QAP4

GROUP 9 F23

Noel Hammett – Brenda Armstrong – Ethan Murphy

Due Nov, 28th 2023

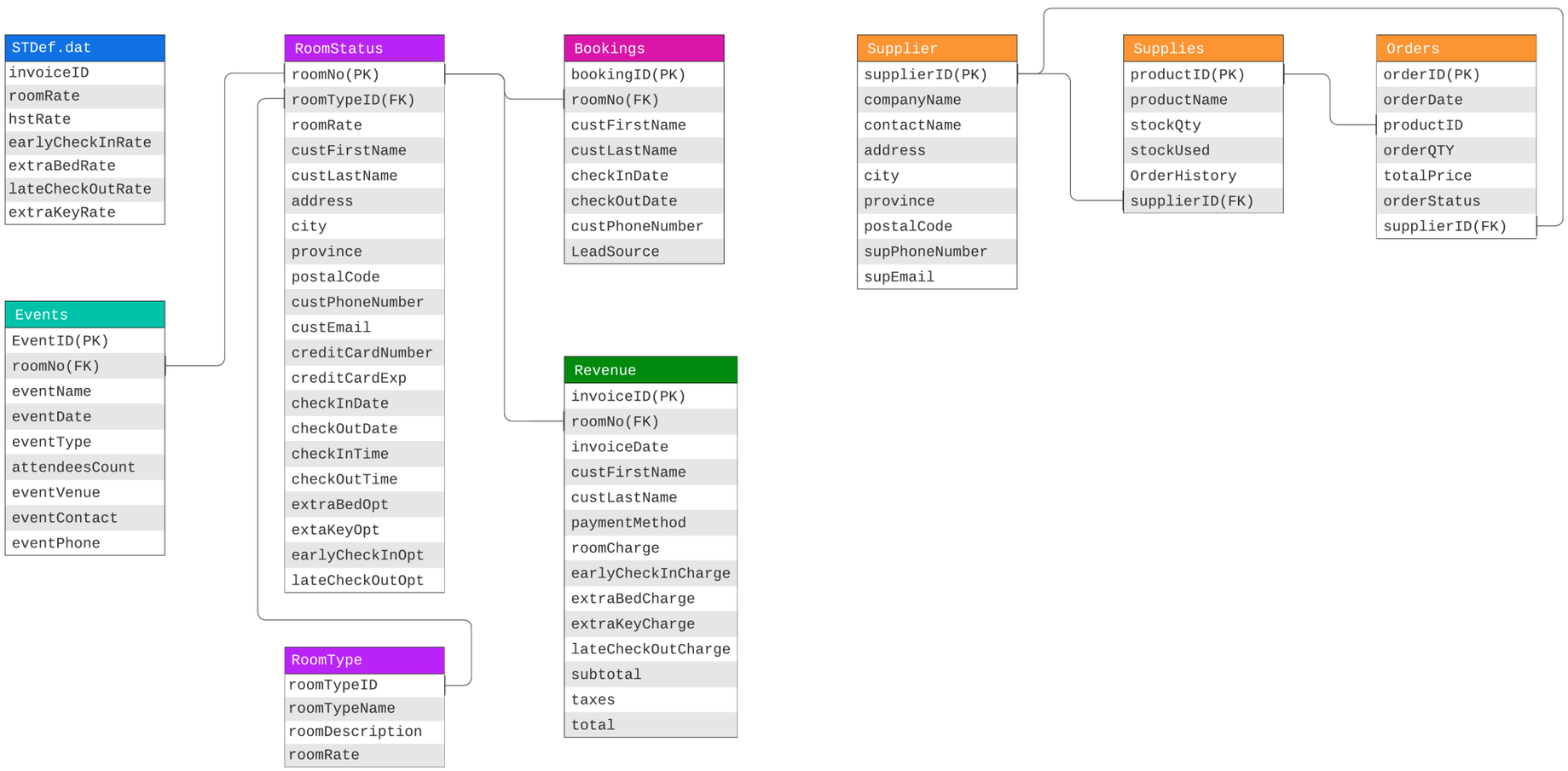
Project 3 – Bash

touch PySample.py vi PySmaple.py (Add code) touch ExtraP.txt vi ExtraP.txt (Add code) touch Stuff.py vi Stuff.py (Add code)	Create 3 files – one called **PySample.py**, one called **ExtraP.txt**, and a third called **Stuff.py**. Add code to each python program and add some text to txt file. Add the words **"RetailCost"** and **"getName"** in one or more of these files.
pwd ls ls -a ls -lh	What is the current directory? List the contents of the current directory. List the contents of the current directory with any hidden files. List the contents of the current directory with any hidden files. List the files with the permissions displayed.
cat PySample.py cat ExtraP.txt cat Stuff.Py head -3 PySample.py tail - -lines 3 Stuff.py tail - -lines 3 Stuff.py ExtraP.txt PySample.py Note: if you want to read a large file in the terminal and not use the gui then you can use the less command to scroll through the information one screen at a time using the space bar.	Display the full contents of each of the files created above. Display only the first 3 lines of **PySample.py** and the last 3 lines of **Stuff.py**. Display the last 3 lines of all files. If you have a file that is large, what options are available to have it appear one screen at a time?
mkdir PythonCurrent mkdir PythonDone mkdir TextFiles mkdir PythonDone/2022 mkdir PythonDone/2023 mkdir TextFiles/Python mkdir TextFiles/Other	Create a directory called **PythonCurrent**, one called **PythonDone** and one called **TextFiles**. In the PythonDone directory create 2 other directories called **2022** and **2023**. In the **TextFiles** directory create 2 other directories called **Python** and **Other**.
cd PythonDone pwd ls	Change the current directory to **PythonDone – notice how the prompt changes to show the current directory**. Use pwd to confirm you are in the PythonDone directory. List the files.
cd PythonCurrent pwd cd PythonDone/2023 pwd cd TextFiles/Other pwd cd ~	Change to the following directories: **PythonCurrent**, **2023**, and **Other**. Prove that you are in the proper directory. Go back to your working directory.
mv Stuff.py PythonCurrent cd PythonCurrent ls cd ~ cp ExtraP.txt TextFiles/Python	Move the **Stuff.py** file to the **PythonCurrent** folder and give it the same name. Check and make sure the file has been moved. Copy the **ExtraP.txt** file to the **Python** folder in **TextFiles**.
cd TextFiles/Python head -6 ExtraP.txt cd ~	Change to the **Python** folder and display the first 6 lines of the file **ExtraP.txt**. Once complete move back to the main folder.
cat TextFiles/Python/ExtraP.txt	Display the contents of the **ExtraP.txt** file from the Python directory in **PythonDone** directory from the current location – you should be in the working directory.
find . -name Stuff.py	From the working directory, find the file called **Stuff.py** searching all subdirectories.
chmod ugo-w PySample.py	Change the permissions in the PySample.py so that the owner, group, and everyone else has only read and execute permission.
