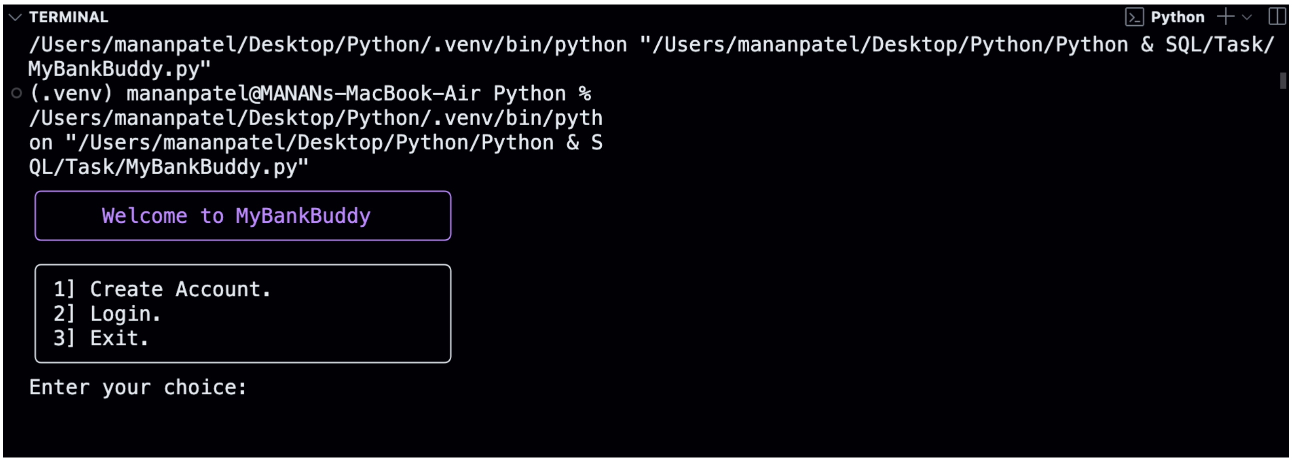
MyBankBuddy

Welcome to **mybankbuddy**, a simple yet efficient banking system built using Python. This mini-project offers a seamless way to manage your account, track transactions, and monitor balances. With its user-friendly interface and secure features, mybankbuddy ensures a smooth banking experience. Whether you're making deposits, withdrawals, or checking your balance, this project showcases essential concepts of Python and database management in a practical, interactive way.

1. Welcome window:

The image shows a Python terminal window displaying the welcome screen for a bank account management program called "MyBankBuddy". It offers three options: 1) Create Account, 2) Login, and 3) Exit.

