surf and adventure company
database project

Ryan Corley, Jerome Rufin, Oliviero Figus, Dylan Randle, Katie Waugh, Charles Custer, Aaron Wong, Sean O’Brien, Narek Akopyan, Akash Sharma

[team 1]
surf & adventure co.

577 sandbridge rd
virginia beach, va
what we do

sell

rent

teach
the owner
current operations

- some inventory is tracked manually
- segregated databases
- customer data consists of a basic profile and past rental data
- information may be lost or incorrect when transferred between systems
- current promotions are not well targeted
- employees in store lose time interacting with customers
- employee performance metrics are not well defined
potential benefits

- seasonal schedule and demand prediction
- improved inventory tracking
- two companies integrated more efficiently
- data integrity (especially regarding employee and event scheduling)
- targeted advertising and promotions
- saved time (efficiency)
- tracked human capital
schema: single entities

1. Employee (SSN, FirstName, LastName, DOB, Telephone, City, State, Zip, StreetAddress, Email, Login, Password, I9, VA4, Picture, WorkStartsDate, Salary, PrimaryLocation)
   1a. RentalDepartment (SSN, DrivingAuthorization, Wage)
   1b. Instructor (SSN, DrivingAuthorization, SkillLevel, Wage)
   1c. Shop (SSN, ShopAuthorizationLevel, Wage)
   1d. Laundry (SSN, Wage, LaundryAuthorizationLevel)
   1e. Accounting (SSN, DOB, Wage, AccountingAuthorizationLevel)
   1f. Key_Holder (SSN, AccessLevel)

2. Customer (ID, Telephone, City, State, Zip, StreetAddress, Email, OnListserv)

3. Inventory (InventoryNo, ItemName, Size, Color, Instance#, PO, DateReceived)
   3a. Rental (InventoryNo, Category, VendorName, Description, Tax, Weight, HalfDayRate, DailyRate, WeeklyRate, Condition, NotTransportableByCustomer)
   3b. Stock (InventoryNo, DeptName, VendorName, Description, Cost, Price, Size)
schema: single entities

4. Event (EID, Date, Time, Duration, WeatherDateandTime, CustomerReview, Num_Customers, Price, RequiresEmployeeType, NumberRequired)
   4a. Activity (EID, Description, Capacity, Category, Price, NonInstructorCost)
   4b. Training (EID, Description)
   4c. Promotion (EID, InventoryNo, ID, Description, PromotionType)
5. Vehicle (VIN, LeasedFrom, MonthlyPayment, CurrentStatus)
6. PurchaseOrder (PO#, OrderedBy, OrderDate, DeliveryDate, Discount)
7. Supplier (SID, SupplierName, StreetAddress, City, State, Zip, Phone, Fax, PayableName, PayableAddress, PayableCity, PayableState, PayableZip, RepName, RepEmail, RepPhone, UniqueBrandFeatures)
8. Store_Location (Address, Phone)
9. Linen_Service (LID, LinenServiceName, Phone, StreetAddress, City, State, Zip, PrimaryLiaison, PickupDay, DropDay)
10. House (Address, NameOfHouse, Realtor, NumberOfKings, NumberOfQueens, NumberOfDoubles, NumberOfTwins, Capacity, OwnerProvidedLinens, idleTime, Region)
schema: single entities

11. Order (OID, RelatesToLinenService)
12. Transaction (TID, PO#, AccountantInCharge, RelatesToOrder, Total, PaymentMethod, DebitOrCredit, TransactionDate, TransactionTime, Status, Account)
13. Place (Address, PlaceName, Description)
14. Employee_Category (Title, Description)
15. Customer_Category (Title, Description, HardGoodDiscount, SoftGoodDiscount, RentalItemDiscount, EventDiscount)
16. Rental_Category (Title, Description)
17. Stock_Category (Title, Description)
40. Unavailability (UnavailabilityDateandTime, LengthOfUnavailability)
41. Weather (WeatherDateandTime, Temp, PercentRain, PercentHumidity, WaveHeight)
schema: weak entities

18. Rental_Instance (InventoryNo³ᵃ, RID, CheckinTime, Location¹³, LocationTimestamp)

42. Order_Item (OID¹¹, ID, StartDate, EndDate, DeliveryRequired, InventoryNo³, DeliveryAddress¹⁰)
schema: many to many relationships

