Phase 1 Report | CS6400 - Fall 2017 | Team 008

Name	GT ID
Xiaohua Cao	Xcao67
Xiaochen Wang	xwang875
Sihao Wang	swang632
Tiantian Zhang	tzhang386

Table of Content

Login

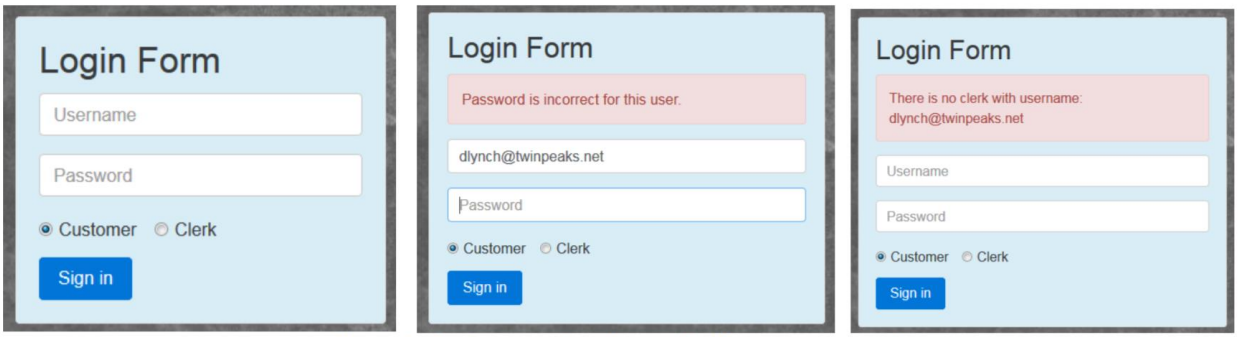


Figure 1 - Login screen for all users (with error messages)

User

username	email	password	first_name	middle_name	last_name

Abstract Code

- User enters *username*($username), and *password*($password) into input fields
- User selects ***Customer*** or ***Clerk*** radio button
- When data validation is successful for both username and password input fields, then:
    - When *Sign in* button is clicked:

```
SELECT password FROM 'User' WHERE user = '$username';
```

- If User record is found but User.password != '$password':
  - Go back to **Login** form, with error message.
- Else:
  - Store login information as session variable '$customer_id' or '$clerk_id'.
  - Go to **Customer Main Menu** or **Clerk Main Menu** form.

- Else if User.username != '$username':
  - If it is the ***Customer***:
    - Go to the **Customer Registration form** to create a new profile
  - Else
    - Go back to **Login** form, with error message

Registration

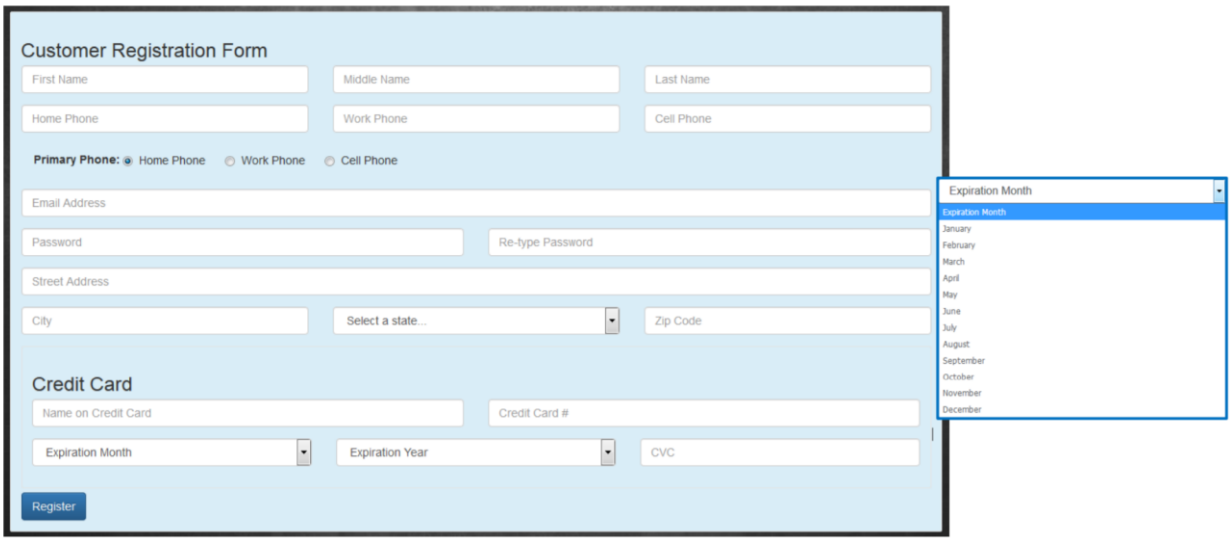


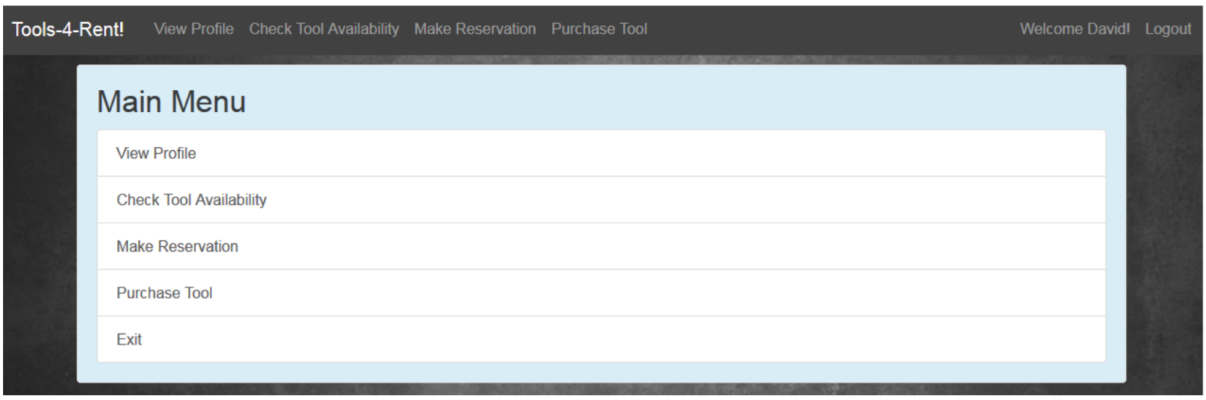
Figure 3 – Register (Customers only)

Abstract Code

- User enters *first_name*($first_name), *middle_name*($middle_name), *last_name*($last_name), *home_phone*($home_phone), *work_phone*($work_phone), *cell_phone*($cell_phone) into input fields.
- User selects either ***Home Phone, Work Phone*** or ***Cell Phone*** radio button as *Primary Phone*.
- User enters *email* ($email), *password*($password), *address*($address), into input fields.
- User inputs *name_on_card* ($name_on_card), *credit_card_number*($credit_card_number), *expiration_date*($experation_date) and cvc_number($cvc_number) into input fields.

```
INSERT INTO Customer (credit_card_number, name_on_card, expiration_date,
cvc_number, address)
VALUES ('$credit_card_number','$name_on_card', '$expiration_date',
'$cvc_number', '$address');
INSERT INTO User (email, password, first_name, middle_name, last_name)
VALUES ('$email', '$password', '$first_name', '$middle_name', '$last_name');
INSERT INTO Phone (phone_type, area_code, extension)
VALUES ('$phone_type', '$area_code', '$extension');
```

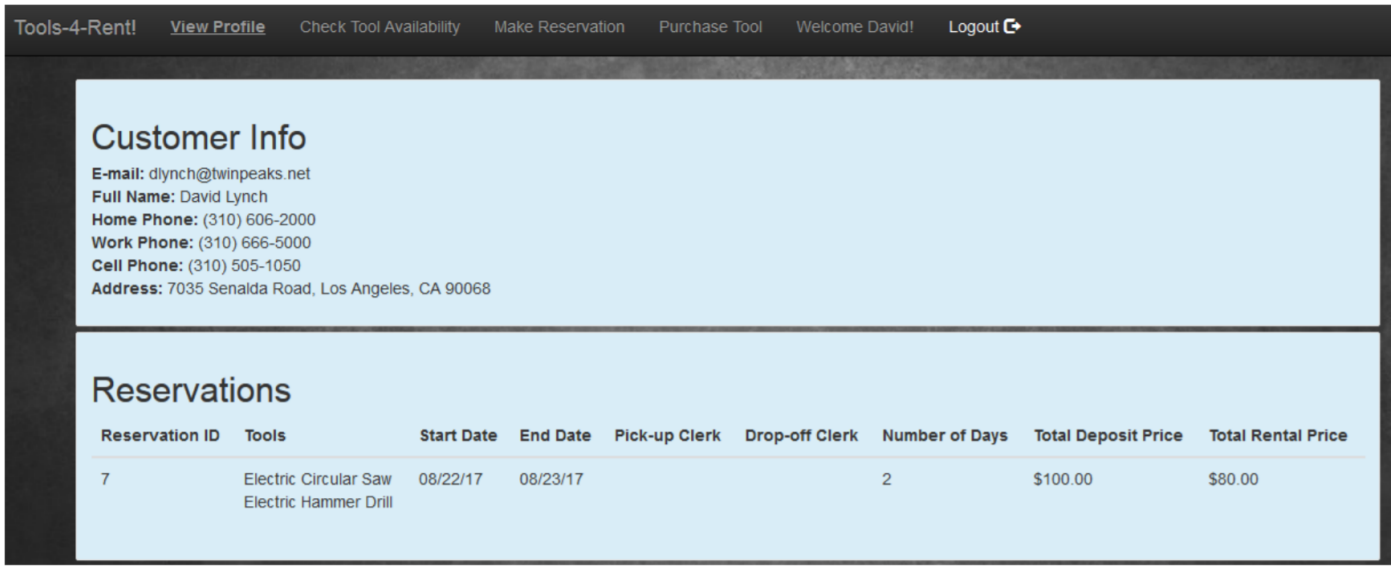
Customer Main Menu



Abstract Code

- Show "***View Profile***", "***Check Tool Availability***", "***Make Reservation***", "***Purchase Tool***", and "***Exit***" tabs.
- Upon:
  - 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 ***Exit*** button – Invalidate login session and go back to the **Login** form.

View Profile



Abstract Code

- User clicked on ***View Profile*** button from **Main Menu**:
- Run the **View Profile** task: query for information about the customer and their profile where #CustomerID is the ID of the current customer using the system from the HTTP Session/Cookie.
  - Find the current Customer using the Customer.username; Display Customer Email, Full Name, Home Phone, Work Phone, Cell Phone and Address.

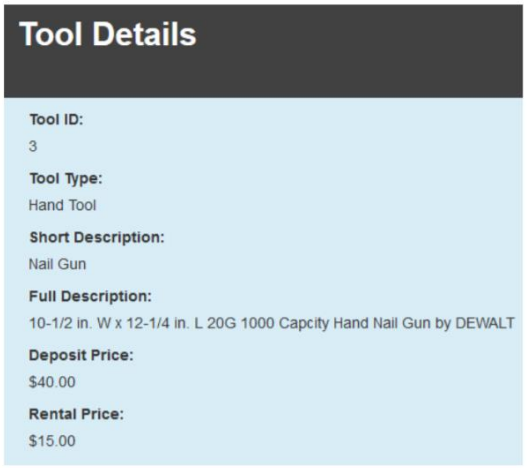
```
CREATE TEMPORARY TABLE CustomerPhone
SELECT CONCAT(Phone.phone_type, ": ", Phone.area_code, Phone.phone_number,
Phone.extension) AS "phone"
FROM Customer INNER JOIN Phone ON Phone.username = Customer.username;

SELECT email AS "E-mail", CONCAT (first_name, " ", middle_name, " ", last_name)
AS "Full Name", phone, address AS "Address"
FROM Customer NATURAL JOIN `User` NATURAL JOIN CustomerPhone
WHERE Customer.username = '$Username';
```

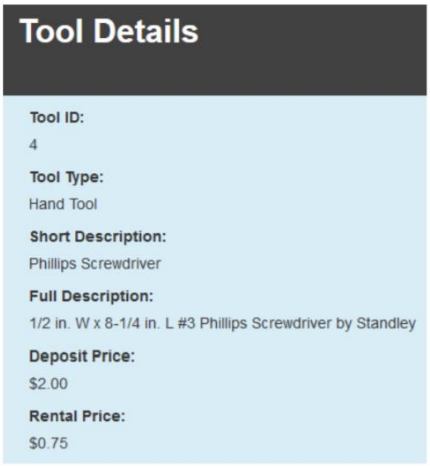
- Find reservations for the Customer:

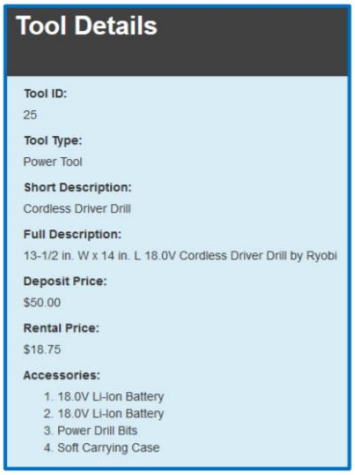
  Display the rental history lists the summaries for all reservations made, including Reservation ID, Tools, Start Date, End Date, names of pickup clerk, etc.

```
CREATE TEMPORARY TABLE PickUpClerk
SELECT clerk_id AS "Pick_UpClerk", reservation_id
FROM Clerk INNER JOIN PickUp ON Clerk.username = PickUp.clerk_username;

CREATE TEMPORARY TABLE DropOffClerk
SELECT clerk_id AS "Drop_OffClerk", reservation_id
FROM Clerk INNER JOIN DropOff ON Clerk.username = DropOff.clerk_username;

CREATE TEMPORARY TABLE Reservation_with_Clerk_Info
SELECT * FROM (Reservation NATURAL JOIN PickUpClerk) NATURAL JOIN
DropOffClerk;

CREATE TEMPORARY TABLE Reservation_Order_Tools
SELECT * FROM (Reservation_with_Clerk_Info NATURAL JOIN Tools)
NATURAL JOIN ToolInfo;

SELECT reservation_id AS "Reservation ID", CONCAT (IF(power_source =
'Manual' , ' ', CONCAT(power_source, ' ')), tool_suboption, ' ', tool_subtype) AS
"Tools", reservation_start_date AS "Start Date", reservation_end_date AS "End
Date", Pick_UpClerk AS "Pick-up Clerk", Drop_OffClerk AS "Drop-off Clerk",
DATEDIFF(reservation_start_date, reservation_end_date) AS "Number of Days",
ROUND(SUM(purchase_price * 0.4), 2) AS "Total Deposite Price",
ROUND(SUM(purchase_price * 0.15 * DATEDIFF(reservation_start_date,
reservation_end_date)), 2) AS "Total Rental Price"
FROM Customer NATURAL JOIN Reservation_Order_Tools
WHERE Customer.username = '$Username';
```

Get Tool Details







Abstract Code

- Customer or Clerk clicks on full description link
- Run the **Get Tool Details** task: query for information about the tool details where $tool_id is the ID of the current tool.
    - Find the current Tools using the Tools.tool_id
    - Display Tool Details, Tool ID, Tool Type, Short Description, Full Description, Deposit Price, Rental Price and Accessories (if it is a power source)
    - If PowerTools are selected:

```
SELECT T.tool_id AS "Tool ID", Toolinfo.tool_type AS "Tool Type",
CONCAT(T.power_source, Toolinfo.tool_subtype,
Toolinfo.tool_suboption) AS "Short Description",
CONCAT(T.width_or_diameter, ' in.W× ', T.length, ' in.L ', T.weight, '
lb. ', T.power_source, Toolinfo.tool_suboption, Toolinfo.tool_subtype,
PowerTools.volt_rating, ' Volt ', PowerTools.amp_rating, ' Amp ',
PowerTools.min_rpm_rating, ' RPM ', PowerTools.max_rpm_rating , '
ft_lb ', ' by ', T.manufacturer) AS "Full Description", ROUND
((T.purchase_price * .40), 2) AS "Deposit Price", ROUND
((T.purchase_price * .15), 2) AS "Rental Price", CONCAT("(",
Power_Accessories.quantity, ")", PowerTools.volt_rating, ' Volt ',
PowerTools.amp_rating, ' Amp ', Power_Accessories. battery_type,
accerssory_description) AS "Accessories" FROM ((Tools as T INNER
JOIN Toolinfo on T.tool_id = Toolinfo.tool_id) INNER JOIN
PowerTools on T.tool_id = PowerTools.tool_id INNER JOIN
Power_Accessories on T.tool_id = Power_Accessories.tool_id)
WHERE T.power_source != "manual";
```

    - If Manual Tools are selected:

```
SELECT T.tool_id AS "Tool ID", Toolinfo.tool_type AS "Tool Type",
CONCAT(Toolinfo.tool_subtype, Toolinfo.tool_suboption) AS "Short
Description", CONCAT(T.width_or_diameter, ' in.W× ', T.length, ' in.L
', T.weight, ' lb. ', Toolinfo.tool_suboption, Toolinfo.tool_subtype, ' by ',
T.manufacturer) AS "Full Description", ROUND ((T.purchase_price *
.40), 2) AS "Deposit Price", ROUND ((T.purchase_price * .15), 2) AS
"Rental Price" FROM Tools as T INNER JOIN Toolinfo on T.tool_id =
Toolinfo.tool_id WHERE T.power_source = 'manual';
```

Check Tool Availability

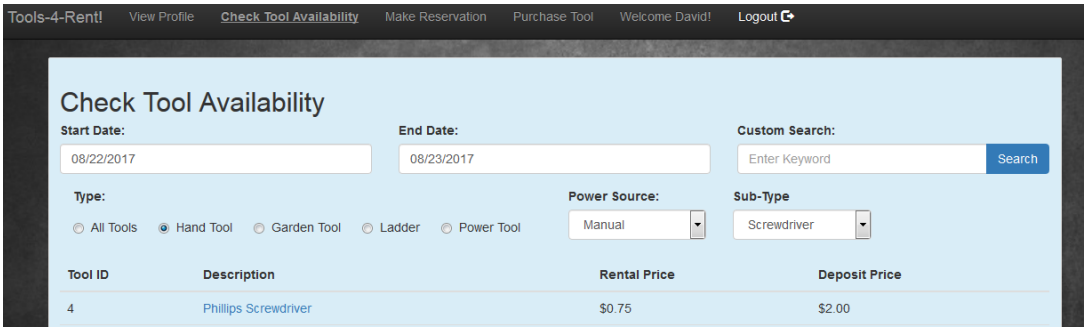
Abstract Code

- Customer clicks on ***Check Tool Availability*** button from **Customer Main Menu**:

Run the **Check Tool Availability** task: query for information about the available tool list where ToolID is the ID of the available tool using the system from the HTTP Session/Cookie.

- Find the available **Tool** using any combination of *start/end dates*, *tool category*, *power-source/sub-types*, and/or *keyword search* and clicking ***Search*** button; Display unique Tool ID, a Short-Description, Deposit price/ Reservation required and Rental Price/Day.

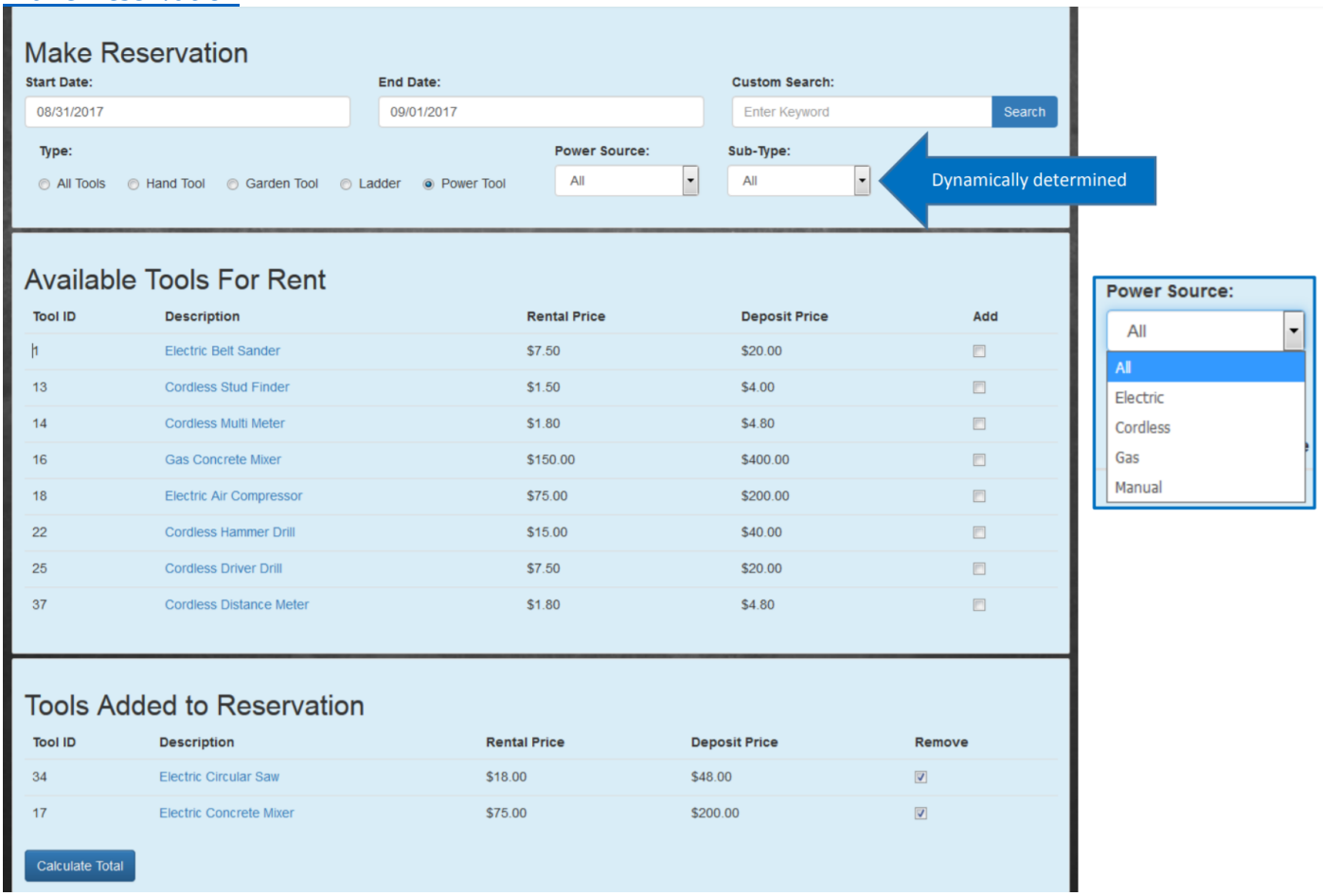
//if customer clicks ***Search*** button



When ready, customer selects next action from choices in **Customer Main Menu**

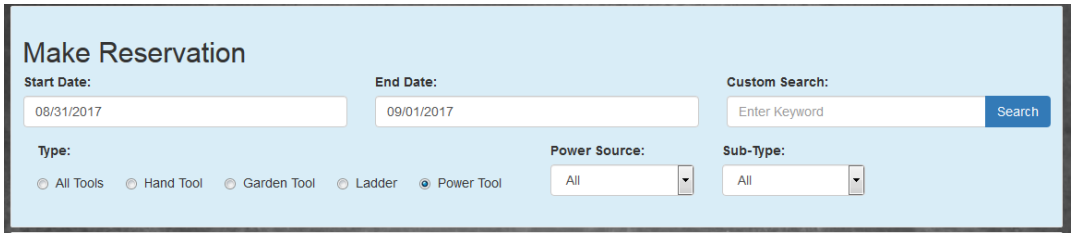
```
SELECT Tools.tool_id,
CONCAT (IF(Tools.power_source =  ' Manual ',   ' ',
CONCAT(Tools.power_source, ' ' )),
Toolinfo.tool_suboption,  ' ',  Toolinfo.tool_subtype)AS Description,
(Tools.purchase_price * 0.15) AS 'Rental Price',
(Tools.purchase_price * 0.4) AS 'Deposit Price'
FROM Tools
INNER JOIN Toolinfo ON Tools.tool_id = Toolinfo.tool_id
INNER JOIN Reservation ON Tools.tool_id = Reservation.tool_id
LEFT OUTER JOIN ServiceOrder ON Tools.tool_id =
ServiceOrder.tool_id
LEFT OUTER JOIN SaleOrder ON Tools.tool_id = SaleOrder.tool_id
WHERE
Tools.power_source =' $customerPowerSource '
AND Toolinfo.tool_type = ' $customerType '
AND Toolinfo.tool_subtype = ' $customerSubType '
AND Toolinfo.tool_suboption = ' $customerSubOption '
AND Reservation.reservation_start_date >  ' $customerEnteredEndDate '
OR Reservation.reservation_end_date <  ' $customerEnteredStartDate '
AND (Toolinfo.tool_type LIKE ' $searchKey' OR Toolinfo.tool_subtype
LIKE '$searchKEY ' OR Toolinfo.tool_suboption LIKE ' $searchKEY '
OR Tools.power_source LIKE ' $searchKEY ')
AND ServiceOrder.service_end_date < '$customerEnteredStartDate '
AND SaleOrder.for_sale_date = 'NULL'
ORDER BY Tools.tool  id;
```

Make Reservation



Abstract Code

- Customer clicks on ***Make Reservation*** button from **Customer Main Menu:**
- Run the **Make Reservation** task: query for information about the available tool list where Tool ID is the ID of the available tool using the system from the HTTP Session/Cookie.
    - Find the available Tool using any combination of *start/end dates*, *tool category*, *power-source/sub-types*, and/or *keyword search* and clicking ***Search*** button; Display unique Tool ID, a Short-Description, Deposit price/Reservation required and Rental Price/Day.
    -



//if customer clicks ***Search*** button

```
SELECT Tools.tool_id,
CONCAT (IF(Tools.power_source =  ' Manual ',   ' ',
CONCAT(Tools.power_source, ' ' )),
Toolinfo.tool_suboption,  ' ',  Toolinfo.tool_subtype)AS Description,
(Tools.purchase_price * 0.15) AS 'Rental Price',
(Tools.purchase_price * 0.4) AS 'Deposit Price'
FROM Tools
INNER JOIN Toolinfo ON Tools.tool_id = Toolinfo.tool_id
INNER JOIN Reservation ON Tools.tool_id = Reservation.tool_id
LEFT OUTER JOIN ServiceOrder ON Tools.tool_id =
ServiceOrder.tool_id
LEFT OUTER JOIN SaleOrder ON Tools.tool_id =
SaleOrder.tool_id
WHERE
Tools.power_source =' $customerPowerSource '
AND Toolinfo.tool_type = ' $customerType '
AND Toolinfo.tool_subtype = ' $customerSubType '
AND Toolinfo.tool_suboption = ' $customerSubOption '
AND Reservation.reservation_start_date >  '
$customerEnteredEndDate '
OR Reservation.reservation_end_date <  '
$customerEnteredStartDate '
AND (Toolinfo.tool_type LIKE ' $searchKey' OR
Toolinfo.tool_subtype LIKE '$searchKEY ' OR
Toolinfo.tool_suboption LIKE ' $searchKEY ' OR
Tools.power_source LIKE ' $searchKEY ')
AND ServiceOrder.service_end_date < '$customerEnteredStartDate '
AND SaleOrder.for_sale_date = 'NULL'
ORDER BY Tools.tool_id;
```

- If customer decides which tool to rent, customer clicks ***Add*** Checkbox; Display Tools Added to Reservation: Tool ID, Description, Rental Price, Deposit Price.