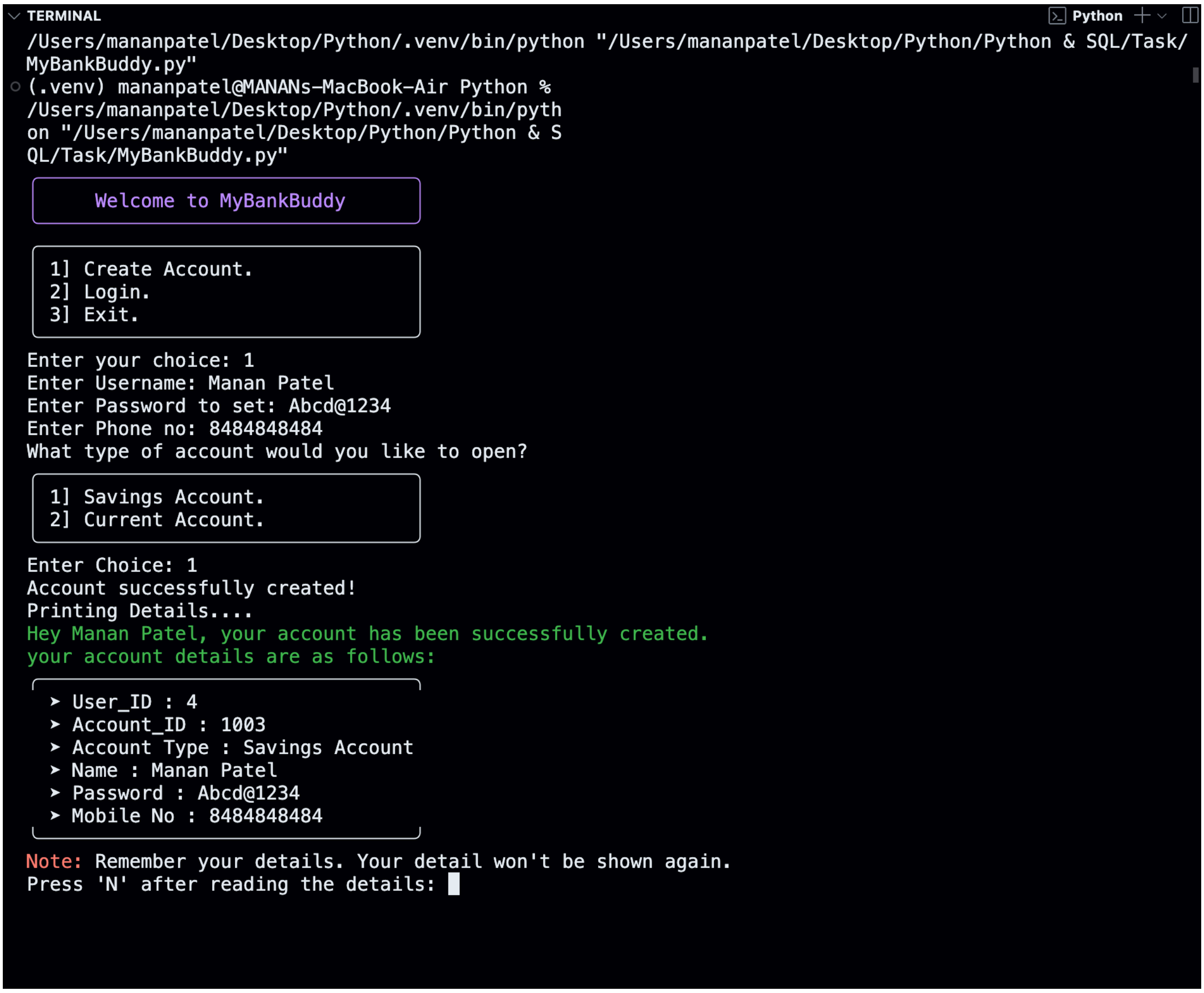


```
/Users/mananpatel/Desktop/Python/.venv/bin/python "/Users/mananpatel/Desktop/Python/Python & SQL/Task/
MyBankBuddy.py"
(.venv) mananpatel@MANANs-MacBook-Air Python %
/Users/mananpatel/Desktop/Python/.venv/bin/pyth
on "/Users/mananpatel/Desktop/Python/Python & S
QL/Task/MyBankBuddy.py"
┌──────────────────────────────────┐
│     Welcome to MyBankBuddy       │
└──────────────────────────────────┘

┌──────────────────────────────────┐
│ 1] Create Account.               │
│ 2] Login.                        │
│ 3] Exit.                         │
└──────────────────────────────────┘

Enter your choice:
```

2. Signup:

The user has selected option 1, "Create Account". The program then prompts the user for their username, password, and phone number. After entering these details, the user is asked to choose between creating a Savings Account or a Current Account. The user selects "Savings Account" and successfully creates the account. And it store all data in MYSQL database.

Finally, the program displays the user's account details, including the User ID, Account ID, Account Type, Name, Password, and Mobile Number.