grep -r getName grep -ir retailprice grep -r getName > FindResults.txt grep -r RetailCost > FindResults.txt note: you could swap would the > for >> to append the list.	Find the text **getName** in a file and indicate which file(s) it is located –search all subdirectories. Do the same with the word **RetailCost** – in this case allow the search by ignoring case. Direct the output for one of these to a file called **FindResults.txt**. How could you send both results to the same file without overwriting it?
A useful command is the pipe command (|) that can be used to funnel output from one result to another area.	Perform one other command that you feel would be useful. Write it down with an explanation indicating what it is doing and why you feel it would be beneficial.

Project 4 – Entity / Relationship Diagram

QAP 4 - ERD Diagram SD10 Group 9
Sleep-Tite Motel Reservation Mgmt System ERD

Group 9:
Ethan Murphy
Noel Hammett
Brenda Armstrong



ERD Data Schema Overview

Table: STDef.dat
This Defaults table shows the default values used in calculating the invoice amount. These values include the invoice ID, room rate, HST rate, early check-in rate, extra bed rate, and late check-out rate. These values are used as the base for calculating the total amount due for a guest's stay. By setting these default values, the invoicing process can be streamlined and automated, reducing the risk of errors and saving time for staff.

Table: RoomStatus
RoomStatus contains information such as room number, room type, room rate, customer details, check-in and check-out dates and times, and additional options such as extra bed and early check-in. The primary key of the table is the room number, while the room type is a foreign key. The flowchart helps hotel staff to keep track of room availability and occupancy, as well as to manage customer bookings and preferences. Overall, the Room Status Flowchart is a useful tool for streamlining hotel operations and providing quality customer service.

Table: Bookings
Bookings consists of as bookingID (primary key), roomNo (foreign key), custFirstName, custLastName, checkInDate, checkOutDate, custPhoneNumber, and LeadSource. This model can be used to store and manage information related to customer bookings such as their personal details, check-in and check-out dates, and the source through which they made the booking.

Table: Revenue
The Revenue table includes the invoice ID (primary key), room number (foreign key), invoice date, customer first and last name, payment method, room charge, early check-in charge, extra bed charge, extra key charge, late check-out charge, subtotal, taxes, and total. This data can be used to track revenue generated by each room, customer, and payment method. By analyzing this data, the hotel can identify which rooms are generating the most revenue, which customers are spending the most money, and which payment methods are the most popular. This information can be used to make informed business decisions such as adjusting room rates, offering discounts, and improving customer service. Overall, the Revenue Invoice Data is a valuable asset for any hotel to manage their finances effectively.

