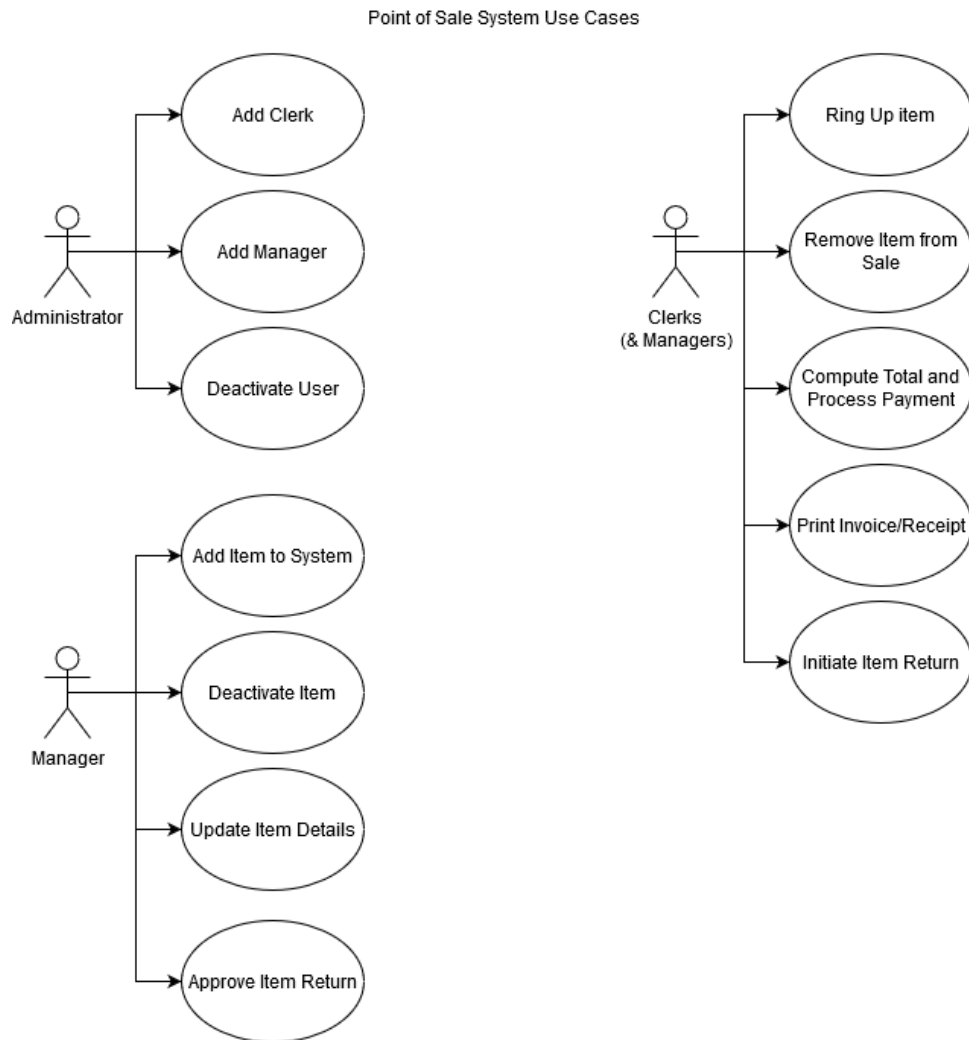


Final Project: Point of Sale System

Use Cases Diagram:



Use Case Elaborations:

Use Case #1

Use Case Name: Add Clerk
 Id: 1
 Scenario: New user is added to system as a clerk
 Triggering Event: New manager is hired
 Brief Description: A new manager is added to the system
 Actors: Administrator
 Assumptions: User is not already in the system
 Frequency of Use: Monthly
 Related Use Cases: Deactivate User, Promote User
 Stakeholders: Administrator, Clerk
 Preconditions: Clerk is not in system. Clerk details are known.
 Postconditions: Clerk has been added to the system with a PIN assigned.
 Main Course: 1. Administrator selects Add Clerk from Employee Management menu

2. Administrator enters clerk details
 3. System prompts for confirmation of details
 4. System offers test of assigned PIN
 5. System adds clerk
- Alternate Course:
1. Employee number already in system
 - a. Offer to change employee number for new employee
 - b. Offer to reactivate deactivated employee
 2. PIN doesn't match
 - a. Re-enter PIN