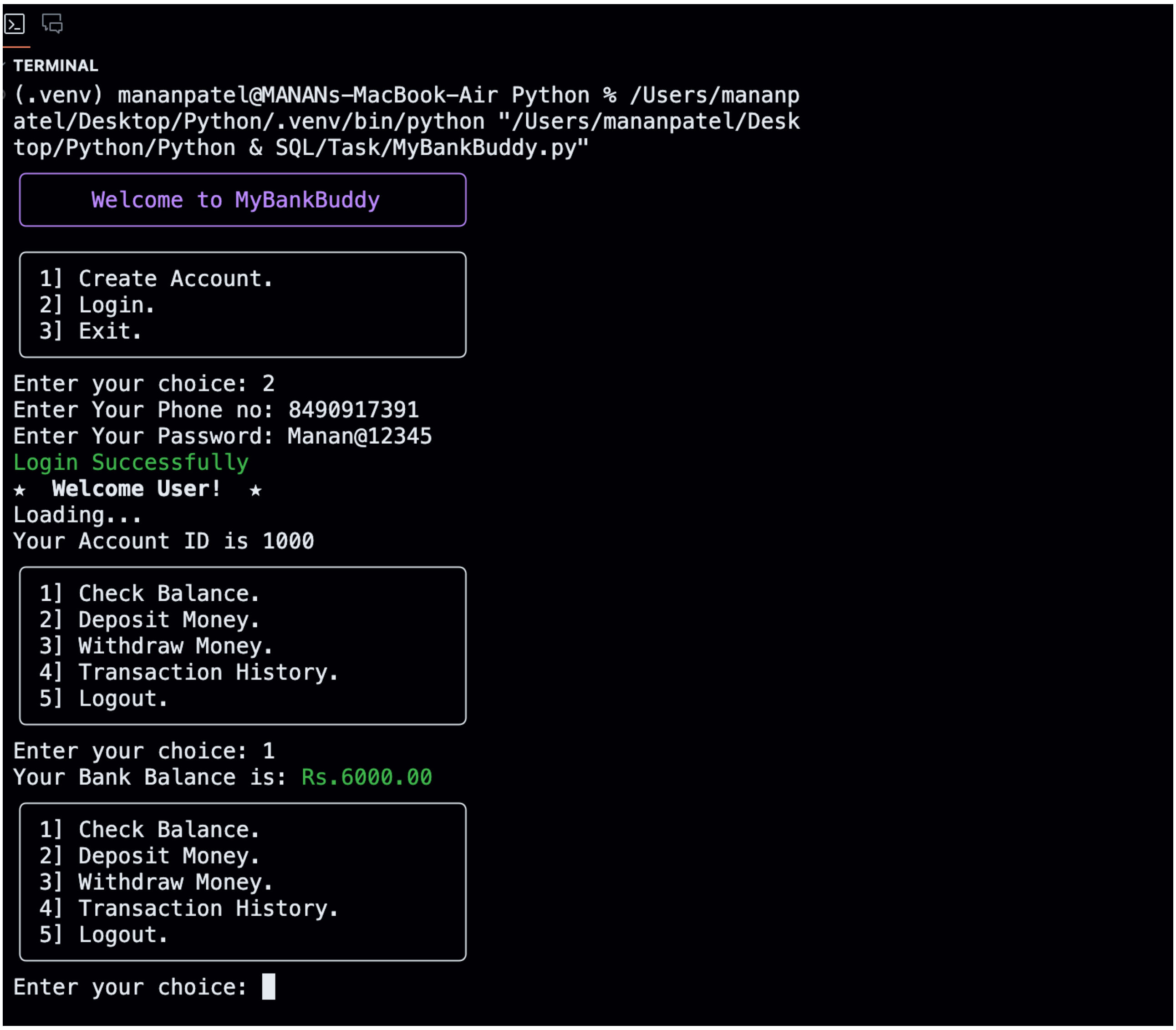


```
/Users/mananpatel/Desktop/Python/.venv/bin/python "/Users/mananpatel/Desktop/Python/Python & SQL/Task/
MyBankBuddy.py"
(.venv) mananpatel@MANANs-MacBook-Air Python %
/Users/mananpatel/Desktop/Python/.venv/bin/pyth
on "/Users/mananpatel/Desktop/Python/Python & S
QL/Task/MyBankBuddy.py"
 ╭──────────────────────────────────╮
 │      Welcome to MyBankBuddy      │
 ╰──────────────────────────────────╯
 ╭──────────────────────────────────╮
 │ 1] Create Account.               │
 │ 2] Login.                        │
 │ 3] Exit.                         │
 ╰──────────────────────────────────╯
Enter your choice: 1
Enter Username: Manan Patel
Enter Password to set: Abcd@1234
Enter Phone no: 8484848484
What type of account would you like to open?
 ╭──────────────────────────────────╮
 │ 1] Savings Account.              │
 │ 2] Current Account.              │
 ╰──────────────────────────────────╯
Enter Choice: 1
Account successfully created!
Printing Details....
Hey Manan Patel, your account has been successfully created.
your account details are as follows:
 ╭──────────────────────────────────╮
   ➤ User_ID : 4
   ➤ Account_ID : 1003
   ➤ Account Type : Savings Account
   ➤ Name : Manan Patel
   ➤ Password : Abcd@1234
   ➤ Mobile No : 8484848484
 ╰──────────────────────────────────╯
Note: Remember your details. Your detail won't be shown again.
Press 'N' after reading the details: █
```

3. Login:

When the user selects option 2 to log in, the program prompts them to enter their phone number and password. If both credentials are correct, the program will display a message saying, "Login successful! Welcome, User!" and show the user's account ID. Following that, the program will present the login interface, offering options like checking the balance, depositing money, withdrawing money, viewing transaction history, or logging out.