Table: Supplies
The Supplies data table includes the following attributes: productID (primary key), productName, stockQty, stockUsed, OrderHistory, and supplierID (foreign key). These attributes are important for tracking the inventory of supplies and managing orders from suppliers. The productID is used as the primary key to uniquely identify each product. The supplierID is used as a foreign key to link the supplies to their respective suppliers. This data model can be used to create a database for managing supplies and orders efficiently.

REQUIREMENT: ADD DATA FIELDS

The Sleep-Tite Motel, a hospitality establishment keen on optimizing its operational efficiency, has devised a comprehensive system for managing rooms, bookings, and various facets of its services. In addition to the fundamental entities, such as RoomStatus, Bookings, Revenue, and Supplies, the hotel has integrated several auxiliary tables to enhance its overall management capabilities. The Orders table facilitates the

seamless handling of supply chain logistics by tracking orders placed with suppliers. Events, captured through the Events table, are efficiently managed, allowing the hotel to organize and coordinate various functions. The Suppliers table centralizes crucial information about the hotel's suppliers, promoting effective communication and relationship management. Furthermore, the RoomType table categorizes different room types, playing a pivotal role in pricing strategies, guest satisfaction, and inventory management. Each of these tables is strategically designed to contribute to the holistic functionality of the Sleep-Tite Motel system, ensuring a streamlined and guest-centric operation.

Table: RoomType

- **PK:** RoomTypeID
- **FK:** None
- **Attributes:**
    1. RoomTypeID
    2. RoomTypeName
    3. RoomDescription
    4. RoomRate

Reasoning: The RoomType table is fundamental for categorizing and managing different room types within Sleep-Tite Motel. RoomTypeID, serving as the primary key, ensures unique identification for each room type. RoomTypeName and RoomDescription offer detailed information on the type and features of each room, aiding in guest preference accommodation. RoomRate is crucial for strategic pricing and allows the motel to implement diverse pricing strategies based on room types. This table supports effective inventory management, enhances the guest experience, and facilitates pricing flexibility for various room offerings.

Table: Events

- **PK:** EventID
- **FK:** RoomNo
- **Attributes:**
    1. EventID
    2. RoomNo
    3. EventName
    4. EventDate
    5. EventType
    6. AttendeesCount
    7. EventVenue
    8. EventContact
    9. EventPhone

Reasoning: The Events table plays a pivotal role in orchestrating and tracking a diverse range of events at Sleep-Tite Motel. EventID, functioning as the primary key, ensures the unique identification of each event entry. The foreign key relationship with RoomNo enables efficient room allocation, aligning each event with suitable spaces. EventName, EventDate, EventType, AttendeesCount, EventVenue, EventContact, and EventPhone provide comprehensive details critical for scheduling, coordinating, and communicating event specifics. This structured dataset not only enhances operational efficiency by ensuring optimal room

allocation and facilitating effective communication for a variety of events but also contributes to the overall guest experience, making event management at the motel seamless and organized.

Table: Suppliers

- **PK:** SupplierID
- **FK:** None
- **Attributes:**
    1. SupplierID
    2. CompanyName
    3. ContactName
    4. Address
    5. City
    6. Province
    7. PostalCode
    8. SupPhoneNumber
    9. SupEmail

Reasoning: The Suppliers table serves as a cornerstone in Sleep-Tite Motel's procurement system, facilitating the efficient management of supplier information. SupplierID, the primary key, ensures the unique identification of each supplier entry. CompanyName, ContactName, Address, City, Province, PostalCode, SupPhoneNumber, and SupEmail collectively provide comprehensive details crucial for effective communication and collaboration with suppliers. This structured dataset empowers the motel to maintain a reliable supply chain, track supplier information, and streamline the procurement process, ultimately contributing significantly to operational efficiency and the sustainability of the establishment's procurement operations.

Table: Orders

- **PK:** OrderID
- **FK:** SupplierID
- **Attributes:**
    1. OrderID
    2. OrderDate
    3. ProductID
    4. OrderQTY
    5. TotalPrice
    6. OrderStatus
    7. SupplierID

Reasoning: The Orders table is a central component in Sleep-Tite Motel's procurement system, designed to efficiently manage and track ordering activities. OrderID, the primary key, ensures the uniqueness of each order, providing a reference point for database operations. OrderDate captures the initiation timestamp of each procurement order, facilitating timeline tracking and analysis of ordering patterns. ProductID links orders to specific products in the hotel's inventory, enabling effective inventory management. OrderQTY specifies the quantity of the product being ordered, crucial for managing stock levels. TotalPrice represents the overall financial commitment for the order, aiding in budgeting and

financial analysis. OrderStatus indicates the progress of the order, offering real-time insights into supply chain logistics. The foreign key relationship with SupplierID establishes a crucial link to supplier details, fostering effective communication and collaboration. This structured approach ensures that Sleep-Tite Motel efficiently manages its procurement activities, from order initiation to delivery, contributing to informed decision-making and the overall effectiveness of the hotel's supply chain management.