Use Case #2

- Use Case Name: Add Manager
Id: 2
Scenario: New user is added to system as a manager
Triggering Event: New manager is hired
Brief Description: A new manager is added to the system
Actors: Administrator
Assumptions: Manager is not already in the system (as Clerk)
Frequency of Use: Monthly
Related Use Cases: Deactivate User, Promote User
Stakeholders: Administrator, Manager
Preconditions: Manager is not in system. Manager details are known.
Postconditions: Manager has been added to the system with a PIN assigned.
Main Course:
1. Administrator selects Add Manager from Employee Management menu
 2. Administrator enters manager details
 3. System prompts for confirmation of details
 4. System offers test of assigned PIN
 5. System adds manager
- Alternate Course:
1. Employee number already in system
 - a. Offer to change employee number for new employee
 - b. Offer to promote clerk to manager
 - c. Offer to reactivate deactivated employee
 2. PIN doesn't match
 - a. Re-enter PIN

Use Case #3

- Use Case Name: Deactivate User
Id: 3
Scenario: Current user is deactivated in system.
Triggering Event: User's employment is terminated
Brief Description: A user is deactivated within the system
Actors: Administrator
Assumptions: User is active in the system
Frequency of Use: Monthly
Related Use Cases: Add Clerk, Add Manager
Stakeholders: Administrator, Manager, Clerk
Preconditions: Employee is in active in the system.
Postconditions: Employee has been deactivated.
Main Course:
1. Administrator selects Deactivate User from Employee Management menu
 2. Administrator specifies employee to be deactivated
 3. System prompts for confirmation of deactivation
 4. System deactivates employee

Alternate Course: 1. Employee already deactivated
a. System notifies administrator that user is not currently active

Use Case #4

Use Case Name: Add Item to System
Id: 4
Scenario: New item for sale gets added to system
Triggering Event: New item becomes available
Brief Description: A new item is added to the sale system.
Actors: Manager
Assumptions: Item is not already in the system.
Item details (description, codes, price) are known.
Frequency of Use: Weekly
Related Use Cases: Deactivate Item
Stakeholders: Manager, Clerk
Preconditions: Item is not in system. Item details are known.
Postconditions: Item is available to be sold through the system.
Main Course: 1. Manager selects Add Item from Item Management menu
2. Manager enters item details
3. System prompts for confirmation of details
4. System offers test scan of item (to verify barcode)
5. System adds item
Alternate Course: 1. Item code already in system
a. Offer to change details of existing item
2. Barcode scan doesn't match code entered
a. Offer to scan item again to try for match (Wrong item was scanned)
b. Offer to update item code to match what was scanned

Use Case #5

Use Case Name: Deactivate Item
Id: 5
Scenario: Existing item for sale gets deactivated in the system
Triggering Event: Item is no longer available for sale
Brief Description: An existing item is deactivated in the sale system.
Actors: Manager
Assumptions: Item is active in the system.
Frequency of Use: Weekly
Related Use Cases: Add Item to System
Stakeholders: Manager, Clerk
Preconditions: Item is available in system.
Postconditions: Item is not available to be sold through the system.
Main Course: 1. Manager selects Deactivate Item from Item Management menu
2. Manager specifies the item
3. System prompts for confirmation of deactivation
4. System deactivates item
Alternate Course: 1. Item has already been deactivated
a. System alerts manager that item has already been deactivated
2. Item is not in the system
a. System alerts manager that item is not in the system

Use Case #6

Use Case Name: Update Item Details

Id: 6
 Scenario: Item for sale gets updated
 Triggering Event: Item details change
 Brief Description: Item details are updated in the sale system.
 Actors: Manager
 Assumptions: Item is already in the system.
 Item details (description, codes, price) are known.
 Frequency of Use: Weekly
 Related Use Cases: Add Item to System
 Stakeholders: Manager, Clerk
 Preconditions: Item is in system. Updated details are known.
 Postconditions: Item is available to be sold through the system with updated details.
 Main Course:

1. Manager selects Update Item Details from Item Management menu
2. Manager selects which detail to update
3. Manager inputs updated detail information
4. System prompts for confirmation of details
5. System updates item details

 Alternate Course:

1. Item code not already in system
 - a. Offer to add new item

Use Case #7

Use Case Name: Approve Item Return
 Id: 7
 Scenario: Item returned is being approved and processed
 Triggering Event: Clerk or manager has initiated an item return
 Brief Description: Item return is approved
 Actors: Manager
 Assumptions: Item is eligible for return, return has been initiated
 Frequency of Use: Daily
 Related Use Cases: Initiate Item Return
 Stakeholders: Manager, Clerk, Customer
 Preconditions: Item return has been initiated
 Postconditions: Item return is processed
 Main Course:

1. Summary of returned items is presented
2. Manager logs in
3. Manager approves return
4. Refund is processed

 Alternate Course:

1. Manager initiated return and summary can be approved as displayed

Use Case #8

Use Case Name: Ring Up Item
 Id: 8
 Scenario: Item to be purchased is entered for the sale
 Triggering Event: Customer wishes to purchase item
 Brief Description: Employee rings up an item
 Actors: Clerk or Manager, Customer
 Assumptions: Item has already been entered into the system
 Frequency of Use: Many times each hour
 Related Use Cases: Add Item to System, Update Item Details, Remove Item from Sale
 Stakeholders: Manager, Clerk, Customer
 Preconditions: Item has been entered into system
 Postconditions: Item has been added to the sale transaction

- Main Course:
1. Item is scanned by clerk or manager
 2. Item is added to sale
 3. System displays item details
- Alternate Course:
1. Employee can specify quantity of items after scanning
 2. System updates quantity on display

Use Case #9

- Use Case Name: Remove Item from Sale
Id: 9
Scenario: Item that has been rung up needs to be removed from the transaction
Triggering Event: Customer changes mind, Item is unavailable, Mistaken item scan
Brief Description: A new manager is added to the system
Actors: Clerk or Manager, Customer
Assumptions: Item has been rung up, item should not be processed with sale
Frequency of Use: Daily or Hourly
Related Use Cases: Ring Up Item
Stakeholders: Manager, Clerk, Customer
Preconditions: Item has been rung up
Postconditions: Item is not included on sale
Main Course:
1. Employee selects item from current sale
 2. Employee either:
 - a. Specifies new (reduced) quantity
 - b. Selects remove to eliminate the item
 3. System shows updated quantity (possibly zero)
- Alternate Course:
1. Employee sets system to item removal mode
 2. Employee scans item to indicate which item to remove
 3. Employee specifies quantity to remove (or all)
 4. System displays updated quantity (zero)

Use Case #10

- Use Case Name: Compute Total and Process Payment
Id: 10
Scenario: Item(s) have been rung up and are ready for the sale to be completed
Triggering Event: Customer has completed purchase and all items have been entered
Brief Description: Total is calculated and payment made.
Actors: Clerk or Manager, Customer
Assumptions: Item(s) have been rung up, sale can proceed
Frequency of Use: Hourly
Related Use Cases: Ring Up Item, Print Invoice/Receipt
Stakeholders: Manager, Clerk, Customer
Preconditions: Item(s) have been rung up
Postconditions: Sale is recorded and payment made
Main Course:
1. Employee selects Start Payment from sale screen
 2. System displays total
 3. Employee specifies payment method
 4. System contacts payment device
 5. System records sale and prints receipt
- Alternate Course:
1. Payment is denied
 - a. System offers to cancel transaction

Use Case #11

- Use Case Name: Print Invoice/Receipt

Id: 11
Scenario: Item is being returned by a customer
Triggering Event: Customer brings previously purchased item back for return
Brief Description: Item is processed for return by clerk
Actors: Clerk or Manager, Customer
Assumptions: Sale has been completed
Frequency of Use: Hourly
Related Use Cases: Compute Total and Process Payment
Stakeholders: Manager, Clerk, Customer
Preconditions: Sale has been processed
Postconditions: Receipt has been printed
Main Course:
 1. Sale is completed
 2. Clerk or Manager selects Print Receipt
 3. Receipt is printed
Alternate Course:
 1. Printer Malfunction
 a. Alert Clerk that printing has failed
 b. Provide options to cancel or retry

Use Case #12

Use Case Name: Initiate Item Return
Id: 12
Scenario: Item is being returned by a customer
Triggering Event: Customer brings previously purchased item back for return
Brief Description: Item is processed for return by clerk
Actors: Clerk or Manager, Customer
Assumptions: Item is eligible for return
Frequency of Use: Daily
Related Use Cases: Approve Item Return, Ring Up Item
Stakeholders: Manager, Clerk, Customer
Preconditions: Item is being returned
Postconditions: Item return is pending approval
Main Course:
 1. Item is scanned by clerk or manager
 2. Item is added to return set
 3. Clerk or Manager specifies original sale price for refund amount
 4. Clerk initiates approval request
Alternate Course:
 1. Manager initiates return and automatically proceeds with approval