Available Tools For Rent

Tool ID	Description	Rental Price	Deposit Price	Add
1	Electric Belt Sander	$7.50	$20.00	☐
13	Cordless Stud Finder	$1.50	$4.00	☐
14	Cordless Multi Meter	$1.80	$4.80	☐
16	Gas Concrete Mixer	$150.00	$400.00	☐
18	Electric Air Compressor	$75.00	$200.00	☐
22	Cordless Hammer Drill	$15.00	$40.00	☐
25	Cordless Driver Drill	$7.50	$20.00	☐
37	Cordless Distance Meter	$1.80	$4.80	☐

Tools Added to Reservation

Tool ID	Description	Rental Price	Deposit Price	Remove
34	Electric Circular Saw	$18.00	$48.00	☑
17	Electric Concrete Mixer	$75.00	$200.00	☑

Calculate Total

//if customer clicks ***Add*** checkbox, UI updates the list of Tools Added to Reservation

```
SELECT Tools.tool_id,
CONCAT (IF(Tools.power_source =  ' Manual ',   ' ',
CONCAT(Tools.power_source, '    ' )),
Toolinfo.tool_suboption,  '    ',  Toolinfo.tool_subtype)AS
Description,
(Tools.purchase_price * 0.15) AS 'Rental Price',
(Tools.purchase_price * 0.4) AS 'Deposit Price'
FROM Tools
INNER JOIN Toolinfo ON Tools.tool_id = Toolinfo.tool_id
WHERE Tools.tool_id = ' $toolID '
ORDER BY tool_id;
```

- If customer wants to remove the tools he/she selected, customer clicks ***Remove*** Checkbox; Display Tools Added to Reservation: Tool ID, Description, Rental Price, Deposit Price.

//if customer clicks ***Remove*** checkbox, UI updates the list of Tools Added to Reservation

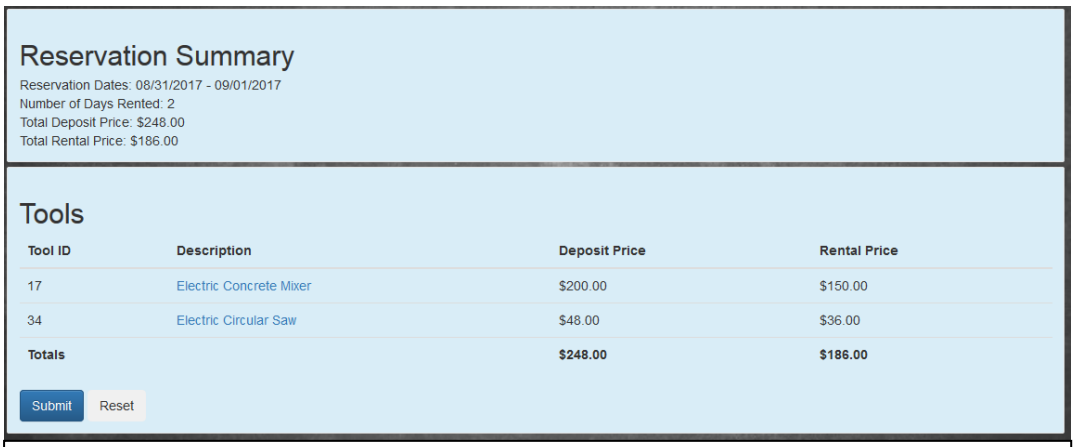
```
SELECT Tools.tool_id,
CONCAT (IF(Tools.power_source =  ' Manual ',   ' ',
CONCAT(Tools.power_source, '    ' )),
Toolinfo.tool_suboption,  '    ',  Toolinfo.tool_subtype)AS
Description,
(Tools.purchase_price * 0.15) AS 'Rental Price',
(Tools.purchase_price * 0.4) AS 'Deposit Price'
FROM Tools
INNER JOIN Toolinfo ON Tools.tool_id = Toolinfo.tool_id
WHERE Tools.tool_id = ' $toolID '
ORDER BY tool_id;
```

Reservation Summary & Confirmation

Abstract Code

- Customer finds the Total Rent Price by clicking ***Calculate Total*** button; Display Reservation Summary: Reservation Start and End Dates, Number of Days Rented, Total Deposit Price, Total Rental Price, Tool ID, a Short-Description, Deposit Price/Reservation, Rental Price/Reservation, Totals.

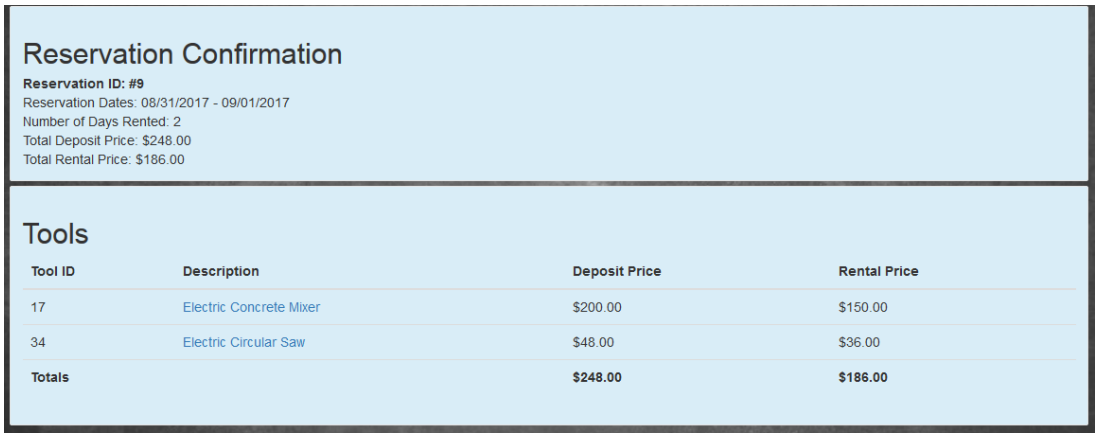
//if customer clicks ***Calculate Total*** button, UI displays Reservation Summary




```
SELECT Tools.tool_id,
CONCAT (IF(Tools.power_source =  ' Manual ',   ' ',
CONCAT(Tools.power_source, '    ' )),
Toolinfo.tool_suboption,  '    ',  Toolinfo.tool_subtype)AS
Description,
(Tools.purchase_price * 0.4) AS 'Deposit Price',
SUM((Tools.purchase_price * 0.15 * DATEDIFF('
$customerEnteredStartDate ' , ' $customerEnteredStartDate
')))AS 'Rental Price',
(SUM(Tools.purchase_price * 0.4) +
SUM((Tools.purchase_price * 0.15* DATEDIFF('
$customerEnteredStartDate ' , ' $customerEnteredStartDate '))))
AS Totals
FROM Tools
INNER JOIN Toolinfo ON Tools.tool_id = Toolinfo.tool_id
GROUP BY Tools.tool_id = ' $toolID '
ORDER BY Tools.tool_id;
```

- When ready, customer clicks ***Submit*** button to submit the reservation; Display Reservation Confirmation: Reservation ID, Reservation Start and End Dates, Number of Days Rented, Total Deposit Price, Total Rental Price, Tool ID, a Short-Description, Deposit Price/Reservation, Rental Price/Reservation, Totals.

```
INSERT INTO Reservation (tool_id, reservation_start_date, reservation_end_date,
customer_username) VALUES
(' $toolId ',  '$startDate ', ' $endDate ', ' $username ' );
```

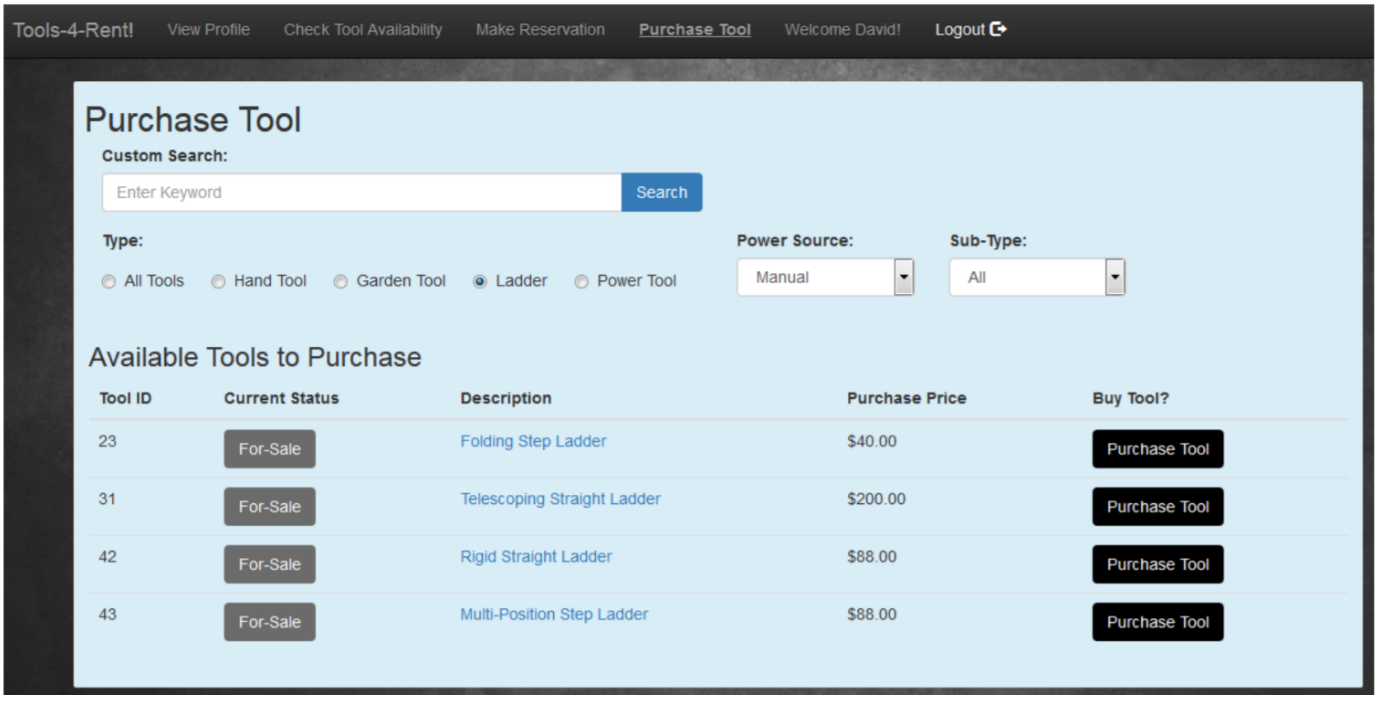


- If customer wants to print the Reservation Confirmation, customer clicks ***Print*** button.

```
SELECT Tools.tool_id,
CONCAT (IF(Tools.power_source =  ' Manual ',  ' ',
CONCAT(Tools.power_source, '  ' )),
Toolinfo.sub_option,  '  ',  Toolinfo.sub_type)AS Description,
(Tools.purchase_price * 0.4) AS 'Deposit Price',
SUM((Tools.purchase_price * 0.15 * DATEDIFF('
$customerEnteredStartDate ' , ' $customerEnteredStartDate ')))AS
'Rental Price',
(SUM(Tools.purchase_price * 0.4) + SUM((Tools.purchase_price *
0.15* DATEDIFF(' $customerEnteredStartDate ' , '
$customerEnteredStartDate '))))  AS Totals
FROM Tools
INNER JOIN Toolinfo ON Tools.tool_id = Toolinfo.tool_id
GROUP BY Tools.tool_id = ' $toolID '
ORDER BY Tools.tool_id;
```

- If customer wants to cancel the reservation, customer clicks ***Reset*** button and go back to the **Make Reservation** form.
  DELETE FROM Reservation WHERE reservation_id = ' $reservationID ' AND tool_id = ' $toolID ' AND reservation_start_date = ' $customerEnteredStartDate ' AND reservation_end_date = ' $customerEnteredEndDate ';

Purchase Tool



Abstract Code

- Customer clicks on ***Purchase Tool*** button from **Customer Main Menu**:
- Run the **Purchase Tool** task: query for information about the available tool to purchase list where $ToolID is the ID of tool using the system from the HTTP Session/Cookie.
  - Restrict the sale **Tool** using any combination of *tool category*, *power-source/sub-types*, and/or *keyword search*; Display unique Tool ID, Current Status, a Short-description, Purchase Price.

```
SELECT Tools.tool_id,
CONCAT (IF(Tools.power_source =  ' Manual ',   ' ',
CONCAT(Tools.power_source, ' ' )),
Toolinfo.tool_suboption,  ' ',  Toolinfo.tool_subtype)AS Description,
(Tools.purchase_price * 0.15) AS 'Rental Price',
(Tools.purchase_price * 0.4) AS 'Deposit Price'
FROM Tools
INNER JOIN Toolinfo ON Tools.tool_id = Toolinfo.tool_id
INNER JOIN Reservation ON Tools.tool_id = Reservation.tool_id
LEFT OUTER JOIN ServiceOrder ON Tools.tool_id = ServiceOrder.tool_id
LEFT OUTER JOIN SaleOrder ON Tools.tool_id = SaleOrder.tool_id
WHERE
Tools.power_source =' $customerPowerSource '
AND Toolinfo.tool_type = ' $customerType '
AND Toolinfo.tool_subtype = ' $customerSubType '
AND Toolinfo.tool_suboption = ' $customerSubOption '
AND Reservation.reservation_start_date >  ' $customerEnteredEndDate '
OR Reservation.reservation_end_date <  ' $customerEnteredStartDate '
AND (Toolinfo.tool_type LIKE ' $searchKey' OR Toolinfo.tool_subtype LIKE
'$searchKEY ' OR Toolinfo.tool_suboption LIKE ' $searchKEY ' OR
Tools.power_source LIKE ' $searchKEY ')
AND ServiceOrder.service_end_date < '$customerEnteredStartDate '
AND SaleOrder.for_sale_date = 'NULL'
ORDER BY Tools.tool_id;
```