19. Order_Transported_By (Vehicle⁵, OID⁴², ID⁴², InventoryNo³, TimeLeft, TimeArrived, SSN¹³)
20. Event_Supplies (InventoryNo¹⁸, RID¹⁶, UsedForEvent⁴)
21. Employee_Training (Employee¹, TrainingEvent⁴ᵇ, EvaluationScore, TimeArrived, TimeCompleted, Leader¹)
22. Employee_Categorization (CategoryTitle¹⁴, Employee¹)
23. Customer_Categorization (CategoryTitle¹⁵, Customer²)
24. Incoming_Inventory (InvoiceNo, InventoryNo³, FromOrder⁶, ArrivalTime, ShippingCosts, TID¹²)
25. Purchase_Order_SuppliedBy (PurchaseOrder⁶, Supplier⁷)
26. Building_Key_Access (KeyHolder⁴ᶠ, KeyAccessTo¹³, TimeOpened, TimeClosed)
27. House_Servicing (House¹⁰, ServicedBy¹)
28. Activity_Staffing (Activity^{4a}, Instructor^{1b})
29. Shop_Order_Handling (Order^{11}, ProcessedBy^{1c})
30. Customers_In_House (Customer^{2}, House^{10}, CheckInDate)
31. Customer_Order (Customer^{2}, Order^{11})
32. Customer_Promotion (Customer^{2}, Promotion^{4c})
33. Customer_Activity (Customer^{2}, Activity^{4a}, ShirtSize, LunchOrder, Yoga, Massage)
35. Stock_Order_Fulfillment (Customer^{2}, StockItemWithOrder^{3b})
37. Stock_Categorization (InventoryNo^{3b}, CategoryTitle)
38. Rental_Categorization (RentalID^{3a}, CategoryTitle)
39. Employee_Unavailability (Employee^{1}, UnavailableFor^{40}, Reason, TimeRequested)
43. Inventory_Promotions (InventoryNo^{3}, EID^{4c})
44. Activity_Promotions (EID^{4a}, EID^{4c})
relationships in access
queries
query one [inventory management]

how many linen sets (and which type) should be processed each day to minimize the time it takes to finish all linen deliveries?
SELECT  
  sum (h.NumberOfKings) as demandking, 
  sum (h.NumberOfQueens) as demandqueen, 
  sum(h.NumberOfDoubles) as demanddouble, 
  sum (h.NumberOfTwins) as demandtwin 
FROM  House as h, CustomerInHouse as c 
WHERE  c.CheckInDate= '12/6/14' and 
  c.address=h.address 
GROUP BY h.Region;
formulation

parameters:
\( t_d \): truck capacity per day (d=1...5)
\( d_{ir} \): demand of set type i in region r
  \( i=k, q, d, t \quad r=1...20 \)
\( s_{id} \): clean sets in inventory of type i on day d
\( p_j \): clean sets picked up of each type i at beginning of week
\( w_i \): weight per unit of each set type i
\( M \): capacity of washing machine

decision variables:
\( x_{id} \): sets of type i washed on day d
\( y_{ird} \): sets of type i delivered to region r on day d
Objective function:

$$\min r_m$$

Subject to

- \(T_{IRD} \times 100 \geq Y_{IRD}; T_{IRD} \text{ binary}\)
- \(R_m \geq \sum R T_{IRD} \forall i, \forall D\)
- \(\sum R Y_{IRD} \geq D_{IR} \forall i, \forall D\)
- \(\sum I \sum R Y_{IRD} \leq T_D \forall D\)
- \(S_{ID} = P_{ID} + X_{ID} + (\sum X_{ID} - \sum R Y_{ID}) \text{ for } D = 1 \ldots D - 1\)
- \(\sum I \sum R Y_{IRD} \leq S_{ID} \forall D\)
- \(X_I \times W_I \leq 3 \times M \forall D, \forall I\)
- \(X_{ID}, Y_{IRD} \text{ integer}\)
constraint explanation