If the user selects option 1 to check their balance, the program will display their current bank balance. If the user chooses option 2 to deposit money, they will be prompted to enter the amount to deposit. After they enter an amount, the program will confirm the transaction, displaying a message such as, "Rs. 100 credited to A/C No. 1000," followed by an update of their bank balance.

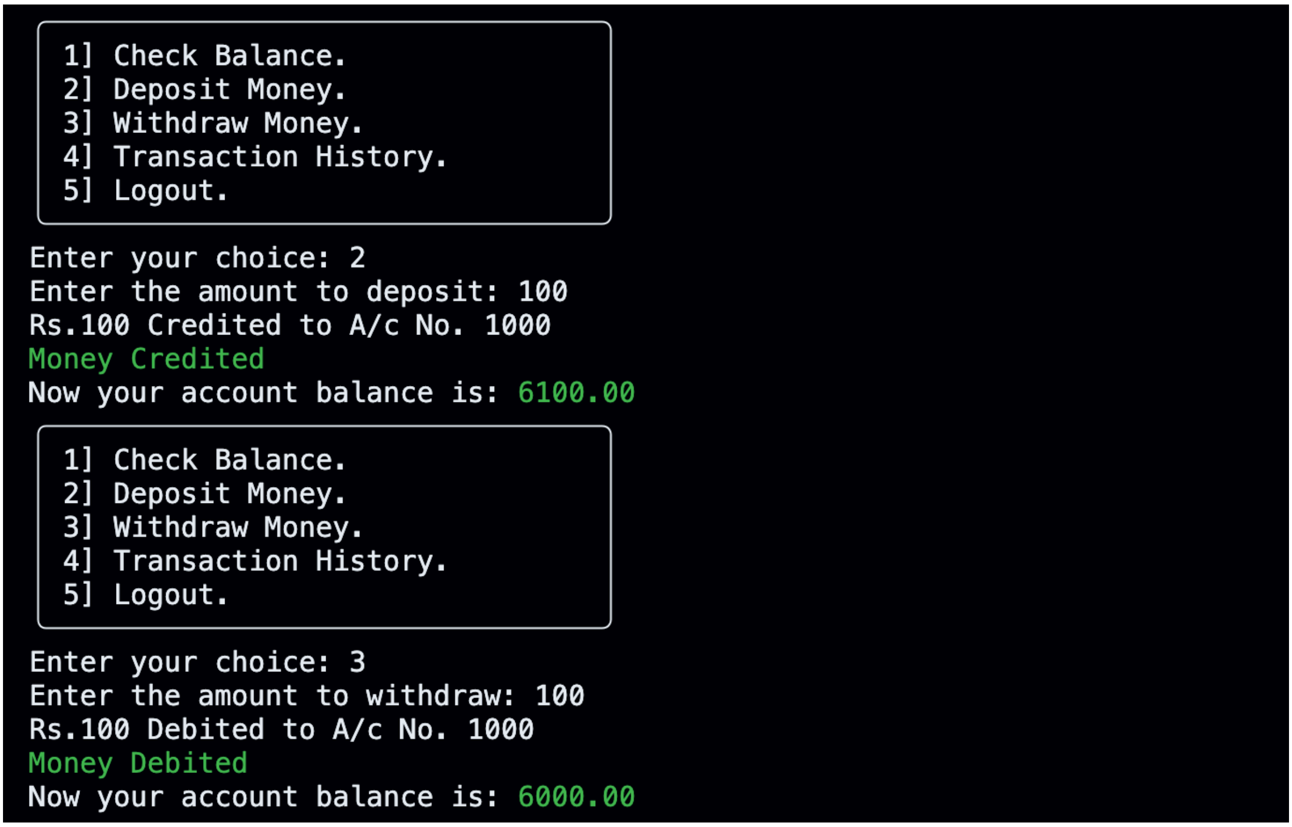


```
(.venv) mananpatel@MANANs-MacBook-Air Python % /Users/mananp
atel/Desktop/Python/.venv/bin/python "/Users/mananpatel/Desk
top/Python/Python & SQL/Task/MyBankBuddy.py"
     Welcome to MyBankBuddy

 1] Create Account.
 2] Login.
 3] Exit.

Enter your choice: 2
Enter Your Phone no: 8490917391
Enter Your Password: Manan@12345
Login Successfully
★  Welcome User!  ★
Loading...
Your Account ID is 1000

 1] Check Balance.
 2] Deposit Money.
 3] Withdraw Money.
 4] Transaction History.
 5] Logout.

Enter your choice: 1
Your Bank Balance is: Rs.6000.00

 1] Check Balance.
 2] Deposit Money.
 3] Withdraw Money.
 4] Transaction History.
 5] Logout.

Enter your choice: 
```