UI Sketch:

Current User: Bob S. 10:45 AM 4/30/2020

Sales

Returns

Reports

Item Management

User Management

Add Items Remove Item

1 ea. Item One	5.99	5.99
2 ea. Item Two	4.75	9.50
1 ea. Item Three	13.45	13.45

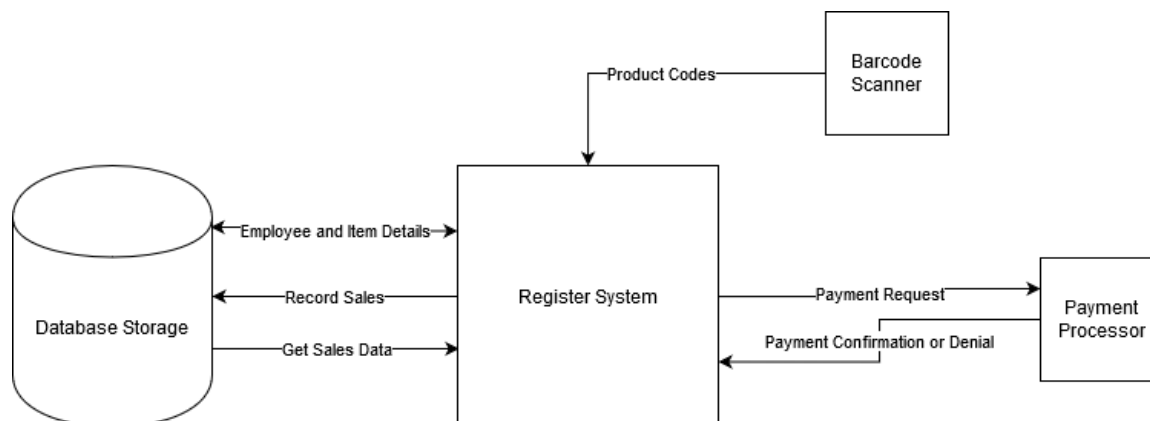