- required to fill demand
- limitation on capacity of delivery trucks
- can only transport clean sets
- weight constraint of washing machine
- number of wash cycles per day
query two [employee performance]

how can we quantify an employee’s productivity to objectively compare his performance with peers?
formulation

performance score : wage
SELECT e.SSN, e.FirstName, e.LastName, i.SkillLevel, 
((e2.EvaluationScore + (sum(e1.CustomerReview) / count(e1.CustomerReview))))/i.wage as ScoreByWage 
FROM Employee e, Event e1, Instructor i, 
Evaluation_Training e2, Activity_Staffing a 
WHERE i.SSN = e.SSN AND i.SSN = a.SSN AND a.EID = 
e1.EID AND e2.SSN = e.SSN 
GROUP BY i.SkillLevel AND e.SSN 
Order BY ScoreByWage DESC;
<table>
<thead>
<tr>
<th>SSN</th>
<th>FirstName</th>
<th>LastName</th>
<th>Skill Level</th>
<th>ScoreByWage</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Nick</td>
<td>Swift</td>
<td>Expert</td>
<td>0.9</td>
</tr>
<tr>
<td>2</td>
<td>Taylor</td>
<td>Long</td>
<td>Expert</td>
<td>0.875</td>
</tr>
<tr>
<td>5</td>
<td>Torin</td>
<td>Kalberg</td>
<td>Expert</td>
<td>0.8</td>
</tr>
<tr>
<td>12</td>
<td>Josh</td>
<td>Childress</td>
<td>Expert</td>
<td>0.65</td>
</tr>
<tr>
<td>17</td>
<td>Blue</td>
<td>Ivy</td>
<td>Expert</td>
<td>0.425</td>
</tr>
<tr>
<td>3</td>
<td>Chse</td>
<td>Shugarman</td>
<td>Advanced</td>
<td>0.777</td>
</tr>
<tr>
<td>4</td>
<td>Austen</td>
<td>Sanders</td>
<td>Advanced</td>
<td>0.722</td>
</tr>
<tr>
<td>8</td>
<td>Steve</td>
<td>Han</td>
<td>Advanced</td>
<td>0.722</td>
</tr>
<tr>
<td>13</td>
<td>Terry</td>
<td>Price</td>
<td>Advanced</td>
<td>0.555</td>
</tr>
<tr>
<td>6</td>
<td>Evan</td>
<td>Sare</td>
<td>Intermediate</td>
<td>0.9375</td>
</tr>
<tr>
<td>7</td>
<td>Rushil</td>
<td>Smith</td>
<td>Intermediate</td>
<td>0.6875</td>
</tr>
<tr>
<td>11</td>
<td>Jay</td>
<td>Zery</td>
<td>Intermediate</td>
<td>0.6875</td>
</tr>
<tr>
<td>16</td>
<td>Mario</td>
<td>Romero</td>
<td>Intermediate</td>
<td>0.625</td>
</tr>
<tr>
<td>18</td>
<td>Jeff</td>
<td>Goff</td>
<td>Beginner</td>
<td>1</td>
</tr>
<tr>
<td>9</td>
<td>Nate</td>
<td>Winston</td>
<td>Beginner</td>
<td>1</td>
</tr>
<tr>
<td>10</td>
<td>Ethan</td>
<td>Xu</td>
<td>Beginner</td>
<td>0.96</td>
</tr>
<tr>
<td>14</td>
<td>Zach</td>
<td>East</td>
<td>Beginner</td>
<td>0.86</td>
</tr>
<tr>
<td>15</td>
<td>Danny</td>
<td>Winn</td>
<td>Beginner</td>
<td>0.5</td>
</tr>
</tbody>
</table>
query three [employee scheduling]