3.1 Deposit & Withdraw Function:

When the program starts, it prompts the user to select an option. Suppose the user chooses option 2 to deposit money. The program then asks the user to enter the amount they wish to deposit. For example, if the user enters ₹100, the program displays the message: "₹100 has been credited to Account No. 1000. Your updated balance is ₹6100.00." This message confirms the successful transaction and shows the updated account balance in a clear and concise manner.

Similarly, for withdrawals, the program prompts the user to select the withdrawal option. When the user selects this option, they are asked to enter the amount they wish to withdraw. For instance, if the user enters ₹100, the program displays the message: "₹100 has been debited from Account No. 1000. Your updated balance is ₹6000.00." This ensures the user is informed of the transaction and the remaining balance in their account. The program provides clear and professional messages for each operation to enhance the user experience.

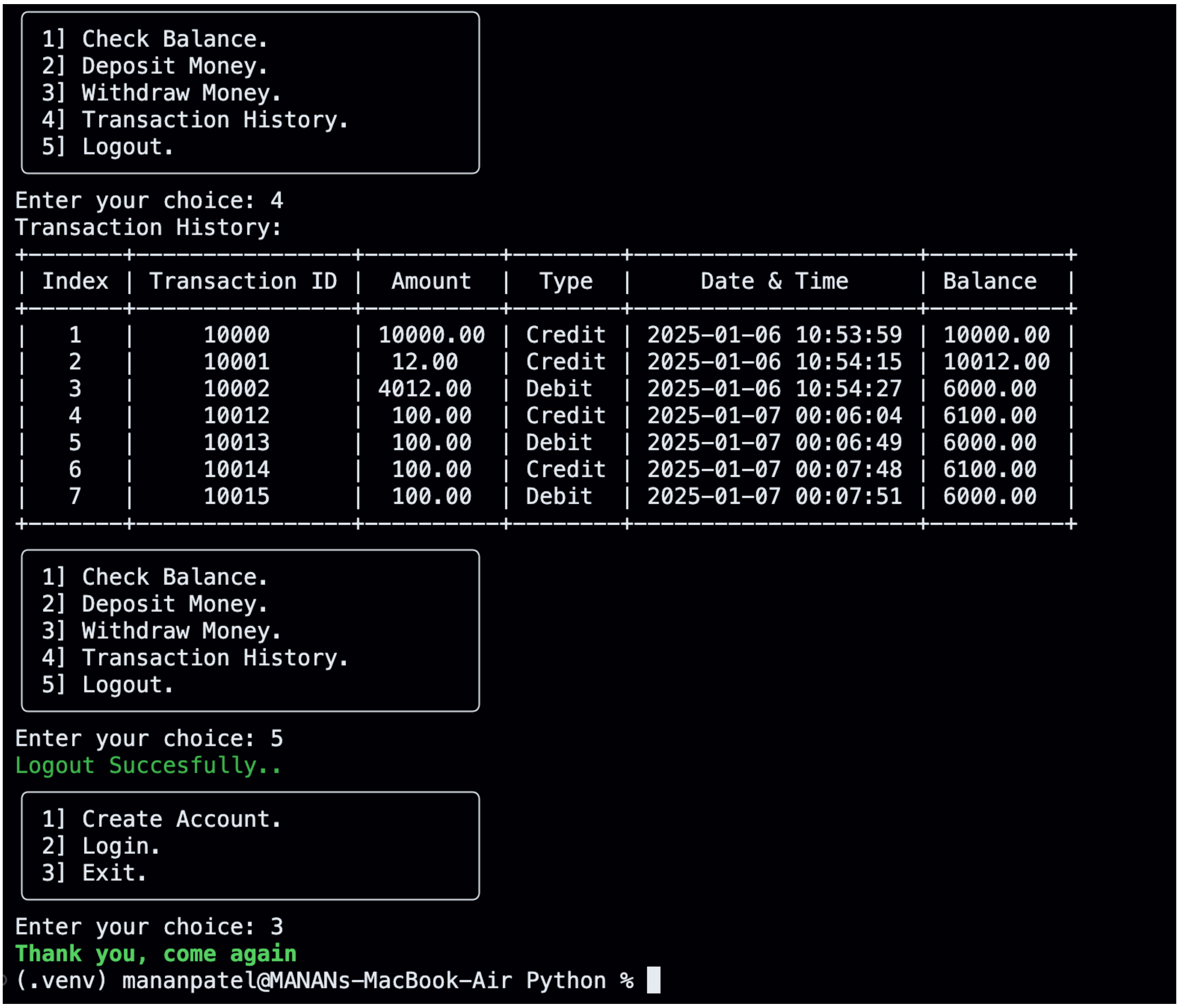


```
  1] Check Balance.
  2] Deposit Money.
  3] Withdraw Money.
  4] Transaction History.
  5] Logout.

Enter your choice: 2
Enter the amount to deposit: 100
Rs.100 Credited to A/c No. 1000
Money Credited
Now your account balance is: 6100.00

  1] Check Balance.
  2] Deposit Money.
  3] Withdraw Money.
  4] Transaction History.
  5] Logout.

Enter your choice: 3
Enter the amount to withdraw: 100
Rs.100 Debited to A/c No. 1000
Money Debited
Now your account balance is: 6000.00
```