- If customer wants to purchase tool, click ***Purchase Tool*** button and the corresponding '$tool_id' is stored in a temporary table 'PurchasedTools'; Display Purchase Summary: Purchase ID, Purchase Date, Tool ID, a Short-Description, Purchase Price, Total Purchase Price.

```
SELECT sale_id AS "Purchase ID", CURDATE() AS "Purchase Date", tool_id AS
"Tool ID", CONCAT (IF(power_source = 'Manual' , '  ',  CONCAT(power_source,
' ' )),  sub_option,  ' ',  sub_type) AS "Description", SUM(purchase_price * 0.5) AS
"Total Purchase Price"
FROM (PurchasedTools NATURAL JOIN SaleOrder) NATURAL JOIN Tools;
```

Purchase Summary & Confirmation

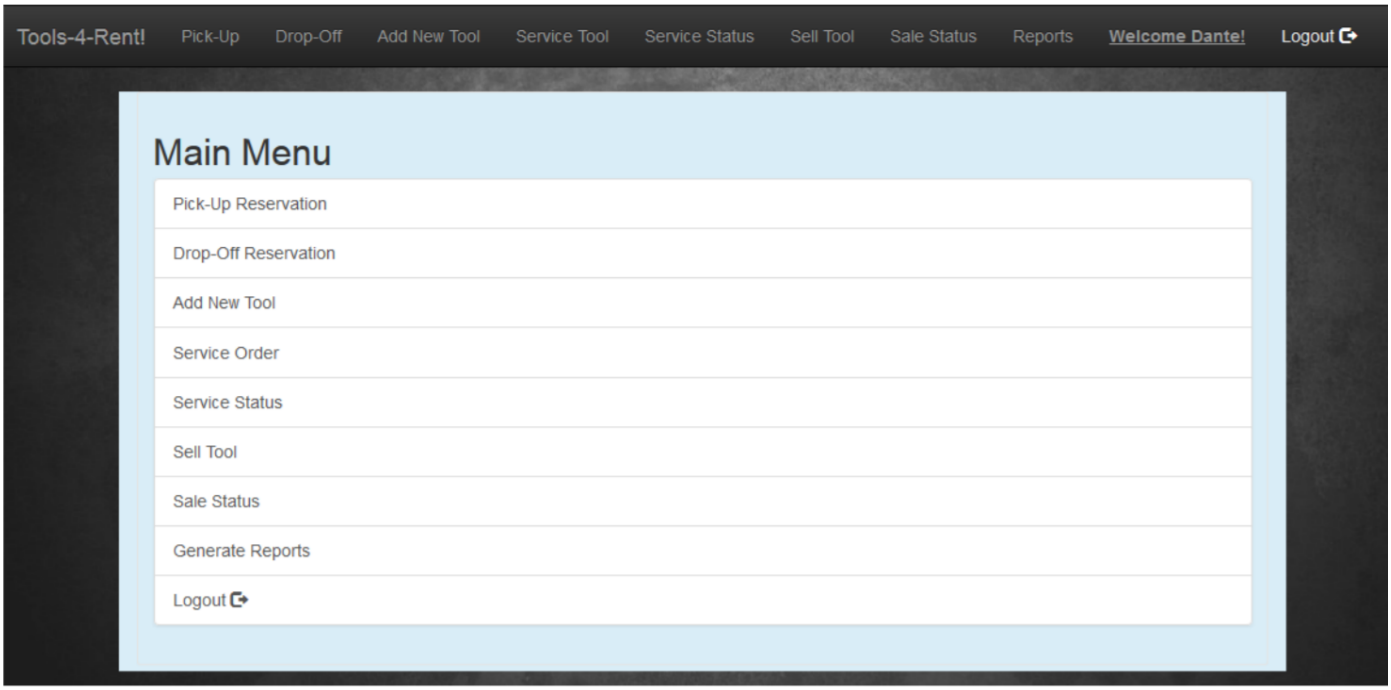
Abstract Code

- When customer clicks ***Submit*** *(purchase)* button to submit the purchase. Display Purchase Confirmation: Purchase ID, Purchase Date, Total Purchase Price, Tool ID, a Short-Description, Total Purchase Price.

```
UPDATE SaleOrder
SET sold_date = CURDATE()
WHERE tool_id = '$tool_id';

SELECT sale_id AS "Purchase ID", sold_date AS "Purchase Date", tool_id AS
"Tool ID", CONCAT (IF(power_source = 'Manual'  , '  ',
CONCAT(power_source, ' ' )),  sub_option,  '  ',  sub_type) AS "Description",
SUM(purchase_price * 0.5) AS "Total Purchase Price"
FROM PurchasedTools NATURAL JOIN SaleOrder NATURAL JOIN Tools
NATURAL JOIN ToolInfo;
```

- If customer wants to cancel the purchase, customer clicks ***Reset*** button and go back to the **Purchase Tool** form.
- If customer wants to print the Reservation Confirmation, customer clicks ***Print*** button.
- If customer wants to cancel the purchase, click ***Reset*** button and go back to the **Purchase Tool** form.

Clerk Main Menu



Abstract Code

- Show "***Pick-Up Reservation***", "***Drop-off Reservation***", "***Add New Tool***", "***Service Order***", "***Service Status***", "***Sell Tool***", "***Sale Status***", "***Generate Reports***", and "***Logout***" tabs.
- Upon:
  - 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** 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 ***Generate Reports*** button **–** Jump to the **Generate Reports** task**.**
  - Click ***Logout*** button **–** Invalidate login session and go back to the **Login** form.

Pick Up Reservations

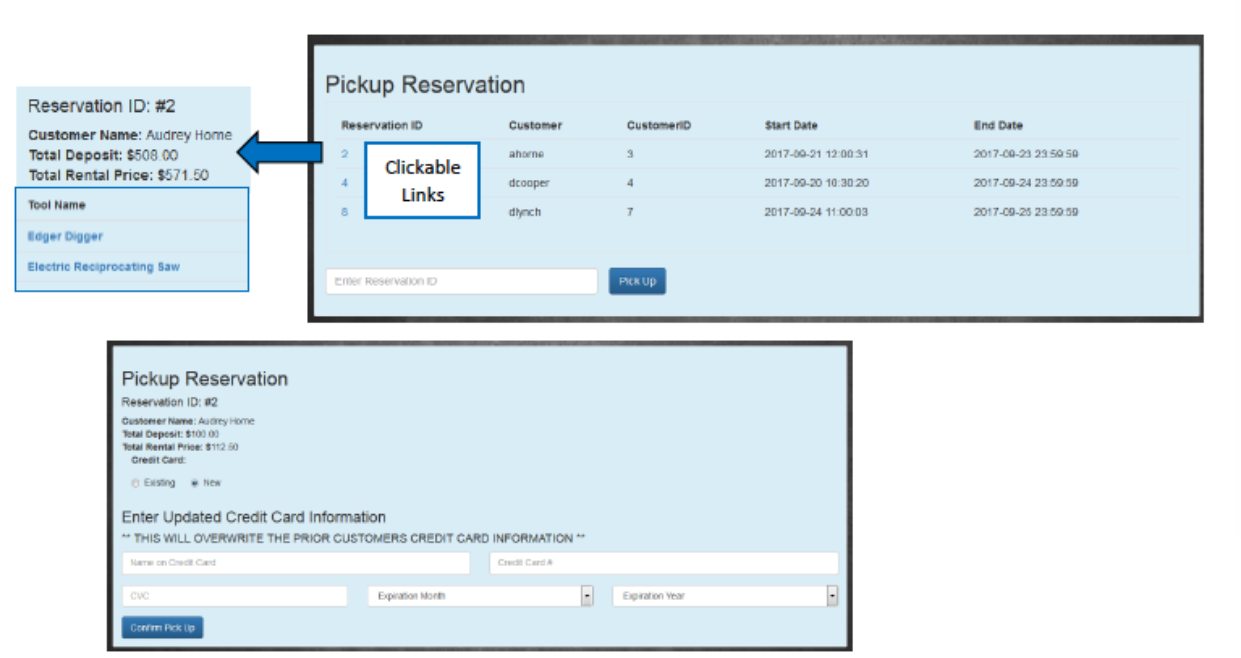


Figure 12 - Pick-up Reservation (with drop-down expiration month)

Abstract Code

- Clerk User clicks on ***Pick-Up Reservation*** button from **Clerk Main Menu:**
- Clerk User clicks the reservation ID to view the reservation information

```
SELECT Tools.tool_id,
CONCAT (IF(Tools.power_source = ' Manual ',  ' ',  CONCAT(Tools.power_source, '   ')),

Toolinfo.tool_suboption,  '   ',  Toolinfo.tool_subtype)AS Description,

(Tools.purchase_price * 0.4) AS 'Deposit Price',
SUM((Tools.purchase_price * 0.15 * DATEDIFF(' $customerEnteredStartDate ' , '
$customerEnteredStartDate ')))AS 'Rental Price', (SUM(Tools.purchase_price * 0.4) +
SUM((Tools.purchase_price * 0.15* DATEDIFF(' $customerEnteredStartDate ' , '
$customerEnteredStartDate ')))) AS Totals
FROM Reservation
INNER JOIN Tools ON Reservation.tool_id = Tools.tool_id
INNER JOIN Toolinfo ON Tools.tool_id = Toolinfo.tool_id
WHERE Reservation.reservation_id = ' $reservationID '
```

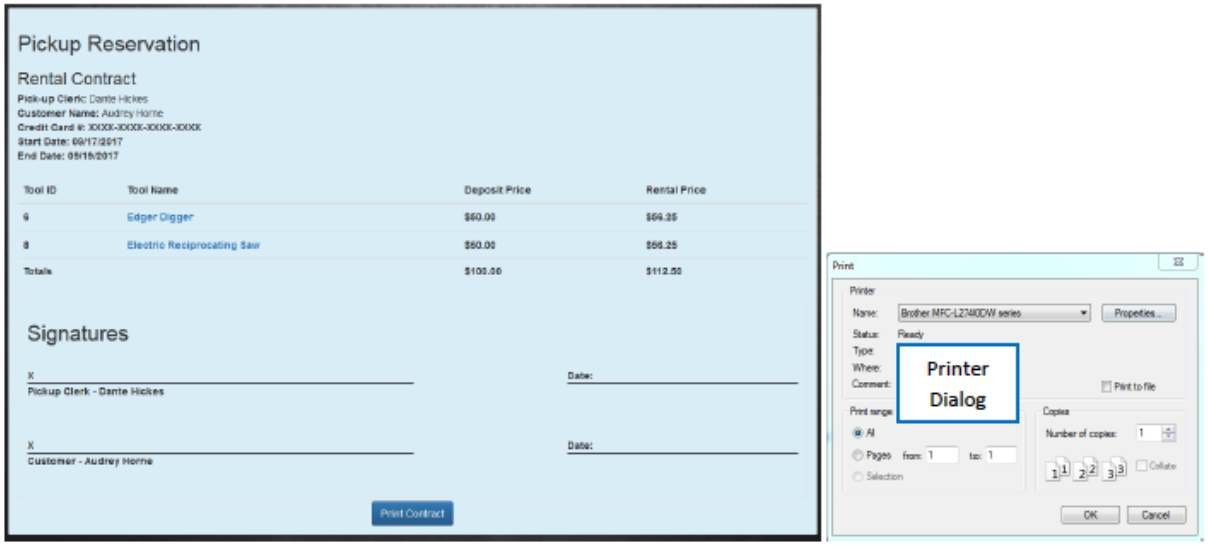
- Clerk User enters the reservation ID and click the ***Pick Up*** button.
- Clerk User clicks the payment option category with two radio buttons: ***Existing*** and ***New***.
    - If ***Existing*** is selected
        - Fields including *Name on Credit Card*, *Credit Card #*, *CVC*, *Expiration Month and Expiration Year*(drop-down menus for *Expiration Month and Expiration Year*) were inputed automatically

```
SELECT Customer.name_on_card, Customer.credit_card_number,
Customer.cvc_number, Customer.expiration_date FROM Reservation
JOIN Customer ON Reservation.customer_username = Customer.username
WHERE Reservation.reservation_id = '$reservation_id'
```

- Clerk User click the confirm ***Pick Up*** button.