how can we automate our database to dynamically optimize our work-scheduling?
<table>
<thead>
<tr>
<th>Sun 7/6</th>
<th>Mon 7/7</th>
<th>Tue 7/8</th>
<th>Wed 7/9</th>
<th>Thu 7/10</th>
<th>Fri 7/11</th>
<th>Sat 7/12</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coelryn Gone</td>
<td>Chris S 9-2</td>
<td>Chris P 9-6</td>
<td>Coelryn 12-6</td>
<td>Chris S 9-6</td>
<td>Chris P 9-6</td>
<td>Paul T Gone</td>
</tr>
<tr>
<td>John - on the road</td>
<td>Danny B 9-6</td>
<td>Pierce H 9-12</td>
<td>Danny B 9-2</td>
<td>Evan M 12-6</td>
<td>Kyle E 9-6</td>
<td>Afton 9-end</td>
</tr>
<tr>
<td>Brendan W Gone</td>
<td>Logan C Gone</td>
<td>Kyle E 9-6</td>
<td>Shop (2) 9-5 Matt M</td>
<td>Shop AM2 (9-1:30 L)</td>
<td>Shop PM1 (1:30-7)</td>
<td>Shop PM1 (9-2) Darbs</td>
</tr>
<tr>
<td>Logan C Gone</td>
<td>Evan M 12-6</td>
<td>Shop AM2 (9-1:30 L)</td>
<td>Shop AM1 (9-2) Can</td>
<td>Shop AM1 (9-2) Can</td>
<td>Shop PM2 (2-7) Alex</td>
<td>Shop PM1 (9-2) Darbs</td>
</tr>
<tr>
<td>Afton 9-6</td>
<td>Paul T 9-6</td>
<td>Pierce H 9-12</td>
<td>Shop (2) 9-5 Tommy</td>
<td>Shop PM1 (1:30-7)</td>
<td>Shop PM2 (2-7) Alex</td>
<td>Shop PM1 (9-2) Darbs</td>
</tr>
<tr>
<td>Associate Needed 2℃</td>
<td>Tyler B 9-2</td>
<td>Evan M 12-6</td>
<td>Shop AM2 (9-1:30 L)</td>
<td>Shop AM1 (9-2) Can</td>
<td>Shop PM2 (2-7) Alex</td>
<td>Shop PM1 (9-2) Darbs</td>
</tr>
<tr>
<td>Julian S 9-2</td>
<td>Will K 9-6</td>
<td>Pierce H 9-12</td>
<td>Shop AM2 (9-1:30 L)</td>
<td>Shop PM1 (1:30-7)</td>
<td>Shop PM2 (2-7) Alex</td>
<td>Shop PM1 (9-2) Darbs</td>
</tr>
<tr>
<td>Kyle E 9-6</td>
<td>Shop (2) 9-5 Matt M</td>
<td>Shop (2) 9-5 Matt M</td>
<td>Shop PM1 (1:30-7)</td>
<td>Shop PM1 (1:30-7)</td>
<td>Shop PM1 (9-2) Darbs</td>
<td>Shop PM1 (9-2) Darbs</td>
</tr>
<tr>
<td>Shop AM2 (9-1:30 L)</td>
<td>Shop PM2 (2-7) Alex</td>
<td>Shop PM2 (2-7) Alex</td>
<td>Shop PM1 (1:30-7)</td>
<td>Shop PM1 (1:30-7)</td>
<td>Shop PM1 (9-2) Darbs</td>
<td>Shop PM1 (9-2) Darbs</td>
</tr>
<tr>
<td>Shop PM1 (1:30-6) Gi</td>
<td>Tommy on print</td>
<td>Tommy on print</td>
<td>Tommy on print</td>
<td>Tommy on print</td>
<td>Tommy on print</td>
<td>Tommy on print</td>
</tr>
<tr>
<td>Shop PM2 (2-6) Darbs</td>
<td>Shop PM2 (2-6) Darbs</td>
<td>Shop PM2 (2-6) Darbs</td>
<td>Shop PM2 (2-6) Darbs</td>
<td>Shop PM2 (2-6) Darbs</td>
<td>Shop PM2 (2-6) Darbs</td>
<td>Shop PM2 (2-6) Darbs</td>
</tr>
<tr>
<td>Chris Off</td>
<td>Karis Class 5-9</td>
<td>Karis Class 5-9</td>
<td>Karis Class 5-9</td>
<td>Karis Class 5-9</td>
<td>Karis Class 5-9</td>
<td>Karis Class 5-9</td>
</tr>
<tr>
<td>David 9-5</td>
<td>LAUNDRY (1-7) Billy</td>
<td>LAUNDRY (1-7) Billy</td>
<td>LAUNDRY (1-7) Billy</td>
<td>LAUNDRY (1-7) Billy</td>
<td>LAUNDRY (1-7) Billy</td>
<td>LAUNDRY (1-7) Billy</td>
</tr>
<tr>
<td>LAUNDRY (1-7) Go</td>
<td>LAUNDRY (1-7) Go</td>
<td>LAUNDRY (1-7) Go</td>
<td>LAUNDRY (1-7) Go</td>
<td>LAUNDRY (1-7) Go</td>
<td>LAUNDRY (1-7) Go</td>
<td>LAUNDRY (1-7) Go</td>
</tr>
<tr>
<td>LAUNDRY (1-7) Bro</td>
<td>LAUNDRY (1-7) Bro</td>
<td>LAUNDRY (1-7) Bro</td>
<td>LAUNDRY (1-7) Bro</td>
<td>LAUNDRY (1-7) Bro</td>
<td>LAUNDRY (1-7) Bro</td>
<td>LAUNDRY (1-7) Bro</td>
</tr>
<tr>
<td>Nick</td>
<td>LAUNDRY (1-7) Bro</td>
<td>LAUNDRY (1-7) Bro</td>
<td>LAUNDRY (1-7) Bro</td>
<td>LAUNDRY (1-7) Bro</td>
<td>LAUNDRY (1-7) Bro</td>
<td>LAUNDRY (1-7) Bro</td>
</tr>
<tr>
<td>David 9-5</td>
<td>Shop AM 2 (9-1:30 B)</td>
<td>Shop AM1 (8-5) Tommi</td>
<td>Shop AM1 (1:30-6) Cts</td>
<td>Shop PM1 (1:30-7) Dri</td>
<td>Shop PM2 (2-7) Alex</td>
<td>Shop PM1 (9-2) Darbs</td>
</tr>
<tr>
<td>Shop PM1 (9-2) Darbs</td>
<td>Shop AM1 (9-2) Darbs</td>
<td>Shop PM1 (9-2) Darbs</td>
<td>Shop PM1 (9-2) Darbs</td>
<td>Shop PM1 (9-2) Darbs</td>
<td>Shop PM1 (9-2) Darbs</td>
<td>Shop PM1 (9-2) Darbs</td>
</tr>
<tr>
<td>SHOP 9-2</td>
<td>SHOP 9-2</td>
<td>SHOP 9-2</td>
<td>SHOP 9-2</td>
<td>SHOP 9-2</td>
<td>SHOP 9-2</td>
<td>SHOP 9-2</td>
</tr>
<tr>
<td>David B’s Last Day</td>
<td>Daniel</td>
<td>Daniel</td>
<td>Daniel</td>
<td>Daniel</td>
<td>Daniel</td>
<td>Daniel</td>
</tr>
<tr>
<td>David / Fake Coa R</td>
<td>David / Fake Coa R</td>
<td>David / Fake Coa R</td>
<td>David / Fake Coa R</td>
<td>David / Fake Coa R</td>
<td>David / Fake Coa R</td>
<td>David / Fake Coa R</td>
</tr>
</tbody>
</table>
part one