3.2 Transaction History:

In the provided scenario, the user selects option 4 to view the transaction history of their bank account. The program responds by displaying a well-formatted transaction history table, created using the PrettyTable library. The table includes the following columns: Index, Transaction ID, Amount, Type, Date & Time, and Balance. This organized view allows the user to easily understand the details of all their transactions.

After viewing the transaction history, the user selects option 5 to log out. Finally, the user chooses option 3 to exit the program. The program then displays the message: "Thank you, come again!" This ensures a smooth and user-friendly experience, leaving a polite and welcoming impression.

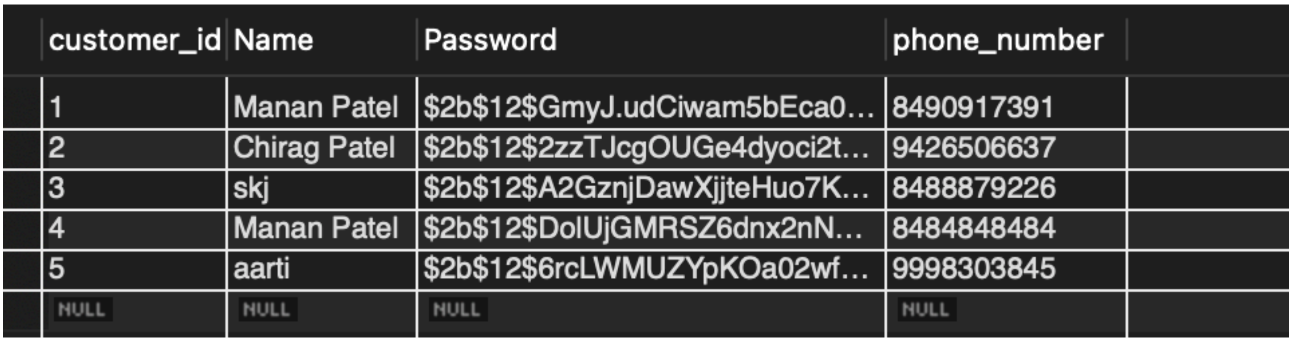


```
  1] Check Balance.
  2] Deposit Money.
  3] Withdraw Money.
  4] Transaction History.
  5] Logout.

Enter your choice: 4
Transaction History:
+-------+----------------+---------+--------+---------------------+----------+
| Index | Transaction ID |  Amount |  Type  |     Date & Time     | Balance  |
+-------+----------------+---------+--------+---------------------+----------+
|   1   |     10000      | 10000.00 | Credit | 2025-01-06 10:53:59 | 10000.00 |
|   2   |     10001      |  12.00  | Credit | 2025-01-06 10:54:15 | 10012.00 |
|   3   |     10002      | 4012.00 | Debit  | 2025-01-06 10:54:27 | 6000.00  |
|   4   |     10012      |  100.00 | Credit | 2025-01-07 00:06:04 | 6100.00  |
|   5   |     10013      |  100.00 | Debit  | 2025-01-07 00:06:49 | 6000.00  |
|   6   |     10014      |  100.00 | Credit | 2025-01-07 00:07:48 | 6100.00  |
|   7   |     10015      |  100.00 | Debit  | 2025-01-07 00:07:51 | 6000.00  |
+-------+----------------+---------+--------+---------------------+----------+

  1] Check Balance.
  2] Deposit Money.
  3] Withdraw Money.
  4] Transaction History.
  5] Logout.

Enter your choice: 5
Logout Succesfully..

  1] Create Account.
  2] Login.
  3] Exit.

Enter your choice: 3
Thank you, come again
(.venv) mananpatel@MANANs-MacBook-Air Python %
```