- If ***New*** is selected:
  - Clerk User inputs *Name on Credit Card, Credit Card #, CVC, Expiration Month and Expiration Year*(drop-down menus for *Expiration Month and Expiration Year*) into input fields.
  - Clerk User click the confirm ***Pick Up*** button.

```
Update Customer
Set Customer.name_on_card= '$name_on_card',
Customer.credit_card_number= '$credit_card_number',
Customer.cvc_number= '$cvc_number',
 Customer.expiration_date= '$expiration_date'
WHERE Customer.username = '$username'
```



- Clerk User clicks the ***Print Contract*** button to print the contract information.
- Clerk User clicks the tool's name to view the tool information
  - If Manual Tools are selected:

```
SELECT T.tool_id AS "Tool ID", Toolinfo.tool_type AS "Tool Type",
CONCAT(Toolinfo.tool_subtype, Toolinfo.tool_suboption) AS "Short
Description", CONCAT(T.width_or_diameter, ' in.W× ', T.length, ' in.L ',
T.weight, ' lb. ', Toolinfo.tool_suboption, Toolinfo.tool_subtype, ' by ',
T.manufacturer) AS "Full Description", ROUND ((T.purchase_price * .40),
2) AS "Deposit Price", ROUND ((T.purchase_price * .15), 2) AS "Rental
Price" FROM Tools as T INNER JOIN Toolinfo on T.tool_id =
Toolinfo.tool_id WHERE T.power_source = 'manual';
```

- If Power Tools are selected:

```
SELECT T.tool_id AS "Tool ID", Toolinfo.tool_type AS "Tool Type",
CONCAT(T.power_source, Toolinfo.tool_subtype,
Toolinfo.tool_suboption) AS "Short Description",
CONCAT(T.width_or_diameter, ' in.W× ', T.length, ' in.L ', T.weight, ' lb. ',
T.power_source, Toolinfo.tool_suboption, Toolinfo.tool_subtype,
PowerTools.volt_rating, ' Volt ', PowerTools.amp_rating, ' Amp ',
PowerTools.min_rpm_rating, ' RPM ', PowerTools.max_rpm_rating , ' ft_lb
', ' by ', T.manufacturer) AS "Full Description", ROUND ((T.purchase_price
* .40), 2) AS "Deposit Price", ROUND ((T.purchase_price * .15), 2) AS
"Rental Price", CONCAT("(", Power_Accessories.quantity, ")",
PowerTools.volt_rating, ' Volt ', PowerTools.amp_rating, ' Amp ',
Power_Accessories. battery_type, accerssory_description) AS "Accessories"
FROM ((Tools as T INNER JOIN Toolinfo on T.tool_id = Toolinfo.tool_id)
INNER JOIN PowerTools on T.tool_id = PowerTools.tool_id INNER JOIN
Power_Accessories on T.tool_id = Power_Accessories.tool_id) WHERE
T.power_source != "manual";
```

Drop Off Reservation

Figure 14 - Drop-off Reservation (with all REQUIRED attributes shown on Tool Details)

Abstract Code

- Clerk User clicks on ***Drop-Off Reservation*** button from **Clerk Main Menu:**
- Run the **DropOffReservation** task:
  - Clerk User inputs Reservation ID.
  - Clerk User clicks ***Drop-Off*** button, find the corresponding reservation; Display the Tool ID, Tool Name, Deposit Price, Rental Price.

```
SELECT Tools.tool_id,
CONCAT (IF(Tools.power_source = ' Manual ',  ' ',  CONCAT(Tools.power_source, '
' )),

Toolinfo.tool_suboption,  '   ',  Toolinfo.tool_subtype)AS Description,

(Tools.purchase_price * 0.4) AS 'Deposit Price',
SUM((Tools.purchase_price * 0.15 * DATEDIFF('
$customerEnteredStartDate ' , ' $customerEnteredStartDate ')))AS 'Rental
Price', (SUM(Tools.purchase_price * 0.4) + SUM((Tools.purchase_price *
0.15* DATEDIFF(' $customerEnteredStartDate ' , '
$customerEnteredStartDate '))))  AS Totals
FROM Reservation
INNER JOIN Tools ON Reservation.tool_id = Tools.tool_id
INNER JOIN Toolinfo ON Tools.tool_id = Toolinfo.tool_id
WHERE Reservation.reservation_id = ' $reservationID '
```

- Clerk User clicks ***Drop-Off*** button, print out the receipt.
- Clerk User clicks the tool's name to view the tool information

    - If Manual Tools are selected:

```
SELECT T.tool_id AS "Tool ID", Toolinfo.tool_type AS "Tool Type",
CONCAT(Toolinfo.tool_subtype, Toolinfo.tool_suboption) AS "Short
Description", CONCAT(T.width_or_diameter, ' in.W× ', T.length, ' in.L ',
T.weight, ' lb. ', Toolinfo.tool_suboption, Toolinfo.tool_subtype, ' by ',
T.manufacturer) AS "Full Description", ROUND ((T.purchase_price * .40),
2) AS "Deposit Price", ROUND ((T.purchase_price * .15), 2) AS "Rental
Price" FROM Tools as T INNER JOIN Toolinfo on T.tool_id =
Toolinfo.tool_id WHERE T.power_source = 'manual';
```

    - If Power Tools are selected:

```
SELECT T.tool_id AS "Tool ID", Toolinfo.tool_type AS "Tool Type",
CONCAT(T.power_source, Toolinfo.tool_subtype,
Toolinfo.tool_suboption) AS "Short Description",
CONCAT(T.width_or_diameter, ' in.W× ', T.length, ' in.L ', T.weight, ' lb. ',
T.power_source, Toolinfo.tool_suboption, Toolinfo.tool_subtype,
PowerTools.volt_rating, ' Volt ', PowerTools.amp_rating, ' Amp ',
PowerTools.min_rpm_rating, ' RPM ', PowerTools.max_rpm_rating , ' ft_lb
', ' by ', T.manufacturer) AS "Full Description", ROUND ((T.purchase_price
* .40), 2) AS "Deposit Price", ROUND ((T.purchase_price * .15), 2) AS
"Rental Price", CONCAT("(", Power_Accessories.quantity, ")",
PowerTools.volt_rating, ' Volt ', PowerTools.amp_rating, ' Amp ',
Power_Accessories. battery_type, accerssory_description) AS "Accessories"
FROM ((Tools as T INNER JOIN Toolinfo on T.tool_id = Toolinfo.tool_id)
INNER JOIN PowerTools on T.tool_id = PowerTools.tool_id INNER JOIN
Power_Accessories on T.tool_id = Power_Accessories.tool_id) WHERE
T.power_source != "manual";
```

Add Tool

Abstract Code

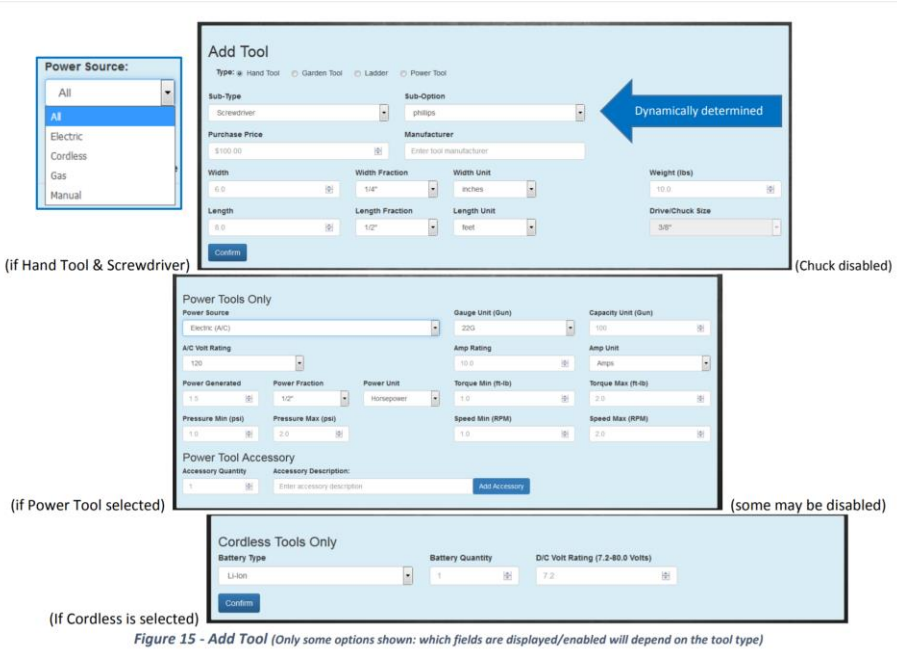


Figure 15 - Add Tool (Only some options shown: which fields are displayed/enabled will depend on the tool type)

Abstract Code

- Clerk clicks on ***Add New Tool*** button from **Clerk Main Menu:**
- Run the **Add Tool** task: clerk chooses the tool category with four radio buttons: ***Hand Tool***, ***Garden Tool***, ***Ladder***, and ***Power Tool***.
    - If ***Hand Tool***($tool_type) and ***Manual*** Power Source($power_source) are selected**:**
        - If Clerk chooses the ScrewDriver ($tool_subtype):
            - Clerk chooses the *Sub-Option* ($tool_suboption)
            - Clerk inputs *purchase_price* ($purchase_price), *manufacturer*($manufacture), *material*($material), *width* ($width)drop-down menus for *Width Fraction* and *Width Unit*), *length*($length) (drop-down menus for *Length Fraction* and *Length Unit*), *weight*($weight) and *screw_size*($screw_size) into input fields.

```
INSERT INTO Tools (purchase_price, manufacturer, material, width,
length, weight, power_source)
VALUES('$purchase_price', '$manufacturer', '$material', '$width',
'$length', '$weight', '$power_source');
```

```
INSERT INTO ScrewDriver(screw_size)
VALUES ('$screw_size');
```

```
INSERT INTO ToolInfo(tool_type, tool_subtype, tool_suboption)
VALUES ('$tool_type', '$tool_subtype', 'tool_suboption');
```

- Else if Clerk chooses the Socket (Sub_Type):
  - Clerk chooses the Sub-Option
  - Clerk inputs *purchase_price* ($purchase_price), *manufacturer*($manufacture), *material*($material), *width* ($width) drop-down menus for *Width Fraction* and *Width Unit*), *length(*$length*)* (drop-down menus for *Length Fraction* and *Length Unit*), *weight*($weight*)*, *drive_size* ($drive_size), *sae_size*($sae_size), *deep_socket* ($deep_socket) into input fields.

```
INSERT INTO Tools (purchase_price, manufacturer, material,
width, length, weight, power_source)
VALUES('$purchase_price', '$manufacturer', '$material', '$width',
'$length', '$weight', '$power_source');
```

```
INSERT INTO Socket (drive_size, sae_size, deep_socket)
VALUES ('$drive_size', '$sae_size', '$deep_socket');
```

```
INSERT INTO ToolInfo(tool_type, tool_subtype, tool_suboption)
VALUES ('$tool_type', '$tool_subtype', 'tool_suboption');
```

- Else if Clerk chooses the Ratchet (Sub_Type):
  - Clerk chooses the Sub-Option
  - Clerk inputs *purchase_price* ($purchase_price), *manufacturer*($manufacture), *width* ($width) drop-down menus for *Width Fraction* and *Width Unit*), *length(*$length*)* (drop-down menus for *Length Fraction* and *Length Unit*), *weight*($weight*)* and *drive_size*($drive_size) into input fields.

```
INSERT INTO Tools (purchase_price, manufacturer, material,
width, length, weight, power_source)
VALUES('$purchase_price', '$manufacturer', '$material', '$width',
'$length', '$weight', '$power_source');
```

```
INSERT INTO Ratchet (drive_size)
VALUES ('$drive_size');
```

```
INSERT INTO ToolInfo(tool_type, tool_subtype, tool_suboption)
VALUES ('$tool_type', '$tool_subtype', 'tool_suboption');
```

- Else if Clerk chooses the Wrench (Sub_Type):

- Clerk chooses the Sub-Option
- Clerk inputs *purchase_price* ($purchase_price), *manufacturer*($manufacture*)*, *width* ($width) drop-down menus for *Width Fraction* and *Width Unit*), *length(*$length*)* (drop-down menus for *Length Fraction* and *Length Unit*), *weight*($weight*)* into input fields.
- ```
  INSERT INTO Tools (purchase_price, manufacturer, material,
  width, length, weight, power_source)
  VALUES('$purchase_price', '$manufacturer', '$material', '$width',
  '$length', '$weight', '$power_source');
  ```
- ```
  INSERT INTO ToolInfo(tool_type, tool_subtype, tool_suboption)
  VALUES ('$tool_type', '$tool_subtype', 'tool_suboption');
  ```

▪ Else if Clerk chooses the Pillers (Sub_Type):

- Clerk chooses the Sub-Option
- Clerk inputs *purchase_price* ($purchase_price), *manufacturer*($manufacture*)*, *width* ($width) drop-down menus for *Width Fraction* and *Width Unit*), *length(*$length*)* (drop-down menus for *Length Fraction* and *Length Unit*), *weight*($weight*)* and *adjustable* ($adjustable) into input fields.
- ```
  INSERT INTO Tools (purchase_price, manufacturer, material,
  width, length, weight, power_source)
  VALUES('$purchase_price', '$manufacturer', '$material', '$width',
  '$length', '$weight', '$power_source');
  ```
- ```
  INSERT INTO Pillers (adjustable)
  VALUES ('$adjustable');
  ```
- ```
  INSERT INTO ToolInfo(tool_type, tool_subtype, tool_suboption)
  VALUES ('$tool_type', '$tool_subtype', 'tool_suboption');
  ```

▪ Else if Clerk chooses the Gun (Sub_Type):

- Clerk chooses the Sub-Option
- Clerk inputs *purchase_price* ($purchase_price), *manufacturer*($manufacture*)*, *width* ($width) drop-down menus for *Width Fraction* and *Width Unit*), *length(*$length*)* (drop-down menus for *Length Fraction* and *Length Unit*), *weight*($weight*)* and *gauge_rating* ($gauge_rating) and *capacity* ($capacity) into input fields.
- ```
  INSERT INTO Tools (purchase_price, manufacturer, material,
  width, length, weight, power_source)
  VALUES('$purchase_price', '$manufacturer', '$material', '$width',
  '$length', '$weight', '$power_source');
  ```
- ```
  INSERT INTO Gun (gauge_rating, capacity)
  VALUES ('$gauge_rating', '$capacity');
  ```
- ```
  INSERT INTO ToolInfo(tool_type, tool_subtype, tool_suboption)
  VALUES ('$tool_type', '$tool_subtype', 'tool_suboption');
  ```

