LOGIN

Abstract Code:

- *If user* selects RentalCustomer**/**Clerk radio button and enters *username* and *password* into input fields, then:
    - If data validation is successful for both username and password input fields, then:
        - When ***Sign-in*** button is clicked:
            - If User record is found but password is incorrect:

            > SELECT password, type FROM `User` WHERE username= '$username ';

            - If User record is found but User.password! = '$Password' or User.type! = '$type'
                - Go back to **Login** form, with error message
            - Else:
            - Go to **Main Menu** form of user (RentalCustomer /clerk)
    - Else *username* and *password* input fields are invalid, display **Login form,** with error message
- Else, with error message, there is no Clerk/RentalCustomer with username

REGISTRATION:

Abstract Code:
- Populate all the input fields

    > '$Value' = SELECT Count(*) FROM User WHERE username = '$Username';

- Step 1: If username does not exists, then:

    > If '$Value' = 0 then;

    - Select at leastid
    - one phone number (either home, work or cell) as primary phone number
    - Select state from the dropdown
    - Push the **Register** button
- Else: populate another username and go to step 1

View Profile:

Abstract Code:
- RentalCustomer clicked on **View profile** button
- Find the current User using the User.username; Display username, full name, available phone numbers and address

    > SELECT User.username, email, CONCAT(first_name, " " ,middle_name, " " ,last_name) as full_name, CONCAT(street, " " ,city, " " ,state, " " ,zip) AS full_address, cpn.*
    > FROM User
    > INNER JOIN RentalCustomer AS rc ON User.username=rc.username
    > INNER JOIN CustomerPhoneNumber AS cpn ON cpn.rcID=rc.rcID

WHERE User.username = '$Username';
- Display reservations of RentalCustomer
- Read each reservation of the RentalCustomer ordered from most recent to oldest; Display names of clerk, tool details, date details and price details

```
SELECT r.rNO, r.start_date, r.end_date, r.pickupClerkENO, r.dropoffClerkENO,
r.deposit_price, r.rental_price, CONCAT(t.power_source, " ", t.sub_option, " ", t.sub_type)
AS Tool
FROM Holds as h
INNER JOIN Reservation as r
ON h.rNO = r.rNO
INNER JOIN Tool as t
ON h.ToolID = t.ToolID
WHERE r.rcID = "$rcID";
```

Check Tool Availability

- When the **Check tool availability** button is pushed on the main menu
- Populate input fields in keyword search section (*start/end dates, tool category, power_source/sub_types*)
- Step 1: push **Search** button
- If number of tools returned is less than 10, then:
  - Find the input fields and display available inventory information (tool number, a short-description, deposit price per reservation, rental price per day)

```
SELECT toolID,  CONCAT(power_source,sub_option,sub_type) AS short_desc,
r.rental_price, r.deposit_price FROM Tool
MINUS
SELECT t.toolID
FROM Holds as h
INNER JOIN Reservation AS r
ON h.rNO = r.rNO
INNER JOIN Tool AS t
ON  h.ToolID = t.ToolID
WHERE
('$Start_date' >= r.start_date AND '$Start_date' <= r.end_date)
OR
('$end-date' >= r.start_date AND '$end-date' <= r.end_date)
OR
('$Start-date' <= r.start_date AND '$end-date' >= r.end_date)
AND Tool.power_source = '$power-source' and Tool.sub_type = '$sub-type' and
Tool.tool_type = '$tool-type'
AND
(t.sub_option= '$sub_option' OR '$sub_option' IS NULL)
AND
(t.rental_price > '$low_repaircost' OR '$low_rental_price' IS NULL)
```

```
AND
(t.rental_price < '$high_repaircost' OR '$high_rental_price' IS NULL)
AND
(t.deposit_price > '$low_deposit_price' OR '$low_deposit_price' IS NULL)
AND
(t.deposit_price < '$high_deposit_price' OR '$high_deposit_price' IS NULL);
```