unavailability

not working  working + category

+ wage
1. SELECT COUNT(e.RequiresEmployeeType)*e.NumberRequired
   FROM Event e
   GROUP BY e.EID
   WHERE e.EventDate > sysDate() AND e.EventDate < sysDate() + 7;

2. SELECT w.SSN, w.Wage*(e.Duration) as WageCost
   FROM Employee w
   WHERE not exists (SELECT *
       FROM Employee_Unavailability u, Event e
       WHERE u.SSN = w.SSN
       AND u.TimeRequested = e.Time);
<table>
<thead>
<tr>
<th>EID</th>
<th>Event_Date</th>
<th>Employee_Type</th>
<th>Number_Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>11/19/2014</td>
<td>Cashier</td>
<td>20</td>
</tr>
<tr>
<td>2</td>
<td>11/14/2014</td>
<td>Yoga Instructor</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>12/3/2014</td>
<td>Surf Instructor</td>
<td>10</td>
</tr>
<tr>
<td>4</td>
<td>10/14/2014</td>
<td>Surf Instructor</td>
<td>15</td>
</tr>
<tr>
<td>5</td>
<td>11/5/2014</td>
<td>Sailing Instructor</td>
<td>20</td>
</tr>
<tr>
<td>*</td>
<td>0</td>
<td></td>
<td>0</td>
</tr>
</tbody>
</table>
$X_{ij}$: if worker $i$ is assigned to event $j$