4. Database Entry

- **Customers Table:**

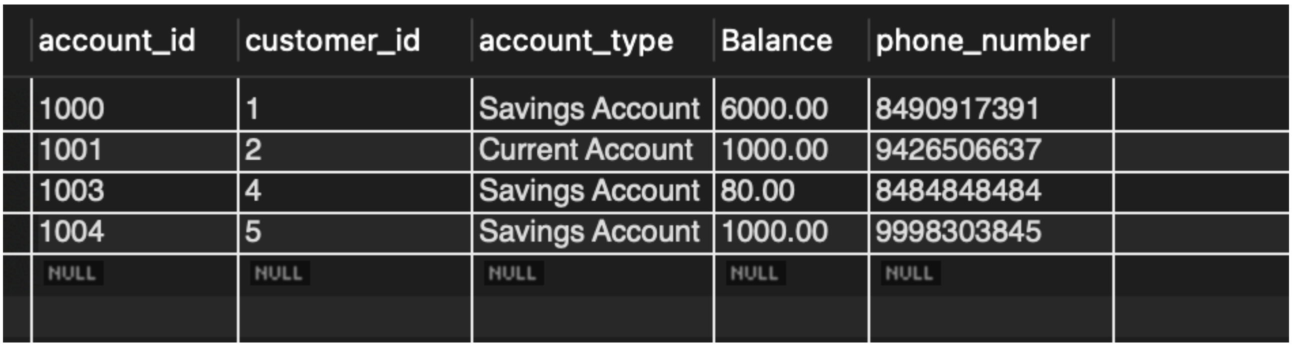
This table represents customer details, including columns for customer ID, name, hashed password, and phone number. It contains five records with customer IDs 1, 2, 3, 4, and 5, where the customer_id column is set as the primary key in the customers table. Passwords are securely hashed using the bcrypt library to enhance data security.



customer_id	Name	Password	phone_number
1	Manan Patel	$2b$12$GmyJ.udCiwam5bEca0…	8490917391
2	Chirag Patel	$2b$12$2zzTJcgOUGe4dyoci2t…	9426506637
3	skj	$2b$12$A2GznjDawXjjteHuo7K…	8488879226
4	Manan Patel	$2b$12$DolUjGMRSZ6dnx2nN…	8484848484
5	aarti	$2b$12$6rcLWMUZYpKOa02wf…	9998303845
NULL	NULL	NULL	NULL