- If customer wishes to see further details on a specific tool, clicks on description column
  - Find the tool and display the description of the tool (Tool id, Tool type, Short description, Full description, Deposit price and rental price)

```
SELECT Tool.ToolID, category, CONCAT(power_source,sub_option,sub_type) AS
short_desc,
CASE
WHEN category="Hand" THEN CONCAT(length,"in. Wx",width," L",weight," lb.",
category," ",sub_option," ",sub_type," ", "#", screw_size, drive_size, " in.",
IF(adjustable, "adjustable", ""), " ", gauge_rating, "G ", capacity, " ", manufacturer)
  WHEN category="Garden" THEN CONCAT(length,"in. Wx",width," L",weight," lb.",
category," ",sub_option," ",sub_type," ",handle_material," ",blade_length," in.
",head_weight," lb. ",blade_width," in. ",IF(blade_length IS NOT NULL,
CONCAT(blade_length, "in. "), ""),tine_count," tine ",bin_material," ",wheel_count,"
",manufacturer)
WHEN category="Power" THEN CONCAT(length,"in. Wx",width," L",weight," lb.",
category," ",sub_option," ",sub_type,IF(volt_rating IS NOT NULL,
CONCAT(volt_rating,"V "), ""),IF(amp_rating IS NOT NULL, CONCAT(amp_rating, "A "),
""),IF(min_torque_rating IS NOT NULL, CONCAT(min_torque_rating, "ft-lb "),
""),IF(blade_size IS NOT NULL, CONCAT(blade_size, "in. "), ""),IF(tank_size IS NOT
NULL, CONCAT(tank_size, "Gal "), ""),IF(motor_rating IS NOT NULL,
CONCAT(motor_rating, "hp "), ""),IF(power_rating IS NOT NULL,
CONCAT(power_rating, "W "), ""),manufacturer)
WHEN category="Ladder" THEN CONCAT(length,"in. Wx",width," L",weight," lb.",
category," ",sub_option," ",sub_type," ",IF(step_count IS NOT NULL,
CONCAT(step_count," step "), ""),IF(weight_capacity IS NOT NULL,
CONCAT(weight_capacity," lb.capacity "), ""),manufacturer)
END AS full_desc, rental_price, deposit_price
FROM Tool
INNER JOIN HandTool AS ht
INNER JOIN LadderTool AS lt
INNER JOIN PowerTool AS pt
INNER JOIN GardenTool AS gt
WHERE Tool.ToolID = '$ToolID';
```

Make Reservation

Abstract Code:
- When the **Make Reservation** button is pushed on the **Main menu**

- Step 1: Populate input fields in keyword search section (*start/end dates, tool category, power_source/sub_types, type*)
- Push **Search** button
- Find the input fields and display available inventory information (tool number, a short-description, deposit price per reservation, rental price per day)

```sql
SELECT toolID,  CONCAT(power_source,sub_option,sub_type) AS short_desc,
r.rental_price, r.deposit_price FROM Tool
MINUS
SELECT t.toolID
FROM Holds as h
INNER JOIN Reservation AS r
ON h.rNO = r.rNO
INNER JOIN Tool AS t
ON  h.ToolID = t.ToolID
WHERE
(('$Start_date' >= r.start_date AND '$Start_date' <= r.end_date)
OR
('$end-date' >= r.start_date AND '$end-date' <= r.end_date)
OR
('$Start-date' <= r.start_date AND '$end-date' >= r.end_date))
AND Tool.power_source = '$power-source'
AND Tool.sub_type = '$sub-type'
AND Tool.tool_type = '$tool-type'
AND
(t.sub_option= '$sub_option' OR '$sub_option' IS NULL)
AND
(t.deposit_price > '$low_deposit_price' OR '$low_deposit_price' IS NULL)
AND
(t.deposit_price < '$high_deposit_price' OR '$high_deposit_price' IS NULL);
```

- Push the **Add** button on all the required tools
- If number of added tool is 11th, then:
  - An error message is displayed
- Else:
  - Go to next step
- If any of the added tool available inventory reached zero, then
  - As error message is displayed
  - If identical tool requested is due to return within the next 24 hours, then
    - A notice is given to the user telling them the date when the tool is expected to become available
  - Else: go to next step
  - When customer is done with adding all the required tools

- Push the **Calculate Total** button is pushed
- A reservation summary form is popped displaying total deposit price, rental price, number of days
- If we wished to complete the reservation, then
  - Push the **Submit** button.
  - If any of the reserving tools is already reserved by another customer, then:
    - Display notice of unavailability and go to step 1
  - Else: go to next step
  - **Reservation confirmation** form pops-up displaying total deposit price, rental price, number of days