$C_{ij}$: cost for employee $i$ for length of event $j$

$i = \{1, 2, \ldots, n\}$ $n = \#$ workers

$j = \{1, 2, \ldots, m\}$ $m = \#$ events

and no employee can be scheduled for more than 3 consecutive events
\[
\begin{align*}
\text{min} & \quad \sum_{i=1}^{n} \sum_{j=1}^{m} c_{ij} x_{ij} \\
\text{s.t.} & \quad \sum_{i=1}^{n} c_{i1} x_{i1} \geq a \\
& \quad \sum_{i=1}^{n} x_{ij} \geq b \quad \forall j > 1 \\
& \quad x_{ij} = \{0, 1\}
\end{align*}
\]
<table>
<thead>
<tr>
<th>SSN</th>
<th>Event</th>
<th>Cost_For_Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>209876349</td>
<td>1</td>
<td>30000</td>
</tr>
<tr>
<td>209876349</td>
<td>2</td>
<td>60000</td>
</tr>
<tr>
<td>209876349</td>
<td>3</td>
<td>15000</td>
</tr>
<tr>
<td>209876349</td>
<td>4</td>
<td>20000</td>
</tr>
<tr>
<td>209876349</td>
<td>5</td>
<td>30000</td>
</tr>
<tr>
<td>234522998</td>
<td>1</td>
<td>40000</td>
</tr>
<tr>
<td>234522998</td>
<td>2</td>
<td>80000</td>
</tr>
<tr>
<td>234522998</td>
<td>3</td>
<td>20000</td>
</tr>
<tr>
<td>234522998</td>
<td>4</td>
<td>26666.66</td>
</tr>
<tr>
<td>234522998</td>
<td>5</td>
<td>40000</td>
</tr>
<tr>
<td>245324564</td>
<td>1</td>
<td>20000</td>
</tr>
<tr>
<td>245324564</td>
<td>2</td>
<td>40000</td>
</tr>
<tr>
<td>245324564</td>
<td>3</td>
<td>10000</td>
</tr>
<tr>
<td>245324564</td>
<td>4</td>
<td>13333.33</td>
</tr>
<tr>
<td>245324564</td>
<td>5</td>
<td>20000</td>
</tr>
<tr>
<td>249453211</td>
<td>1</td>
<td>45000</td>
</tr>
<tr>
<td>249453211</td>
<td>2</td>
<td>90000</td>
</tr>
<tr>
<td>249453211</td>
<td>3</td>
<td>22500</td>
</tr>
<tr>
<td>249453211</td>
<td>4</td>
<td>30000</td>
</tr>
<tr>
<td>249453211</td>
<td>5</td>
<td>45000</td>
</tr>
<tr>
<td>257894909</td>
<td>1</td>
<td>32500</td>
</tr>
<tr>
<td>257894909</td>
<td>2</td>
<td>65000</td>
</tr>
<tr>
<td>257894909</td>
<td>3</td>
<td>16250</td>
</tr>
<tr>
<td>257894909</td>
<td>4</td>
<td>216666.66</td>
</tr>
<tr>
<td>257894909</td>
<td>5</td>
<td>32500</td>
</tr>
<tr>
<td>279884234</td>
<td>1</td>
<td>27500</td>
</tr>
<tr>
<td>279884234</td>
<td>2</td>
<td>55000</td>
</tr>
<tr>
<td>279884234</td>
<td>3</td>
<td>13750</td>
</tr>
<tr>
<td>279884234</td>
<td>4</td>
<td>18333.33</td>
</tr>
<tr>
<td>279884234</td>
<td>5</td>
<td>27500</td>
</tr>
</tbody>
</table>
query four: event profit

Which events are consistently the most profitable?
formulation

\[ p = (\# \text{ customers} \times \$ \text{ activity}) - (w \text{ instructor} + (vc \times \# \text{ customers})) \]

\[ \text{ap (by category)} = \text{sum } p \div \# \text{ events} \]

\[ \% \text{ attendance} = \# \text{ customers} \div \text{capacity} \]
FROM Activity a, Customer_Activity c, Event e, Instructor i, Activity_Staffing a2
WHERE e.EID=a.EID and a.EID=c.Activity and i.SSN = a2.Instructor and a2.Activity=a.EID and e.EventDate>”” and e.EventDate<””
GROUP BY a.Category
ORDER BY Profit desc;
query five [strategic promotions]

which promotions are most profitable or increase turnover?

(given weather and time-of-year)
RAD GRADS & COOL DADS
SUMMER KICKOFF SALE!
June 6 - June 15

We've lined up the sickest deals of the season!