- **Accounts Table:**

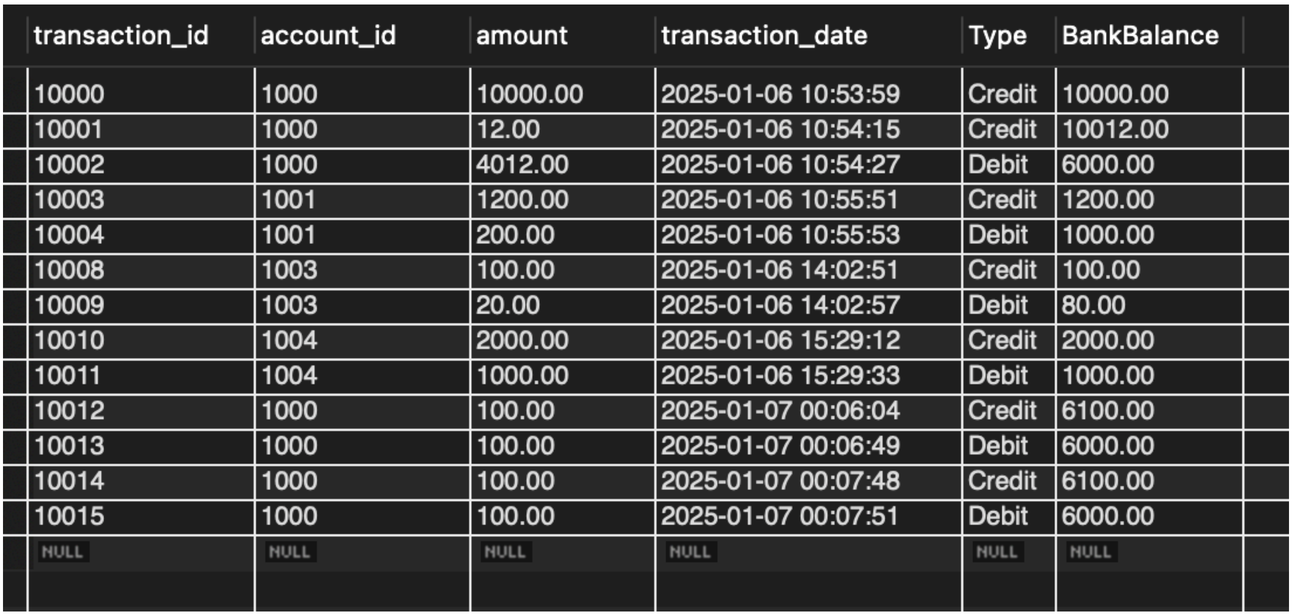
This table contains account details with columns for account ID, customer ID, account type, balance, and phone number. It includes four records with account ID values of 1000, 1001, 1002, and 1003. In this table, account_id serves as the primary key, while customer_id and phone_number are foreign keys referencing the customer table.



account_id	customer_id	account_type	Balance	phone_number
1000	1	Savings Account	6000.00	8490917391
1001	2	Current Account	1000.00	9426506637
1003	4	Savings Account	80.00	8484848484
1004	5	Savings Account	1000.00	9998303845
NULL	NULL	NULL	NULL	NULL

- **Transactions Table:**

This table records all transactions with columns for transaction_id, account_id, transaction_amount, transaction_date, transaction_type (credit or debit), and updated_balance. Each transaction is uniquely identified by the transaction_id, and the account_id links to the account table via a foreign key. The transaction_date captures the exact timestamp, and the updated_balance reflects the account's balance after each transaction, tracking deposits and withdrawals.



transaction_id	account_id	amount	transaction_date	Type	BankBalance
10000	1000	10000.00	2025-01-06 10:53:59	Credit	10000.00
10001	1000	12.00	2025-01-06 10:54:15	Credit	10012.00
10002	1000	4012.00	2025-01-06 10:54:27	Debit	6000.00
10003	1001	1200.00	2025-01-06 10:55:51	Credit	1200.00
10004	1001	200.00	2025-01-06 10:55:53	Debit	1000.00
10008	1003	100.00	2025-01-06 14:02:51	Credit	100.00
10009	1003	20.00	2025-01-06 14:02:57	Debit	80.00
10010	1004	2000.00	2025-01-06 15:29:12	Credit	2000.00
10011	1004	1000.00	2025-01-06 15:29:33	Debit	1000.00
10012	1000	100.00	2025-01-07 00:06:04	Credit	6100.00
10013	1000	100.00	2025-01-07 00:06:49	Debit	6000.00
10014	1000	100.00	2025-01-07 00:07:48	Credit	6100.00
10015	1000	100.00	2025-01-07 00:07:51	Debit	6000.00
NULL	NULL	NULL	NULL	NULL	NULL