> **Reservation Confirmation**
>
> SELECT
> rNO, rental_price, deposit_price,
> FROM Reservation as r
> INNER JOIN RentalCustomer as rc ON rc.rcID = r.rcID
> INNER JOIN User ON User.username = rc.username
> WHERE r.rNO = '$ReservationNumber';
>
> **Tools under reservation confirmation**
>
> SELECT
> t.toolID, CONCAT(t.power_source, t.sub_type, t.sub_option) AS Description,
> t.deposit_price, t.rental_price
> FROM Holds as h
> INNER JOIN Reservation as r
> ON h.rNO = r.rNO
> INNER JOIN Tool as t
> ON h.ToolID = t.ToolID)
> WHERE r.rNO = '$ReservationNumber';

- **Else:** Push the **Reset** button and go to step 1

Purchase Tool

Abstract Code:
- When the **Purchase Tool** button is pushed on the main menu
- Step 1: Populate input fields in keyword search section (*start/end dates, tool category, power_source/sub_types, type*)
- Push **Search** button
- Find available tools for sale and display tool description and sale price

```
SELECT
toolID, status , CONCAT(power_source, sub_type, sub_option), sale_price
FROM Tool INNER JOIN SalesOrder ON Tool.toolID = SalesOrder.toolID
WHERE Tool.status = '$for-sale'
AND
Tool.power_source = $power_source'
AND
Tool.sub_type = '$sub_type' and Tool.category = '$category'
AND
(Tool.sub_option= '$sub_option' OR '$sub_option' IS NULL);
```

- Select the tools from the list and push the **Submit** button
- If the credit file is not on file, then
  - Customer is required to enter the information
- Else:
  - Purchase confirmation form pops-up displaying total sale price and sold-date

---

Purchase Confirmation

```
SELECT
r.saleID, r.sale_price
FROM SalesOrder as r
INNER JOIN RentalCustomer as rc ON rc.rcID = r.rcID
INNER JOIN User ON User.username = rc.username
WHERE r.saleID = '$Sales_ID';
```

Tools under Purchase confirmation

```
SELECT
r.saleID, CONCAT( t.power_source, t.sub_type, t.sub_option) AS Description, r.sale_price
FROM SalesOrder as r
INNER JOIN Tool as t
ON t.toolID = r.toolID )
WHERE r.saleID =  '$Sales_ID';
```

PICK-UP Reservation:

Abstract Code:
- When the **Pick-up reservation** button is pushed on the main menu
- Find all the reservations which are waiting to be picked up

```
SELECT rNO, CONCAT(first_name, " " ,middle_name, " " ,last_name) as full_name, rcID,
start_date, end_Date
FROM Reservation as r
INNER JOIN RentalCustomer as rc ON rc.RCID = r.RCID
INNER JOIN User ON User.username = rc.username
WHERE  r.pickupClerkENO IS NULL;
```

- Select a reservation either by entering the specific reservation number and pushing **Pick-up** button or click a link
- Display summary of the reservation
- If the CreditCard information does not exist in the customer's profile, then:

```
If the '$value' = 0 (CreditCard is not on file), then:
'$value' = SELECT count(*) FROM RentalCustomer WHERE
RentalCustomer.username = CreditCard.username ;
```

  - Step 1: Clerk should enter the customer's CreditCard information (*number, name on card, expiration month as a drop down, expiration year as a drop down, cvc 3 digit number*)
- Else:
  - Leave as it is
- Push the **confirm Pick-up** button
- If the information is valid, then:
  - Display rental contract with summary for the transaction

Fetching Customer Name, start and end dates

```
SELECT CONCAT(first_name, " " ,middle_name, " " ,last_name) as Customer_Name,
start_date, end_Date
FROM Reservation as r
INNER JOIN RentalCustomer as rc ON rc.RCID = r.RCID
INNER JOIN User as u ON User.username = rc.username
WHERE Reservation.rNO = '$ReservationNUmber';
```

Fetching Clerk Name

```
SELECT CONCAT(first_name, " " ,middle_name, " " ,last_name) as Pick-up_Clerk,
FROM Reservation as r
INNER JOIN Clerk as c ON c.empNO = r.pickupClerkENO
INNER JOIN User as u ON User.username = c.username
WHERE Reservation.rNO = '$ReservationNUmber';
```