Subtotal: 28.94

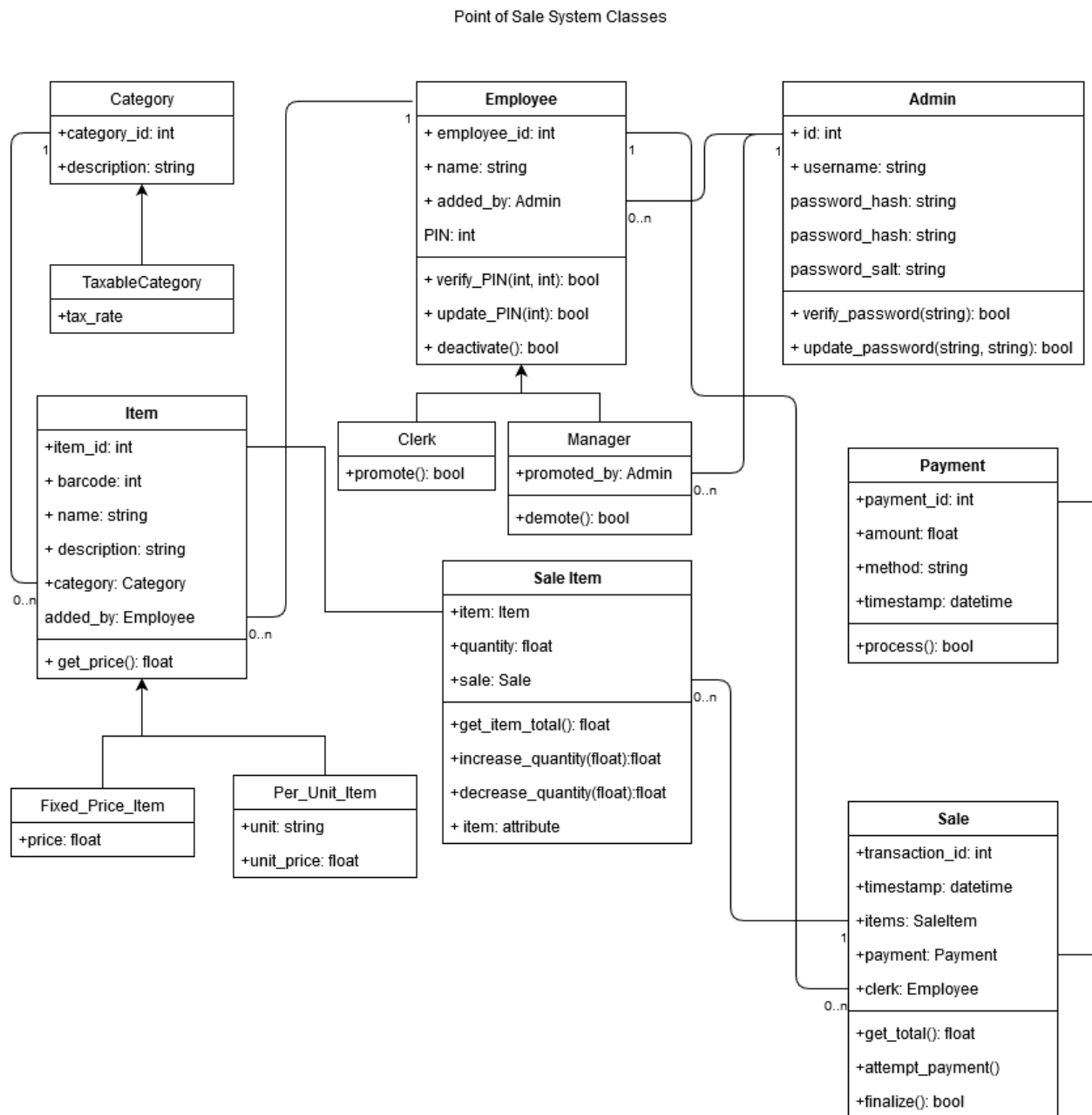
Tax: 1.71

Total: 30.68

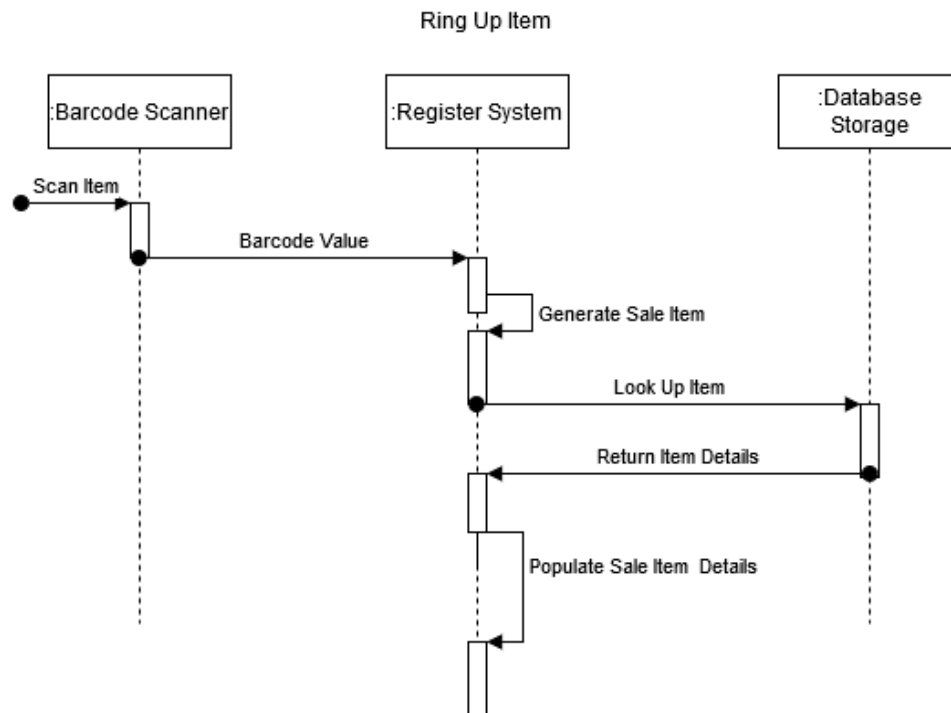
Start Payment

Lock / Sign Out

Architecture Diagram:

Class Diagram:

Sequence Diagrams:



Application:

```

1 # Patrick Johnson          5/1/2020 #
2 # SWDV 630 3W 20/SP2 Final Project #
3 #####
4 #Point of Sale System - Partial Implementation
5 #
6
7 class Category:
8     __next_id = 1
9
10    def __init__(self, description):
11        self.category_id = Category.__next_id
12        Category.__next_id +=1
13        self.description = description
14
15    def __str__(self):
16        return self.description
17
18 class Employee:
19     __next_id = 1
20
21    def __init__(self, name, PIN):
22        self.employee_id = Employee.__next_id
23        Employee.__next_id +=1
24        self.name = name
25        self.__PIN = PIN
26
27    def __repr__(self):
28        return "Employee - ID: {}; Name: {}".format(self.employee_id, self.name)
29
  
```

```

30     def verify_PIN(self, PIN):
31         return self.__PIN == PIN
32
33     def update_PIN(self, old_PIN, new_PIN):
34         if self.verify_PIN(old_PIN):
35             self.__PIN = new_PIN
36             return True
37         else:
38             return False
39
40 class Manager(Employee):
41     pass
42
43 class Clerk(Employee):
44     pass
45
46 class Item:
47     __next_id = 1
48
49     def __init__(self, barcode, description, category, employee):
50         self.item_id = Item.__next_id
51         Item.__next_id += 1
52         self.barcode = barcode
53         self.description = description
54         self.category = category
55         self.employee = employee
56
57     def __repr__(self):
58         return "Item - ID: {} Barcode: {}; Category: {}; Description: {}".format(self.item_id,
59                                                                                     self.barcode,
60                                                                                     self.category,
61                                                                                     self.description)
62
63     def get_price(self):
64         #pass
65         return 5
66
67     def get_Summary(self):
68         return "{} @ {:.2f} ea.".format(self.Quantity, self.UnitCost)
69
70     Total = property(get_price, None, None, "Item Price")
71     Summary = property(get_Summary)
72
73     @classmethod
74     def get_item_from_barcode(cls, barcode):
75         return Item(barcode, "Something", "Generic Category", "Employee")
76
77 class SaleItem:
78     __next_id = 1
79
80     def __init__(self, barcode, quantity, sale):
81         self.sale_item_id = SaleItem.__next_id
82         SaleItem.__next_id += 1
83         self.item = Item.get_item_from_barcode(barcode)
84         self.quantity = quantity
85         self.sale = sale
86
87     def __str__(self):
88         return "{} - {} @ ${}".format(self.item.description,

```

```

89             self.quantity,
90             self.item.get_price())
91
92 if __name__ == "__main__":
93     # Add Employees
94     bob = Manager("Bob S.", "1234")
95     ann = Clerk("Ann D.", "0000")
96
97     print("Bob's employee_id:", bob.employee_id)
98     print("Ann's employee_id:", ann.employee_id)
99
100    # Check PINs:
101    print("Check incorrect PIN for Bob:", bob.verify_PIN("1111"))
102    print("Check correct PIN for Bob: ", bob.verify_PIN("1234"))
103
104    # Update PIN
105    print("Updating Bob's PIN to '5309'")
106    bob.update_PIN("1234", "5309")
107    print("Check old PIN for Bob:", bob.verify_PIN("1234"))
108    print("Check new PIN for Bob:", bob.verify_PIN("5309"))
109
110
111    # Add a category
112    paper_category = Category("Paper Products")
113
114    # Create Some Items
115    itemA = Item(2622977076, "Notebook", paper_category, ann)
116    itemB = Item(5, "Something", 1, bob)
117
118    print("itemA:", itemA)
119    print("itemB:", itemB)
120
121    # Create some sale items:
122    first_sale = None
123    line_item_1 = SaleItem(2622977076, 1, first_sale)
124    line_item_1.item = itemA
125    line_item_2 = SaleItem(5, 3, first_sale)
126
127    print(line_item_1)

```

Output:

```

Bob's employee_id: 1
Ann's employee_id: 2
Check incorrect PIN for Bob: False
Check correct PIN for Bob:  True
Updating Bob's PIN to '5309'
Check old PIN for Bob: False
Check new PIN for Bob: True
itemA: Item — ID: 1 Barcode: 2622977076; Category: Paper Products; Description: Notebook
itemB: Item — ID: 2 Barcode: 5; Category: 1; Description: Something
Notebook — 1 @ $5

```

GitHub Link:

https://github.com/PJohnson9/SWDV630_POS