▪ Else if Clerk chooses the Hammer(Sub_Type):

- Clerk chooses the Sub-Option
- Clerk inputs *purchase_price* ($purchase_price), *manufacturer*($manufacture*)*, *width* ($width) drop-down menus for *Width Fraction* and *Width Unit*), *length(*$length*)* (drop-down menus for *Length Fraction* and *Length Unit*), *weight*($weight*)* and *anti_vibration*($anti_vibration) into input fields.

```
INSERT INTO Tools (purchase_price, manufacturer, material,
width, length, weight, power_source)
VALUES('$purchase_price', '$manufacturer', '$material', '$width',
'$length', '$weight', '$power_source');
```

```
INSERT INTO Pillers (adjustable)
VALUES ('$anti_vibration');
```

```
INSERT INTO ToolInfo(tool_type, tool_subtype, tool_suboption)
VALUES ('$tool_type', '$tool_subtype', 'tool_suboption');
```

- Else if ***Garden Tool*** and ***Manual*** Power Source are selected**:**
  - **Clerk** inputs *handle_material* ($handle_material**) into the field.**

```
INSERT INTO GardenTools(handel_material)
VALUES ('$handel_material');
```

- If Clerk chooses the Digger (Sub_Type):
  - Clerk chooses the Sub-Option
  - Clerk inputs *purchase_price* ($purchase_price), *manufacturer*($manufacture*)*, *width* ($width) drop-down menus for *Width Fraction* and *Width Unit*), *length(*$length*)* (drop-down menus for *Length Fraction* and *Length Unit*), *weight*($weight*), blade_weight*($blade_weight*), blade_length*($blade_length) into input fields.

```
INSERT INTO Tools (purchase_price, manufacturer, material,
width, length, weight, power_source)
VALUES('$purchase_price', '$manufacturer', '$material', '$width',
'$length', '$weight', '$power_source');
```

```
INSERT INTO Digger(blade_weight, blade_length)
VALUES ('$blade_weight', '$blade_length');
```

```
INSERT INTO ToolInfo(tool_type, tool_subtype, tool_suboption)
VALUES ('$tool_type', '$tool_subtype', 'tool_suboption');
```

- Else if Clerk chooses the Pruner (Sub_Type):
  - Clerk chooses the Sub-Option
  - Clerk inputs *purchase_price* ($purchase_price), *manufacturer*($manufacture*)*, *width* ($width) drop-down menus for *Width Fraction* and *Width Unit*), *length(*$length*)* (drop-down menus for *Length Fraction* and *Length Unit*), *weight*($weight*), blade_material*($blade_material*),* *blade_length*($blade_length) into input fields.

```
INSERT INTO Tools (purchase_price, manufacturer, material,
width, length, weight, power_source)
VALUES('$purchase_price', '$manufacturer', '$material', '$width',
'$length', '$weight', '$power_source');
```

```
INSERT INTO Pruner(blade_material, blade_length)
VALUES ('$blade_material', '$blade_length');
```

```
INSERT INTO ToolInfo(tool_type, tool_subtype, tool_suboption)
VALUES ('$tool_type', '$tool_subtype', 'tool_suboption');
```

- Else if Clerk chooses the Rakes (Sub_Type):
  - Clerk chooses the Sub-Option
  - Clerk inputs *purchase_price* ($purchase_price), *manufacturer*($manufacture*)*, *width* ($width) drop-down menus for *Width Fraction* and *Width Unit*), *length(*$length*)* (drop-down menus for *Length Fraction* and *Length Unit*), *weight*($weight*), and tine_count*($tine_count*)* into input fields.

```
INSERT INTO Tools (purchase_price, manufacturer, material,
width, length, weight, power_source)
VALUES('$purchase_price', '$manufacturer', '$material', '$width',
'$length', '$weight', '$power_source');
```

```
INSERT INTO Rakes(tine_count)
VALUES ('$tine_count');
```

```
INSERT INTO ToolInfo(tool_type, tool_subtype, tool_suboption)
VALUES ('$tool_type', '$tool_subtype', 'tool_suboption');
```

- Else if Clerk chooses the Wheelbarrows (Sub_Type):
  - Clerk chooses the Sub-Option
  - Clerk inputs *purchase_price* ($purchase_price), *manufacturer*($manufacture*)*, *width* ($width) drop-down menus for *Width Fraction* and *Width Unit*), *length(*$length*)* (drop-down menus for *Length Fraction* and *Length Unit*), *weight*($weight*), bin_material*($bin_material*), bin_volume*($bin_volume) and wheel_count($wheel_count) into input fields.

```
INSERT INTO Tools (purchase_price, manufacturer, material,
width, length, weight, power_source)
VALUES('$purchase_price', '$manufacturer', '$material', '$width',
'$length', '$weight', '$power_source');
```

```
INSERT INTO Wheelbarrows (bin_material, bin_volume,
wheel_count)
VALUES ('$bin_material', '$bin_colume', '$wheel_count');
```

```
INSERT INTO ToolInfo(tool_type, tool_subtype, tool_suboption)
VALUES ('$tool_type', '$tool_subtype', 'tool_suboption');
```

- Else if Clerk chooses the Striking (Sub_Type):
  - Clerk chooses the Sub-Option
  - Clerk inputs *purchase_price* ($purchase_price), *manufacturer*($manufacture*)*, *width* ($width) drop-down menus for *Width Fraction* and *Width Unit*), *length(*$length*)* (drop-down menus for *Length Fraction* and *Length Unit*), *weight*($weight*)*, and *head_weight*($head_weight*)* into input fields.

```
INSERT INTO Tools (purchase_price, manufacturer, material,
width, length, weight, power_source)
VALUES('$purchase_price', '$manufacturer', '$material', '$width',
'$length', '$weight', '$power_source');
```

```
INSERT INTO Striking(head_weight)
VALUES ('$head_weight');
```

```
INSERT INTO ToolInfo(tool_type, tool_subtype, tool_suboption)
VALUES ('$tool_type', '$tool_subtype', 'tool_suboption');
```

- Else if ***Ladder*** and ***Manual*** Power Source are selected**:**
  - Clerk inputs *step_count*($step_count**)** and *weight_capacity*($weight_capacity) into the field.

```
INSERT INTO Ladder(step_count, weight_capacity)
VALUES ('$step_count', '$weight_capacity');
```

- If Clerk chooses the Straight (Sub_Type):
  - Clerk chooses the Sub-Option
  - Clerk inputs *purchase_price* ($purchase_price), *manufacturer*($manufacture*)*, *width* ($width) drop-down menus for *Width Fraction* and *Width Unit*), *length(*$length*)* (drop-down menus for *Length Fraction* and *Length Unit*), *weight*($weight*)* and *rubber_feet*($rubber_feet*)* into input fields.

```
INSERT INTO Tools (purchase_price, manufacturer, material,
width, length, weight, power_source)
VALUES('$purchase_price', '$manufacturer', '$material', '$width',
'$length', '$weight', '$power_source');
```

```
INSERT INTO Straight(rubber_feet)
VALUES ('$rubber_feet');
```

```
INSERT INTO ToolInfo(tool_type, tool_subtype, tool_suboption)
VALUES ('$tool_type', '$tool_subtype', 'tool_suboption');
```

- Else if Clerk chooses the Step (Sub_Type):
  - Clerk chooses the **Sub-Option**

- Clerk inputs *purchase_price* ($purchase_price), *manufacturer*($manufacture*)*, *width* ($width) drop-down menus for *Width Fraction* and *Width Unit*), *length(*$length*)* (drop-down menus for *Length Fraction* and *Length Unit*), *weight*($weight*)* and *pail_shelf(*$pail_shelf)into input fields.
-

```
INSERT INTO Tools (purchase_price, manufacturer, material,
width, length, weight, power_source)
VALUES('$purchase_price', '$manufacturer', '$material', '$width',
'$length', '$weight', '$power_source');
```

```
INSERT INTO Step(pail_shelf)
VALUES ('$pail_shelf');
```

```
INSERT INTO ToolInfo(tool_type, tool_subtype, tool_suboption)
VALUES ('$tool_type', '$tool_subtype', 'tool_suboption');
```

- Else if ***Power Tool*** is selected:
  - Clerk User chooses the *Power Source* by a drop-down menu.
    - If the Clerk chooses Drill (Sub_Type):
      - The Clerk chooses the **Sub-Option.**
      - Clerk User inputs *purchase_price* ($purchase_price), *manufacturer*($manufacture*)*, *width* ($width) drop-down menus for *Width Fraction* and *Width Unit*), *length(*$length*)* (drop-down menus for *Length Fraction* and *Length Unit*), *weight*($weight*), volt_rating* ($volt_rating), *amp_rating*($amp_rating), *min_rpm_rating* ($min_rpm_rating), *max_rpm_rating* ($max_rpm_rating), *quantity*($quantity) and *accessory_description* ($accessory_description) into input fields.
      -

```
INSERT INTO Tools (purchase_price, manufacturer, material,
width, length, weight, power_source)
VALUES('$purchase_price', '$manufacturer', '$material', '$width',
'$length', '$weight', '$power_source');
```

      -

```
INSERT INTO PowerTools(volt_rating, amp_rating,
min_rpm_rating, max_rpm_rating)
VALUES ('$volt_rating', '$amp_rating', '$min_rpm_rating',
'$max_rpm_rating');
```

      -

```
INSERT INTO Power_Accessories(quantity, accessory_description)
VALUES ('$quantity', 'accessory_description');
```

      -

```
INSERT INTO ToolInfo(tool_type, tool_subtype, tool_suboption)
VALUES ('$tool_type', '$tool_subtype', 'tool_suboption');
```

      - Clerk **input** *adjustable_clutch*($adjustable_clutch), *min_torque_rating*($min_torque_rating), *max_torque_rating*($max_torque_rating) into the fields.

```
INSERT INTO Drill(adjustable_clutch, min_torque_rating,
max_torque_rating)
VALUES ('$adjustable_clutch', '$min_torque_rating',
'$max_torque_rating');
```

- If the Clerk chooses **Electric (D/C)**:
  - Clerk **input** the battery_type ($battery_type) into the field

```
INSERT INTO Power_Accessories(battery_type)
VALUES ('$battery_type');
```

- Else if the Clerk chooses Saw (Sub_Type):
  - Clerk chooses the **Sub-Option.**
  - Clerk User inputs *purchase_price* ($purchase_price), *manufacturer*($manufacture*), width* ($width) drop-down menus for *Width Fraction* and *Width Unit*), *length(*$length*)* (drop-down menus for *Length Fraction* and *Length Unit*), *weight*($weight*), volt_rating* ($volt_rating), *amp_rating*($amp_rating), *min_rpm_rating* ($min_rpm_rating), *max_rpm_rating* ($max_rpm_rating), *quantity*($quantity) and *accessory_description* ($accessory_description) into input fields.

```
INSERT INTO Tools (purchase_price, manufacturer, material,
width, length, weight, power_source)
VALUES('$purchase_price', '$manufacturer', '$material', '$width',
'$length', '$weight', '$power_source');
```

```
INSERT INTO PowerTools(volt_rating, amp_rating,
min_rpm_rating, max_rpm_rating)
VALUES ('$volt_rating', '$amp_rating', '$min_rpm_rating',
'$max_rpm_rating');
```

```
INSERT INTO ToolInfo(tool_type, tool_subtype, tool_suboption)
VALUES ('$tool_type', '$tool_subtype', 'tool_suboption');
```

```
INSERT INTO Power_Accessories (quantity,
accessory_description)
VALUES ('$quantity', 'accessory_description');
```

  - Clerk **input** *blade_size* ($blade_size) into the fields.

```
INSERT INTO Saw(blade_size)
VALUES ('$bladez_size');
```

  - If the Clerk chooses **Electric (D/C)**:
    - Clerk **input** the battery_type ($battery_type) into the field

```
INSERT INTO Power_Accessories (battery_type)
VALUES ('$battery_type');
```

- If the Clerk chooses Sander (Sub_Type):
  - The Clerk chooses the **Sub-Option.**
  - Clerk User inputs *purchase_price* ($purchase_price), *manufacturer*($manufacture*), width* ($width) drop-down menus for *Width Fraction* and *Width Unit*), *length(*$length*)* (drop-down menus for *Length Fraction* and *Length Unit*), *weight*($weight*), volt_rating* ($volt_rating), *amp_rating*($amp_rating), *min_rpm_rating* ($min_rpm_rating), *max_rpm_rating* ($max_rpm_rating), *quantity*($quantity) and *accessory_description* ($accessory_description) into input fields.
  - ```
    INSERT INTO Tools (purchase_price, manufacturer, material,
    width, length, weight, power_source)
    VALUES('$purchase_price', '$manufacturer', '$material', '$width',
    '$length', '$weight', '$power_source');
    ```
  - ```
    INSERT INTO PowerTools(volt_rating, amp_rating,
    min_rpm_rating, max_rpm_rating)
    VALUES ('$volt_rating', '$amp_rating', '$min_rpm_rating',
    '$max_rpm_rating');
    ```
  - ```
    INSERT INTO ToolInfo(tool_type, tool_subtype, tool_suboption)
    VALUES ('$tool_type', '$tool_subtype', 'tool_suboption');
    ```
  - ```
    INSERT INTO Power_Accessories (quantity,
    accessory_description)
    VALUES ('$quantity', 'accessory_description');
    ```
  -
  - Clerk inputs *dust_bag* ($dust_bag**)**into the fields.
  - ```
    INSERT INTO Sander(dust_bag)
    VALUES ('$dust_bag');
    ```
  -
  - If the Clerk chooses **Electric (D/C)**:
    - Clerk **input** the battery_type ($battery_type) into the field
    - ```
      INSERT INTO Power_Accessories (battery_type)
      VALUES ('$battery_type');
      ```

- If the Clerk chooses Air_Compressor (Sub_Type):
  - The Clerk chooses the **Sub-Option.**
  - Clerk User inputs *purchase_price* ($purchase_price), *manufacturer*($manufacture*), width* ($width) drop-down menus for *Width Fraction* and *Width Unit*), *length(*$length*)* (drop-down menus for *Length Fraction* and *Length Unit*), *weight*($weight*), volt_rating* ($volt_rating), *amp_rating*($amp_rating), *min_rpm_rating* ($min_rpm_rating), *max_rpm_rating* ($max_rpm_rating), *quantity*($quantity) and *accessory_description* ($accessory_description) into input fields.