Fetching Tools information

```
SELECT
t.toolID, CONCAT( t.power_source, t.sub_type, t.sub_option) AS Tool_Name,
t.deposit_price, t.rental_price
FROM Holds as h
INNER JOIN Reservation as r
ON h.rNO = r.rNO
INNER JOIN Tool as t
ON h.toolID = t.toolID )
WHERE Reservation.rNO = '$ReservationNUmber'
```

- Else:
  - Go to step 1

Drop-Off Reservation

Abstract Code:
- If user **Drop-off reservation** button is pushed on the main menu, then:
  - Find all the reservations which are waiting to be dropped-off

```
SELECT rNO, CONCAT(first_name, " " ,middle_name, " " ,last_name) as Customer_Name,
rcID, start_date, end_date
FROM Reservation as r
INNER JOIN RentalCustomer as rc ON rc.rcID = r.rcID
INNER JOIN User ON User.username = rc.username
WHERE r.end_date <= Today();
```

  - Select a reservation either by entering the specific reservation number by:
    - By pushing **Drop off** button or
    - by clicking a link
  - Final receipt pops-up up (user can optionally print for their own records)

    ### Reservation Details

```
SELECT
rNO, CONCAT(first_name, " " ,middle_name, " " ,last_name) as Customer_Name,
rental_price, deposit_price,
(rental_price  – deposit_price) AS TotalDue
FROM Reservation as r
INNER JOIN RentalCustomer as rc ON rc.rcID = r.rcID
INNER JOIN User ON User.username = rc.username
WHERE r.rNO = '$ReservationNumber';
```

    ### Tool Details

```
SELECT
t.toolID, CONCAT( t.power_source, t.sub_type, t.sub_option) AS Tool_Name,
t.deposit_price, t.rental_price
FROM Holds as h
INNER JOIN Reservation as r
ON h.rNO = r.rNO
INNER JOIN Tool as t
ON  h.toolID = t.toolID )
WHERE r.rNO = '$ReservationNumber';
```

- Else: remain as it is

Add Tool

Abstract Code:
- When the **Add tool** button is pushed on the main menu, **Add tool form** is displayed
- If any of the categories entered first, then
    - If selected category is hand tool, garden tool or ladder, then:
        - If next entered is sub_type, then
            - Populate suboption
            - Enable sub_optiondropdown
        - Else: Error message displaying sub_type should be entered first
    - Else: **Power tool form** pops-up
    - if next entered value is power source
        - if subtype is cordless tools only, then:
            - **Cordless Tools only** form pops-up
            - Populate the fields
        - Else: populate other fields in the **Power Tools Only** form
    - Else: Error message displaying power source should be entered first
- Else: Error message displaying category should be entered first

Repair-Tool

Abstract Code:
- When the **Repair tool** button is pushed on the main menu, **Repair Tool form** is displayed
- Populate input fields in keyword search section (*start/end dates, tool category, power_source/sub_types, type*)
- Read all the tools which are "In-repair" status with key word search

```
SELECT toolID,  CONCAT(power_source,sub_option,sub_type) AS short_desc,
r.rental_price, r.deposit_price FROM Tool
MINUS
SELECT t.toolID
FROM Holds as h
INNER JOIN Reservation AS r
ON h.rNO = r.rNO
INNER JOIN Tool AS t
ON  h.ToolID = t.ToolID
WHERE
(('$Start_date' >= r.start_date AND '$Start_date' <= r.end_date)
OR
('$end-date' >= r.start_date AND '$end-date' <= r.end_date)
OR
('$Start-date' <= r.start_date AND '$end-date' >= r.end_date))
AND Tool.power_source = '$power-source'
AND Tool.sub_type = '$sub-type'
AND Tool.tool_type = '$tool-type'
AND
(t.sub_option= '$sub_option' OR '$sub_option' IS NULL)
AND
(t.rental_price > '$low_repaircost' OR '$low_rental_price' IS NULL)
AND
(t.rental_price < '$high_repaircost' OR '$high_rental_price' IS NULL)
AND
(t.deposit_price > '$low_deposit_price' OR '$low_deposit_price' IS NULL)
AND
(t.deposit_price < '$high_deposit_price' OR '$high_deposit_price' IS NULL) AND
Tool.Status IS NULL;
```

