id FROM SaleOrder AS so WHERE so.sold_date is NULL AND so.Tool_Id
= t.id) AS forsale,
EXISTS(SELECT id FROM SaleOrder AS so WHERE so.sold_date is not NULL AND
so.Tool_Id = t.id) AS sold,
, t.id Tool_ID, concat(c.name, " ", ps.name, " ", st.name) AS Description, so.start_date AS
Service_StartDate, so.end_date AS Service_EndDate, @RepairCost, so.Clerk_Username AS
Clerk
FROM ServiceOrder AS so
LEFT JOIN Tool AS t ON t.id = so.Tool_Id
LEFT JOIN Category AS c ON c.id = t.Category_Id
LEFT JOIN PowerSourceCategory AS psc ON psc.Category_Id = t.Category_Id
LEFT JOIN PowerSource AS ps ON ps.id = psc.PowerSource_Id
LEFT JOIN SubType AS st ON st.id = t.SubType_Id
WHERE t.id = @tool_id;
```

    - RepairCost, Clerk.Name
- User clicks ***Fix Now? Button***
    - Update EndDate to now(), ClerkNumber to current Clerk

```
SET @toolid = 2;
UPDATE ServiceOrder
SET end_date = NOW()
WHERE Tool_id = @toolid;
```

- Users clicks ***Close*** button, go to **Main Menu** form

Sell Tool

Abstract Code

- User($clerk_username) clicks on ***Sell Tool*** button from **Main Menu**
- User enters *StartDate*($start_date), *EndDate*($end_date), *Keyword*($keyword), *Type*($type), *Power Source*($power_source), *Sub-Type*($sub_type)
- Run **Tool Search** task: In Rentals where number of rentals is greater than or equal to 50

```
SELECT Tool_Id FROM Rentals WHERE number_of_rentals >= 50;
```

    - For each Tool
        - Display ToolNumber, Description, RentalPrice, DepositPrice
- User clicks ***Sell Tool***
    - Create SaleOrder, with ToolNumber($tool_id), ForSaleDate

```
INSERT INTO SaleOrder (for_sale_date, purchase_price, Customer_UserName, Tool_Id,
```

Table of Contents

```
Clerk_UserName) VALUES (NOW(), @purchase_price, @customer_username, @tool_id,
@clerk_username);
```

- Users clicks ***Close*** button, go to **Main Menu** form

Sale Status

Abstract Code

- User clicks ***Sale Status*** button from **Main Menu**
- User enters *Keyword*($keyword), *Type*($type)
- Run **Tool Search** task: where ServiceOrder exists

```
SELECT Tool_Id FROM SaleOrder;
```

    - Display ServiceNumber, Status (ServiceOrder.EndDate <> NULL -> Sold), ToolNumber, Description, StartDate, EndDate RepairCost, Clerk.ID
    - If “Sold” (ServiceOrder.EndDate <> NULL -> Sold)
        - Display SalePrice, SaleDate

```
SELECT id, Tool_Id, Customer_UserName, purchase_price, for_sale_date, sold_date,
CL.employee_number FROM SaleOrder AS S INNER JOIN Clerk AS CL ON
S.Clerk_UserName=CL.user_name WHERE Tool_Id=@tool_id;
```

- Users clicks ***Close*** button, go to **Main Menu** form

Generate Report

Abstract Code

- User clicks ***Reports*** button from **Main Menu**
- Display Clerk Report link, Customer Report link, and Tool Inventory link
- User clicks ***Clerk Report*** link
    - Go to **Clerk Report**
    - For each Clerk
        - Display Number, First Name, Middle Name, Last Name, Email, Hiring Date
        - Sum number of pickups and dropoffs with Clerk.Number
        - Calculate Total; Display Number of Pickups, Dropoffs, Combined Total
    -

```
SELECT first_name, middle_name, last_name, email, date_of_hire, employee_number,
(SELECT count(PickupClerk_UserName) FROM Reservation AS r WHERE
```

Table of Contents

```
r.PickupClerk_UserName = c.user_name AND MONTH(r.booking_date) = MONTH(NOW())
AND YEAR(r.booking_date) = YEAR(NOW())) AS numPickups,
(SELECT count(DropOffClerk_UserName) FROM Reservation AS q WHERE
q.DropOffClerk_UserName = c.user_name AND MONTH(q.booking_date) = MONTH(NOW())
AND YEAR(q.booking_date) = YEAR(NOW())) AS numDropOffs,

# For some reason this syntax isn't working AND it should (numPickups + numDropOffs) AS
#CombinedTotal
# So instead use ugly syntax
((SELECT count(PickupClerk_UserName) FROM Reservation AS r WHERE
r.PickupClerk_UserName = c.user_name AND MONTH(r.booking_date) = MONTH(NOW())
AND YEAR(r.booking_date) = YEAR(NOW()))
+
(SELECT count(DropOffClerk_UserName) FROM Reservation AS q WHERE
q.DropOffClerk_UserName = c.user_name AND MONTH(q.booking_date) = MONTH(NOW())
AND YEAR(q.booking_date) = YEAR(NOW()))
) AS CombinedTotal
FROM Clerk AS c
ORDER BY CombinedTotal DESC;
```

- User clicks ***Customer Report*** link
    - Go to **Customer Report**
        - For each Customer
            - Display Customer Number, First Name, Middle Name, Last Name, Email, Phone
            - Count Reservation with Customer.Number; Display Total Reservations
            - Count Reservation.ToolNumber with Customer.Number; Display Total Tools Rented