```
INSERT INTO Tools (purchase_price, manufacturer, material,
width, length, weight, power_source)
VALUES('$purchase_price', '$manufacturer', '$material', '$width',
'$length', '$weight', '$power_source');
```

```
INSERT INTO PowerTools(volt_rating, amp_rating,
min_rpm_rating, max_rpm_rating)
VALUES ('$volt_rating', '$amp_rating', '$min_rpm_rating',
'$max_rpm_rating');
```

```
INSERT INTO ToolInfo(tool_type, tool_subtype, tool_suboption)
VALUES ('$tool_type', '$tool_subtype', 'tool_suboption');
```

```
INSERT INTO Power_Accessories (quantity, accessory_description)
VALUES ('$quantity', 'accessory_description');
```

- Clerk **inputs** *tank_size*($tank_size), *pressure_rating(*$pressure_rating) into the fields.

```
INSERT INTO Air_Compressor (tank_size, pressure_rating)
VALUES ('$tank_size', '$pressure_rating');
```

- If the Clerk chooses Mixer(Sub_Type):
  - The Clerk chooses the **Sub-Option.**
  - Clerk User inputs *purchase_price* ($purchase_price), *manufacturer*($manufacture*), width* ($width) drop-down menus for *Width Fraction* and *Width Unit*), *length(*$length*)* (drop-down menus for *Length Fraction* and *Length Unit*), *weight*($weight*), volt_rating* ($volt_rating), *amp_rating*($amp_rating), *min_rpm_rating* ($min_rpm_rating), *max_rpm_rating* ($max_rpm_rating), *quantity*($quantity) and *accessory_description* ($accessory_description) into input fields.

```
INSERT INTO Tools (purchase_price, manufacturer, material,
width, length, weight, power_source)
VALUES('$purchase_price', '$manufacturer', '$material', '$width',
'$length', '$weight', '$power_source');
```

```
INSERT INTO PowerTools(volt_rating, amp_rating,
min_rpm_rating, max_rpm_rating)
VALUES ('$volt_rating', '$amp_rating', '$min_rpm_rating',
'$max_rpm_rating');
```

```
INSERT INTO ToolInfo(tool_type, tool_subtype, tool_suboption)
VALUES ('$tool_type', '$tool_subtype', 'tool_suboption');
```

```
INSERT INTO Power_Accessories (quantity, accessory_description)
VALUES ('$quantity', 'accessory_description');
```

- Clerk **input** *motor_rating* ($motor_rating*)*, *drum_size* ($drum_size*)* into the fields.

```
INSERT INTO Mixer(motor_rating, drum_size)
VALUES ('$motor_rating', '$drum_size');
```

- If the Clerk chooses Generator (Sub_Type):
  - The Clerk chooses the **Sub-Option.**
  - Clerk User inputs *purchase_price* ($purchase_price), *manufacturer*($manufacture*)*, *width* ($width) drop-down menus for *Width Fraction* and *Width Unit*), *length(*$length*)* (drop-down menus for *Length Fraction* and *Length Unit*), *weight*($weight*)*, *volt_rating* ($volt_rating), *amp_rating*($amp_rating), *min_rpm_rating* ($min_rpm_rating), *max_rpm_rating* ($max_rpm_rating), *quantity*($quantity) and *accessory_description* ($accessory_description) into input fields.

```
INSERT INTO Tools (purchase_price, manufacturer, material,
width, length, weight, power_source)
VALUES('$purchase_price', '$manufacturer', '$material', '$width',
'$length', '$weight', '$power_source');
```

```
INSERT INTO PowerTools(volt_rating, amp_rating,
min_rpm_rating, max_rpm_rating)
VALUES ('$volt_rating', '$amp_rating', '$min_rpm_rating',
'$max_rpm_rating');
```

```
INSERT INTO ToolInfo(tool_type, tool_subtype, tool_suboption)
VALUES ('$tool_type', '$tool_subtype', 'tool_suboption');
```

```
INSERT INTO Power_Accessories (quantity, accessory_description)
VALUES ('$quantity', 'accessory_description');
```

  - Clerk **input** *power_rating* ($power_rating*)* into the fields.

```
INSERT INTO Generator(power_rating)
VALUES ('$power_rating');
```

  - When all required fields are filled, Clerk User clicks on ***Confirm*** button to add new tool with description into database.

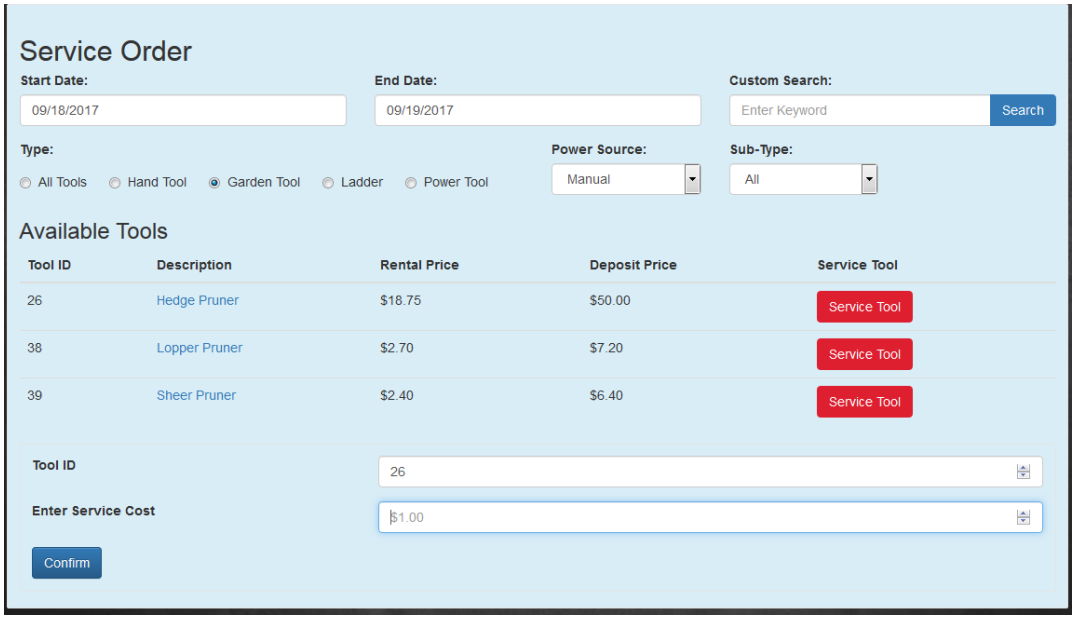
Repair Tool

Abstract Code

- Clerk clicks on ***Service Order*** button from **Clerk Main Menu**.

- Run the **Repair Tool** task: query for information about the available tool list where $ToolID is the ID of the available tool using the system from the HTTP Session/Cookie.
    - Find the available **Tool** using any combination of *start/end dates*, *tool category*, *power-source/sub-types*, and/or *keyword search* and clicking ***Search*** button; Display unique Tool ID, a Short-Description, Deposit price/ Reservation required and Rental Price/Day.

      //if customer clicks ***Search*** button



```
SELECT Tools.tool_id,
CONCAT (IF(Tools.power_source =  ' Manual ',   ' ',
CONCAT(Tools.power_source, ' ' )),
Toolinfo.tool_suboption,  ' ',  Toolinfo.tool_subtype)AS
Description,
(Tools.purchase_price * 0.15) AS 'Rental Price',
(Tools.purchase_price * 0.4) AS 'Deposit Price'
FROM Tools
INNER JOIN Toolinfo ON Tools.tool_id = Toolinfo.tool_id
INNER JOIN Reservation ON Tools.tool_id =
Reservation.tool_id
INNER JOIN ServiceOrder ON Tools.tool_id =
ServiceOrder.tool_id
LEFT OUTER JOIN SaleOrder ON Tools.tool_id =
SaleOrder.tool_id
WHERE
Tools.power_source =' $clerkPowerSource '
AND Toolinfo.tool_type = ' $clerkType '
AND Toolinfo.tool_subtype = ' $clerkSubType '
AND Toolinfo.tool_suboption = ' $clerkSubOption '
AND Reservation.reservation_start_date >  '
$clerkEnteredEndDate '
OR Reservation.reservation_end_date <  '
$clerkEnteredStartDate '
AND (Toolinfo.tool_type LIKE ' $searchKey' OR
Toolinfo.tool_subtype LIKE '$searchKEY ' OR
Toolinfo.tool_suboption LIKE ' $searchKEY ' OR
Tools.power_source LIKE ' $searchKEY ')
AND SaleOrder.for_sale_date = 'NULL'
ORDER BY Tools.tool_id;
```

- If clerk wants to order service, click ***Service Tool*** button; Display Tool ID needs to be serviced.

```
SELECT tool_id
FROM Tools
WHERE Tools.tool_id = ' $toolID ';
```

- Clerk enters *Service Cost*, *Service Start Date*, *Service End Date*, when ready, customer clicks ***Confirm*** button;

```
INSERT INTO ServiceOrder (tool_id, service_start_date,
service_end_date, repair_cost ) VALUES
(' $toolId ', ' $serviceStartDate ', ' $serviceEndDate ', ' $repairCost ');
```

-

Sell Tool

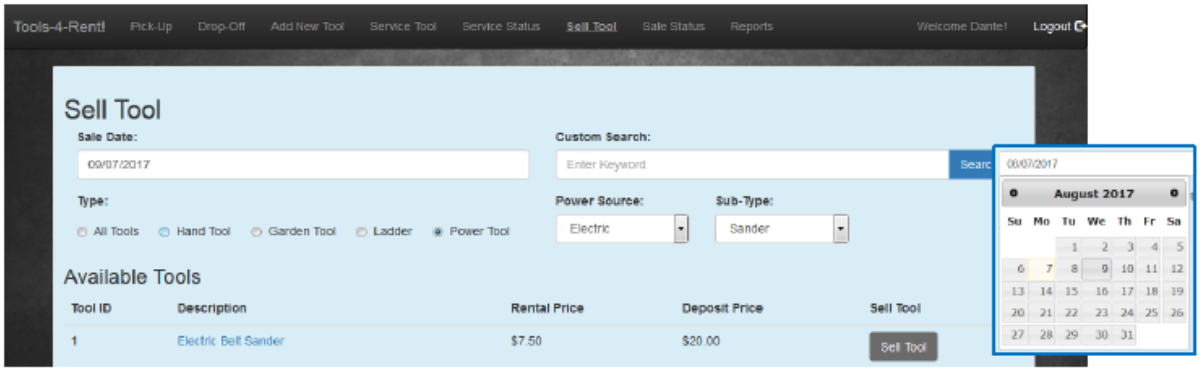


Figure 16 – Sell Tool – with Date Picker and power-source /sub-type dropdown sections

Sell Tool

Abstract Code

- If a tool's rent times is 50 after returned, the tools status is automatically marked as 'for-sale'.

```
INSERT INTO SaleOrder (tool_id, sale_id, for_sale_date) Values ('$tool_id', '$sale_id', '$for_sale_date')
```

- If a tool's rent time is smaller than 50:
  - Clerk User clicks on ***Sell Tool*** button from **Clerk Main Menu:**
  - Run the **Sell Tool** task:
    - Clerk User inputs the *Sale Date*.
    - Clerk User run **Check Tool Availability** task: Clerk User clicks ***Search*** button, find the tools which fulfill the search conditions; Display the Tool ID, Tool Description, Rental Price, Deposit Price.