- Enter Tool ID number and service cost, push **confirm** button

View Service Status page

Abstract Code:
- When the **View repair status page** button is pushed on the main menu, <u>**view repair status page form**</u> is displayed
- Populate input fields in keyword search section (*start/end dates, tool category, power_source/sub_types, type*)
- Read all the tools which are "In-repair" status with key word search

```
SELECT serviceId, status, toolID, CONCAT(power_source, sub_type, sub_option), start_date,
end_date, repair_cost, CONCAT(first_name, " " ,middle_name, " " ,last_name) as
clerk_name
FROM ServiceOrder AS s
INNER JOIN Tool ON Tool.toolID = s.toolID
INNER JOIN Clerk ON Clerk.empNO = s.empNO
INNER JOIN User ON User.username = Clerk.username
WHERE Tool.status = '$in-repair'
AND
Tool.sub_option= '$sub_option'
AND
(s.repair_cost > '$low_repairprice' OR '$low_repairprice' IS NULL)
AND
(s.repair_cost < '$high_repairprice' OR '$high_repairprice' IS NULL)
```

- If clerk wishes to restrict results of the tool, then:
  - Use filter option on any combination of input (clerk names, sort by dates, repair cost, keyword description etc.)
  - Read all the tools which are "In-repair" status with key word search and restrictions applied using filters
  - Go to step 1
  - Else: go to step 1
- If clerk wishes to sort any of the field, then:
  - Clerk selects the required field (service/tool ID, short-description, repair start date, current repair end date, repair cost)
  - Go to step 1
  - Else: go to step 1
- Step 1: if clerk wants to override the "in-repair" status for any of the displayed tools, then:
  - Push on the **fix now** button
  - Repair end date of the tool changes to now() and returns back to available inventory and drops off the **service status page**
  - Else: Remain as it is

Sell Tool

Abstract Code:
- When the **Sell tool** button is pushed on the main menu, **sell tool page form** is displayed
- Populate input fields in keyword search section (*start/end dates, tool category, power_source/sub_types, type*)
- Find all the tools displaying description, rental and tool price with key word search

```
SELECT toolID,  CONCAT(power_source,sub_option,sub_type) AS short_desc,
r.rental_price, r.deposit_price FROM Tool
MINUS
SELECT t.toolID
FROM Holds as h
INNER JOIN Reservation AS r
ON h.rNO = r.rNO
INNER JOIN Tool AS t
ON  h.toolID = t.toolID
WHERE
(('$Start_date' >= r.start_date AND '$Start_date' <= r.end_date)
OR
('$end-date' >= r.start_date AND '$end-date' <= r.end_date)
OR
('$Start-date' <= r.start_date AND '$end-date' >= r.end_date))
AND Tool.power_source = '$power-source'
AND Tool.sub_type = '$sub-type'
AND Tool.tool_type = '$tool-type'
AND
(t.sub_option= '$sub_option' OR '$sub_option' IS NULL) AND Tool.status IS NULL
AND
(t.deposit_price > '$low_deposit_price' OR '$low_deposit_price' IS NULL)
AND
(t.deposit_price < '$high_deposit_price' OR '$high_deposit_price' IS NULL);
```

- If clerk wishes to restrict results of the tool, then:
    - Use filter option on any combination of input (rental price, deposit price, etc.)
    - Read all the tools which are available with key word search and restrictions applied using filters
    - Go to step 1
- Step 1: if clerk wants to sell tool for any of the displayed tools, then:
    - Push on the **Sell tool** button
    - Unique sale transaction number is created which tracks the tool number, customer who bought the tool, clerk who marked tool 'for sale', sale price, for sale date, and sale date
    - Else: Remain as it is

View Sale Status

Abstract Code:
- When the **View sale status page** button is pushed on the main menu, **view sale status page form** is displayed
- Populate input fields in keyword search section (*start/end dates, tool category, power_source/sub_types, type*)
- Find all the tools which are under for 'For-sale' and 'Sold' status
- If clerk wishes to restrict results of the tool, then:

- Use filter option on any combination of input (clerk names, sort by dates, repair cost, keyword description etc.)
- Else: Remain as it is
- Display the current status of each tool with tool id, description, sale price, sale date and clerk id

```
SELECT saleID, status, toolID, CONCAT(power_source, sub_type, sub_option) as Description,
sale_price, sale_date, empNO FROM SalesOrder
INNER JOIN Tool ON SalesOrder.toolID = Tool.toolID
WHERE Tool.sub_option= '$sub_option'
AND
(SalesOrder.price > '$low_price' OR '$low_price' IS NULL)
AND
(SalesOrder.price < '$high_price' OR '$high_price' IS NULL)
```