```
SELECT id, first_name, middle_name, last_name, email,
(SELECT concat('(', area_code, ') ', `number`, case when extension is null then '' else concat('
x',extension) end) FROM PhoneNumber AS p WHERE p.Customer_UserName =
c.user_name AND p.primary = true) AS phone,
(SELECT count(Customer_UserName) FROM Reservation AS r WHERE
r.Customer_UserName = c.user_name AND MONTH(r.booking_date) = MONTH(NOW())
AND YEAR(r.booking_date) = YEAR(NOW())) AS totalReservations,
(SELECT count(Tool_id) FROM ToolReservations AS tr WHERE tr.Reservations_Id in
(SELECT id FROM Reservation AS r WHERE r.Customer_UserName = c.user_name AND
MONTH(r.booking_date) = MONTH(NOW()) AND YEAR(r.booking_date) = YEAR(NOW())))
AS ToolsRented
FROM Customer AS c
ORDER BY ToolsRented, last_name;
```

Table of Contents

    - User clicks ***View Profile***, then go to **View Profile** page
- User clicks ***Tool Inventory*** link
    - Go to **Tool Inventory Report**
    - User enters *All Tools, Hand Tool, Garden Tool, Ladder, Power Tool, Keyword i*
    - User clicks ***Search button***
    - Run **Tool Search** task
        - For each Tool
            - Display ToolNumber, Description,
            - Find CurrentStatus, Date; Display CurrentStatus Date
            - Sum all rental prices collected and subtract cost
            - Sum all cost, original cost, repairs;
            - Calculate Total Profit; Display Rental Profit, Total Cost, Total Profit
            -

```
SELECT t.id AS toolId,
concat_ws(' ',so.name, st.name) AS description,
EXISTS(SELECT id FROM Tool t2 WHERE t.id = t2.id
and t2.id not in (SELECT Tool_Id FROM ToolReservations AS tr WHERE tr.Tool_Id = t2.id)
and t2.id not in (SELECT id FROM ServiceOrder AS so WHERE so.Tool_Id = t2.id AND
so.end_date > NOW() )
and t2.id not in (SELECT id FROM SaleOrder AS so WHERE so.sold_date is NULL AND
so.Tool_Id = t2.id)
and t2.id not in (SELECT id FROM SaleOrder AS so WHERE so.sold_date is not NULL AND
so.Tool_Id = t2.id)
) AS available,
EXISTS(SELECT Tool_Id FROM Rentals WHERE start_date <= NOW() AND end_date >=
NOW() AND Tool_Id = t.id) AS rented,
EXISTS(SELECT id FROM ServiceOrder AS so WHERE so.Tool_Id = t.id AND so.end_date
> NOW() ) AS inrepair,
EXISTS(SELECT id FROM SaleOrder AS so WHERE so.sold_date is NULL AND so.Tool_Id
= t.id) AS forsale,
EXISTS(SELECT id FROM SaleOrder AS so WHERE so.sold_date is not NULL AND
so.Tool_Id = t.id) AS sold,
(SELECT sold_date FROM SaleOrder AS so WHERE so.sold_date is not NULL AND
so.Tool_Id = t.id) AS soldDate,
(SELECT sold_date FROM SaleOrder AS so WHERE so.sold_date is NULL AND so.Tool_Id
= t.id) AS forsaleDate,
(SELECT start_date FROM ServiceOrder AS so WHERE so.Tool_Id = t.id AND so.end_date
> NOW() ) AS inrepairDate,
(SELECT booking_date FROM ToolReservations AS tr JOIN Reservation AS r ON r.id
=tr.Reservations_Id WHERE tr.Tool_Id = t.id) AS rented_date,
(IFNULL((SELECT sum(DATEDIFF(end_date, start_date)) FROM Reservation AS r JOIN
ToolReservations AS tr ON tr.Reservations_Id = r.id WHERE t.id = tr.Tool_Id),0) *
rental_price) AS RentalProfit,
(t.original_price + IFNULL((SELECT sum(service_cost) FROM ServiceOrder AS so WHERE
so.Tool_Id = t.id),0)) AS TotalCost,
```

Table of Contents

```
((IFNULL((SELECT sum(DATEDIFF(end_date, start_date)) FROM Reservation AS r JOIN
ToolReservations AS tr ON tr.Reservations_Id = r.id WHERE t.id = tr.Tool_Id),0) *
rental_price)
-
(t.original_price + IFNULL((SELECT sum(service_cost) FROM ServiceOrder AS so WHERE
so.Tool_Id = t.id),0))) AS TotalProfit
FROM Tool AS t
JOIN SubOption AS so ON so.id = t.SubOption_Id
JOIN SubType AS st ON st.id = t.SubType_Id
WHERE t.Category_Id = @categoryId
ORDER BY TotalProfit DESC;
```

- Users clicks ***Close*** button, go to **Main Menu** form

Table of Contents