```
SELECT Tools.tool_id,
CONCAT (IF(Tools.power_source =  ' Manual ',   ' ',
CONCAT(Tools.power_source, ' ' )),
Toolinfo.tool_suboption,  ' ',  Toolinfo.tool_subtype)AS Description,
(Tools.purchase_price * 0.15) AS 'Rental Price',
(Tools.purchase_price * 0.4) AS 'Deposit Price'
FROM Tools
INNER JOIN Toolinfo ON Tools.tool_id = Toolinfo.tool_id
INNER JOIN Reservation ON Tools.tool_id = Reservation.tool_id
LEFT OUTER JOIN ServiceOrder ON Tools.tool_id = ServiceOrder.tool_id
LEFT OUTER JOIN SaleOrder ON Tools.tool_id = SaleOrder.tool_id
WHERE
Tools.power_source =' $customerPowerSource '
AND Toolinfo.tool_type = ' $customerType '
AND Toolinfo.tool_subtype = ' $customerSubType '
AND Toolinfo.tool_suboption = ' $customerSubOption '
AND Reservation.reservation_end_date <  ' $currentTime '
AND (Toolinfo.tool_type LIKE ' $searchKey' OR Toolinfo.tool_subtype LIKE
'$searchKEY ' OR Toolinfo.tool_suboption LIKE ' $searchKEY ' OR
Tools.power_source LIKE ' $searchKEY ')
AND ServiceOrder.service_end_date < '$currentTime '
AND SaleOrder.for_sale_date = 'NULL'
ORDER BY Tools.tool_id;
```

- If clerk clicks the **Tool Description**, detail description of the tool is shown in new window.

- If Manual Tools are selected:

```
SELECT T.tool_id AS "Tool ID", Toolinfo.tool_type AS "Tool Type",
CONCAT(Toolinfo.tool_subtype, Toolinfo.tool_suboption) AS "Short
Description", CONCAT(T.width_or_diameter, ' in.W× ', T.length, ' in.L ',
T.weight, ' lb. ', Toolinfo.tool_suboption, Toolinfo.tool_subtype, ' by ',
T.manufacturer) AS "Full Description", ROUND ((T.purchase_price * .40),
2) AS "Deposit Price", ROUND ((T.purchase_price * .15), 2) AS "Rental
Price" FROM Tools as T INNER JOIN Toolinfo on T.tool_id =
Toolinfo.tool_id WHERE T.power_source = 'manual';
```

- If Power Tools are selected:

```
SELECT T.tool_id AS "Tool ID", Toolinfo.tool_type AS "Tool Type",
CONCAT(T.power_source, Toolinfo.tool_subtype,
Toolinfo.tool_suboption) AS "Short Description",
CONCAT(T.width_or_diameter, ' in.W× ', T.length, ' in.L ', T.weight, ' lb. ',
T.power_source, Toolinfo.tool_suboption, Toolinfo.tool_subtype,
PowerTools.volt_rating, ' Volt ', PowerTools.amp_rating, ' Amp ',
PowerTools.min_rpm_rating, ' RPM ', PowerTools.max_rpm_rating , ' ft_lb
', ' by ', T.manufacturer) AS "Full Description", ROUND ((T.purchase_price
* .40), 2) AS "Deposit Price", ROUND ((T.purchase_price * .15), 2) AS
"Rental Price", CONCAT("(", Power_Accessories.quantity, ")",
PowerTools.volt_rating, ' Volt ', PowerTools.amp_rating, ' Amp ',
Power_Accessories. battery_type, accerssory_description) AS "Accessories"
FROM ((Tools as T INNER JOIN Toolinfo on T.tool_id = Toolinfo.tool_id)
INNER JOIN PowerTools on T.tool_id = PowerTools.tool_id INNER JOIN
Power_Accessories on T.tool_id = Power_Accessories.tool_id) WHERE
T.power_source != "manual";
```

- If Clerk User clicks the ***Sell Tool*** button, mark the tool status as ‘for-sale’ and clerk ID is recorded in database.

```
Insert into SaleOrder (clerk_username, tool_id, for_sale_date, sale_id)
Values  ('$clerk_username', '$tool_id', '$for_sale_date', '$sale_id')
```

View Sale Status

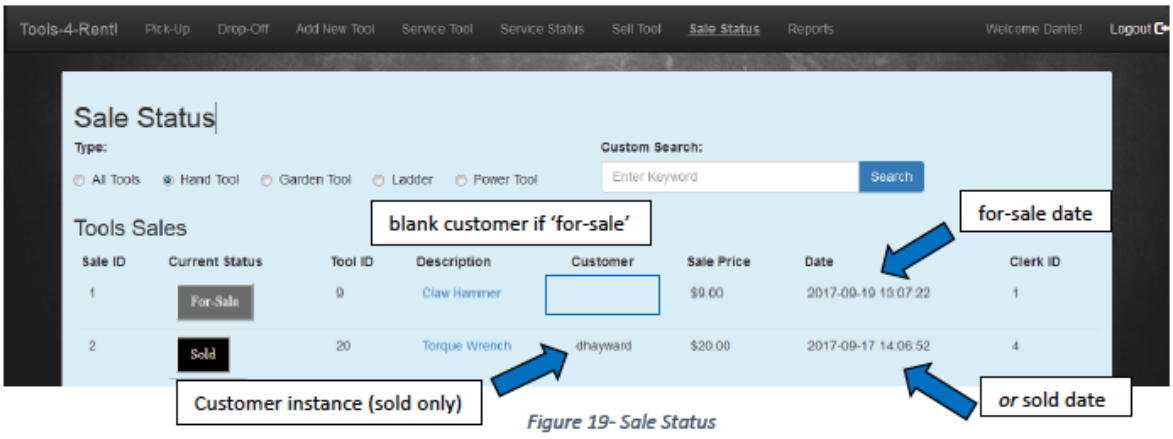


Figure 19- Sale Status

View Sale Status

Abstract Code

- Clerk User clicks on ***Sale Status*** button from **Clerk Main Menu:**
- Run the **View Sale Status** task:
  - Clerk User inputs *Tool Category* (radio buttons) and *keyword tool short-description*.
  - Clerk User clicks ***Search*** button, find the tools was marked as 'for-sale' which fulfill the search conditions; Display the Tool ID, Current Status, Description, Customer, Sale Price, Sale Date and Clerk ID.

```
SELECT Tools.tool_id,
CONCAT (IF(Tools.power_source =  ' Manual ',   ' ',
CONCAT(Tools.power_source, ' ' )),
Toolinfo.tool_suboption,  ' ',  Toolinfo.tool_subtype)AS Description,
(Tools.purchase_price * 0.15) AS 'Rental Price',
(Tools.purchase_price * 0.4) AS 'Deposit Price'
FROM Tools
INNER JOIN Toolinfo ON Tools.tool_id = Toolinfo.tool_id
INNER JOIN Reservation ON Tools.tool_id = Reservation.tool_id
LEFT OUTER JOIN ServiceOrder ON Tools.tool_id = ServiceOrder.tool_id
LEFT OUTER JOIN SaleOrder ON Tools.tool_id = SaleOrder.tool_id
WHERE
Tools.power_source =' $customerPowerSource '
AND Toolinfo.tool_type = ' $customerType '
AND Toolinfo.tool_subtype = ' $customerSubType '
AND Toolinfo.tool_suboption = ' $customerSubOption '
AND (Toolinfo.tool_type LIKE ' $searchKey' OR Toolinfo.tool_subtype LIKE
'$searchKEY ' OR Toolinfo.tool_suboption LIKE ' $searchKEY ' OR
Tools.power_source LIKE ' $searchKEY ')
AND SaleOrder.for_sale_date < '$currentTime '
ORDER BY Tools.tool_id;
```

  - If Clerk User clicks the **Tool Description**, detail description of the tool is shown in a new window.

- If Manual Tools are selected:

```
SELECT T.tool_id AS "Tool ID", Toolinfo.tool_type AS "Tool Type",
CONCAT(Toolinfo.tool_subtype, Toolinfo.tool_suboption) AS "Short
Description", CONCAT(T.width_or_diameter, ' in.W× ', T.length, ' in.L ',
T.weight, ' lb. ', Toolinfo.tool_suboption, Toolinfo.tool_subtype, ' by ',
T.manufacturer) AS "Full Description", ROUND ((T.purchase_price * .40),
2) AS "Deposit Price", ROUND ((T.purchase_price * .15), 2) AS "Rental
Price" FROM Tools as T INNER JOIN Toolinfo on T.tool_id =
Toolinfo.tool_id WHERE T.power_source = 'manual';
```

- If Power Tools are selected:

```
SELECT T.tool_id AS "Tool ID", Toolinfo.tool_type AS "Tool Type",
CONCAT(T.power_source, Toolinfo.tool_subtype, Toolinfo.tool_suboption) AS
"Short Description", CONCAT(T.width_or_diameter, ' in.W× ', T.length, ' in.L ',
T.weight, ' lb. ', T.power_source, Toolinfo.tool_suboption,
Toolinfo.tool_subtype, PowerTools.volt_rating, ' Volt ', PowerTools.amp_rating,
' Amp ', PowerTools.min_rpm_rating, ' RPM ', PowerTools.max_rpm_rating , '
ft_lb ', ' by ', T.manufacturer) AS "Full Description", ROUND ((T.purchase_price
* .40), 2) AS "Deposit Price", ROUND ((T.purchase_price * .15), 2) AS "Rental
Price", CONCAT("(", Accessories.quantity, ")", PowerTools.volt_rating, ' Volt ',
PowerTools.amp_rating, ' Amp ', Cordless. battery_type, accerssory_description)
AS "Accessories" FROM ((Tools as T INNER JOIN Toolinfo on T.tool_id =
Toolinfo.tool_id) INNER JOIN PowerTools on T.tool_id = PowerTools.tool_id
INNER JOIN Accessories on T.tool_id = Accessories.tool_id INNER JOIN
Cordless on T.tool_id = Cordless.tool_id ) WHERE T.power_source != "manual"
```

Generate Reports

Abstract Code

- Clerk clicked on ***Generate Reports*** button from **Main Menu** – jump to the Select a Report menu.
- Click ***Clerk Report button*** button, Jump to **Generate Clerk Report** task
  - Run the **Generate Clerk Report** task: query for information about the clerks and their profile where #ClerkID is the ID of the current clerk using the system from the HTTP Session/Cookie.
  - Find the report lists of the Clerk including employee number, full name, email, total pick-ups, total drop-offs etc.
  - Display the report lists of the above.

```
CREATE TEMPORARY TABLE TempClerk
SELECT * FROM `User` NATURAL JOIN Clerk;

CREATE TEMPORARY TABLE PickUpClerk
SELECT * FROM TempClerk INNER JOIN PickUp ON TempClerk.username =
PickUp.clerk_username;

CREATE TEMPORARY TABLE PickUpCount
SELECT clerk_username, COUNT(*) AS "pick_up_count" FROM PickUpClerk
GROUP BY clerk_username;

CREATE TEMPORARY TABLE DropOffClerk
SELECT * FROM TempClerk INNER JOIN DropOff ON TempClerk.username =
DropOff.clerk_username;

CREATE TEMPORARY TABLE DropOffCOunt
SELECT clerk_username, COUNT(*) AS "drop_off_count" FROM DropOffClerk
GROUP BY clerk_username;

CREATE TEMPORARY TABLE ClerkCount
SELECT * FROM PickUpCount NATURAL JOIN DropOffCount;

SELECT clerk_id AS "Clerk ID", first_name AS "First Name", middle_name AS
"Middle Name", last_name AS "Last Name", email AS "Email", date_of_hire AS
"Hire Date", pick_up_count AS "Number of Pickups", drop_off_count AS "Number
of Dropoffs", pick_up_count+drop_off_count AS "Combined Total"
FROM TempClerk NATURAL JOIN ClerkCount;
```

- Click ***Customer Report*** button, Jump to **Generate Customer Report** task
  - Run the **Generate Customer Report** task: query for information about the customers and their profile where #CustomerID is the ID of the current customer using the system from the HTTP Session/Cookie.
  - If Clerk clicks the ***View Profile***, run the **View Profile** task.
  - Find the report lists of the Customers including Customer ID, First Name, Middle Name, Last Name, total reservations etc.
  - Display the report lists of the above.

```
CREATE TEMPORARY TABLE Customer_User
SELECT * FROM `User` NATURAL JOIN Customer;

CREATE TEMPORARY TABLE num_of_reservation
SELECT username, COUNT(*) AS "Total_Reservations" FROM Customer
NATURAL JOIN Reservation
GROUP BY username;

CREATE TEMPORARY TABLE num_of_rent
SELECT username, COUNT(*) AS "Total_Tools_Rented" FROM Customer
NATURAL JOIN Rented
GROUP BY username;

CREATE TEMPORARY TABLE Customer_stat
SELECT * FROM num_of_Reservation NATURAL JOIN num_of_rent;

SELECT customer_id AS "Customer ID", first_name AS "First Name",
middle_name AS "Middle Name", last_name AS "Last Name", email AS "Email",
Total_Reservations AS "Total # Reservations", Total_Tools_Rented AS "Total #
Tools Rented"
FROM Customer_User NATURAL JOIN Customer_stat;
```

- Click ***Tool Inventory Report*** button, Jump to **Tool Inventory Report** task.
  - Run the **Generate tool Report** task: query for information about the tools and their profile where #ToolID is the ID of the current tools using the system from the HTTP Session/Cookie.
  - Find the report lists of the tools including tools ID, Current Status, Date, Description, total cost, total profit and etc.
  - Display the report lists of the above.

```
CREATE TEMPORARY TABLE ToolFullInfo
SELECT *
FROM Tools NATURAL JOIN `With` NATURAL JOIN Rented NATURAL
JOIN Reservation NATURAL JOIN ServiceOrder NATURAL JOIN SaleOrder
NATURAL JOIN ToolInfo;

SELECT tool_id AS "Tool ID", for_sale_date, sold_date, service_start_date,
service_end_date, reservation_start_date, reservation_end_date, rented_start_date,
rented_end_date, ROUND(SUM(purchase_price * times_rented * 0.15), 2) AS
"Rental Profit", ROUND(SUM(repair_cost), 2) AS "Total Cost", "Rental Profit" -
"Total Cost" AS "Total Profit"
FROM ToolFullInfo
GROUP BY tool_id;
```

For the Current Status:
If sold_date is not NULL, then Sold;
Else If for_sale_date is not NULL, then For-Sale;
Else If CurrentDate in the range of (service_start_date, service_end_date), then In-Repair;
Else If CurrentDate in the range of (rented_start_date, rented_end_date), then Rented;
Else If CurrentDate not in the range of (reservation_start_date, reservation_end_date), then Available.