TENT SALE: $10, $20, $30 racks
25% off surf & SUP accessories
Buy-One, Get One Half Off on ALL APPAREL.
Special Markdowns & Deals on Select Surfboards & SUPs
the most profitable promotions based on time of year and weather (for events, inventory items, and rental items)

profit from activity:
(#customers \times $ activity)-($ instructor \times duration)-non-instructor cost

profit from inventory item:
price-cost

SELECT e.EID, SUM(e.Price* e.Num_Customers - w.Wage* e.Duration) as Profit, w.Temp, e.eventDate
FROM Promotion p, Event e, Event_Supplies s, Employee w, Activity_Staffing a, Weather w
WHERE p.EID = e.EID and e.EID = a.EID and w.SSN = a.SSN and p.PromotionType = "general" and w.PercentRain < 0.20 and e.Date LIKE '%-Jun-%' OR e.Date LIKE '%-Aug-%' OR e.Date like '%-Jul-%'
GROUP BY e.EID
ORDER BY Profit DESC;
the most profitable customer groups for a given promotion (based on customer city/zipcode)

1. calculate total profit per promotion
2. calculate total profit
3. sort customers groups by profit

```
SELECT e.EID, SUM(s.price - s.cost) as Profit, p.InventoryNo, e.eventDate, c.ID, c.Zip
FROM stock s, Promotion p, Event e, Weather w, Customer c
WHERE p.InventoryNo = s.InventoryNo and p.ID = c.ID and p.PromotionType = "mail"
GROUP BY c.City and c.State
ORDER BY Profit DESC;
```
promotions that will lead to the highest turnover of inventory items

1. Current date - promotion date
2. Sum turnover time of specific stock item category / # items bought in promotion
3. Sort item categories by avg. turnover

SELECT e.EID, SUM(p.EventDate - t.TransactionDate) as TurnOverTime, p.InventoryNo, e.eventDate, i.PO#, pos.Supplier
FROM stock s, Promotion p, Customer c, Stock_Categorization sc, Transaction t, Order o, Stock_Order_Fulfillment sof, inventory i, PurchaseOrder po, Supplier s SID, Purchase_Order_SuppliedBy pos
WHERE p.InventoryNo = s.InventoryNo and sc.InventoryNo = s.InventoryNo and p.PromotionType = “mail” and p.ID = c.ID, t.RelatesToOrder = order.OID and sof.StockItemWithOrder = s.InventoryNo and c.ID = sof.Customer and i.InventoryNo = s.InventoryNo and i.PO# = po.PO# and po.PO# = pos.PurchaseOrder and s.SID = pos.Supplier
GROUP BY sc.Title
ORDER BY TurnOverTime;
normalization \([1nf]\)

Entity number 7 is not in First Normal Form:

7. Supplier\((SID, \text{SupplierName}, \text{StreetAddress}, \text{City}, \text{State}, \text{Zip}, \text{Phone}, \text{Fax}, \text{PayableName}, \text{PayableAddress}, \text{PayableCity}, \text{PayableState}, \text{PayableZip}, \text{RepName}, \text{RepEmail}, \text{RepPhone}, \text{UniqueBrandFeatures})\)

The attribute UniqueBrandFeatures is multi-valued thus it is, by definition, not in 1NF

In order to fix this, we can add a new relation:

\(\text{SupplierFeatures}(SID, \text{UniqueBrandFeatures})\)
normalization [2nf]

19. Order_Transported_By(Vehicle, OID, ID, InventoryNo, TimeLeft, TimeArrived, SSN)
   ** not in 2NF (OID, ID -> InventoryNo)

This relation is not in 2NF because a proper subset of the PK can determine one of the non-prime attributes. Namely: {OID,ID} → InventoryNo

To fix this, we can simply delete the InventoryNo attribute. This is because we already have a relation which determines InventoryNo from {OID,ID}:

42. Order_Item (OID, ID, StartDate, EndDate, DeliveryRequired, InventoryNo, DeliveryAddress)
normalization \([3nf]\)

1. Employee(\(SSN, \) FirstName, LastName, DOB, Telephone, City, State, Zip, StreetAddress, Email, Login, Password, I9, VA4, Picture, WorkStartDate, Salary, Wage, PrimaryLocation\(^8\))

NOT in 3NF because a non-prime attribute can determine another non-prime attribute.
Namely: an employee’s city can determine their primary location. To fix this, we add the relation:

PrimaryLocation(\(SSN, \text{City}, \text{PrimaryLocation}\)^8)
questions?