- If the current status of the tool is sold, then:
  - The tool is marked as sold in black back ground
  - Display customer name
- Else: The tool is marked as "For sale" in grey back ground

Generate Reports

Abstract Code:
- When the **Generate Reports** button is pushed on the main menu, <u>**Generate reports**</u> form is displayed
- Display a page with clerk report, customer report and tool inventory report buttons

Generate Clerk Reports

Abstract Code:
- When the *clerk reports* button is pushed on the generate reports form, <u>**Clerk report**</u> form is displayed
- Find clerk information
- Display name, username, hire date and number of the clerk
- Find pickups and drop-offs for the clerk
- Display pickup, drop-off and their sum in the report

```
PickUpTable = SELECT pickupClerkENO, COUNT(*)  AS no_of_pickups FROM Reservation
WHERE MONTH(start_date) = MONTH(CURRENT_DATE() - INTERVAL 1 MONTH)
GROUP BY pickupClerkENO

DropOffTable = SELECT dropoffClerkENO, COUNT(*)  AS no_of_dropoffs FROM Reservation
WHERE MONTH(start_date) = MONTH(CURRENT_DATE() - INTERVAL 1 MONTH)
GROUP BY dropoffClerkENO

AllClerksTable = SELECT empNO, first_name, middle_name, last_name, username, hire_date
FROM Clerk INNER JOIN User ON Clerk.username = User.username

SELECT
empNO, first_name, middle_name, last_name, username, hire_date, ISNULL(no_of_pickups, 0)
AS no_pickups, ISNULL(no_of_dropoffs,0) AS no_dropoffs, (no_pickups + no_dropoffs) as
combined_total
FROM AllClerksTable AS all
FULL OUTER JOIN PickUpTable as pk ON all.empNO = pk.pickupClerkENO
FULL OUTER JOIN DropOffTable as df ON all.empNO = df. dropoffClerkENO
ORDER BY no_pickups DESC, no_dropoffs DESC;
```

Generate Customer Report

Abstract Code:

- When the ***customer reports*** button is pushed on the generate reports form, **<u>Customer report</u>** form is displayed
- Find customer information
- Display name, username, phone, profile link and number of the customer
- Find reservations made by customer
- Display reservation number and computed tool number

```
TotalReservTable = SELECT rcID, COUNT(*) AS total_reservations FROM Reservation GROUP BY
rcID;
ToolsPerReservTable = SELECT rNO, COUNT(*) AS tools_per_reserv FROM Holds GROUP BY
rNO;
MergeReservTools = SELECT rcID, SUM(tools_per_reserv) AS total_tools_rented FROM
Reservation AS r
INNER JOIN ToolsPerReservTable AS tp ON r.rNO = tp.rNO
GROUP BY rcID

SELECT rcID, first_name, middle_name, last_name, username, phone_number, total_reserv,
total_tools_rented
FROM RentalCustomer AS rc
INNER JOIN User ON User.username = rc.username
INNER JOIN CustomerPhoneNo AS cp ON cp.rcID = rc.rcID
INNER JOIN MergeReservTable AS mr ON mr.rcID = rc.rcID
WHERE MONTH(start_date) = MONTH(CURRENT_DATE())
ORDER BY total_tool_rented DESC, last_name;
```

Generate Tool Inventory

Abstract Code:

- When the **tool inventory report** button is pushed on the generate reports form, **tool inventory form** is displayed
- Populate the tool type options and display custom search option
- Find the information from tools table
- Display current status, description, rental profit, total cost and total profit

Calculate total costs

```
TotalCosts =
SELECT t.toolID, t.purchase_price, SUM(repair_cost) AS total_repair_cost, (t.purchase_price +
total_repair_cost) AS total_cost
FROM Tool AS t
INNER JOIN ServiceOrder AS sor
ON t.toolId = sor.toolID
GROUP BY t.toolID;
```

Get tool report from TotalCosts and Tool

```
SELECT t.toolID, ISNULL(t.status,"Available") AS status, t.statusdate, CONCAT(power_source,
sub_type, sub_option) as Description, (rental_count * rental_price) AS rental_profit,
total_cost, (rental_profit – total_cost) AS total_profit
FROM Tool AS t
INNER JOIN TotalCosts AS tc
ON t.toolId = tc.toolId
ORDER BY total_profit DESC;
```