Week 5 - Wednesday

Questions

1. List all customers who live in Texas (use

JOINs)

**SELECT** customer.address\_id, customer\_id, first\_name, last\_name

**FROM** customer

**LEFT** **JOIN** address **ON** customer.address\_id = address.address\_id

**WHERE** district **ILIKE** 'texas'

**GROUP** **BY** customer.address\_id, customer\_id

**ORDER** **BY** last\_name **ASC**;

Returns:

address\_id|customer\_id|first\_name|last\_name|

----------+-----------+----------+---------+

122| 118|Kim |Cruz |

10| 6|Jennifer |Davis |

405| 400|Bryan |Hardison |

310| 305|Richard |Mccrary |

567| 561|Ian |Still |

2. Get all payments above $6.99 with the Customer's Full

Name:

**SELECT** last\_name, first\_name, amount

**FROM** customer

**LEFT** **JOIN** payment **ON** customer.customer\_id = payment.customer\_id

**WHERE** amount >= 6.99

**GROUP** **BY** last\_name, first\_name, amount

**ORDER** **BY** last\_name **ASC**

Returns:

last\_name|first\_name|amount|

---------+----------+------+

Andrews |Ida | 12.96|

Deloach |Alvin | 41.44|

Graf |Douglas |927.67|

Martino |Harold | 89.99|

Mcadams |Alfredo | 82.94|

Menard |Peter |175.99|

Menard |Peter |176.99|

Menard |Peter |177.99|

Menard |Peter |179.99|

Menard |Peter |180.99|

Menard |Peter |181.99|

Menard |Peter |182.99|

Menard |Peter |183.99|

Menard |Peter |184.99|

Smith |Mary | 12.95|

Smith |Mary | 42.95|

Smith |Mary | 85.95|

Smith |Mary |486.86|

Smith |Mary |988.45|

3. Show all customers names who have made payments over $175(use

subqueries):

**SELECT** last\_name, first\_name, amount

**FROM** (

**SELECT** last\_name, first\_name, amount

**FROM** customer

**LEFT** **JOIN** payment **ON** customer.customer\_id = payment.customer\_id

**WHERE** amount >= 6.99

**GROUP** **BY** last\_name, first\_name, amount

) **AS** six\_nienty\_nine\_payment\_customers

**WHERE** amount > 175

**GROUP** **BY** last\_name, first\_name, amount

**ORDER** **BY** last\_name **ASC**

Returns:

last\_name|first\_name|amount|

---------+----------+------+

Graf |Douglas |927.67|

Menard |Peter |175.99|

Menard |Peter |176.99|

Menard |Peter |177.99|

Menard |Peter |179.99|

Menard |Peter |180.99|

Menard |Peter |181.99|

Menard |Peter |182.99|

Menard |Peter |183.99|

Menard |Peter |184.99|

Smith |Mary |486.86|

Smith |Mary |988.45|

4. List all customers that live in Nepal (use the city

table)

**SELECT** last\_name, first\_name, customer\_id

**FROM** customer

**FULL** **JOIN** address **ON** customer.address\_id = address.address\_id

**FULL** **JOIN** city **ON** address.city\_id = city.city\_id

**FULL** **JOIN** country **ON** city.country\_id = country.country\_id

**WHERE** country **ILIKE** 'nepal'

**GROUP** **BY** customer\_id, last\_name, first\_name

**ORDER** **BY** last\_name **ASC**;

Returns:  
  
last\_name|first\_name|customer\_id|

---------+----------+-----------+

Schuler |Kevin | 321|

5. Which staff member had the most

transactions?

-- Tried to combine payment and rental tables and count rental id\_transactions by staff\_id repeated in both tables only once

**SELECT** staff.staff\_id, staff.first\_name, staff.last\_name, **count**(\*) **AS** transaction\_count

**FROM** staff

**JOIN** (**SELECT** staff\_id, rental\_id **FROM** payment

**UNION**

**SELECT** staff\_id, rental\_id **FROM** rental

) **AS** combined\_data

**ON** staff.staff\_id = combined\_data.staff\_id

**GROUP** **BY** staff.staff\_id, staff.first\_name, staff.last\_name

**ORDER** **BY** transaction\_count **DESC**;

Returned:

staff\_id|first\_name|last\_name|transaction\_count|

--------+----------+---------+-----------------+

2|Jon |Stephens | 11685|

1|Mike |Hillyer | 11683|

6. How many movies of each rating are

there?

**SELECT** rating, **count**(rating)

**FROM** inventory

**LEFT** **JOIN** film **ON** inventory.film\_id = film.film\_id

**GROUP** **BY** rating

**ORDER** **BY** **count**(rating) **DESC**

Returns:  
rating|count|

------+-----+

PG-13 | 1018|

NC-17 | 940|

PG | 924|

R | 908|

G | 791|

7.Show all customers who have made a single payment

above $6.99 (Use Subqueries):

**SELECT** customer.customer\_id, last\_name, first\_name, pmt\_count

**FROM** customer

**RIGHT** **JOIN** (

**SELECT** **DISTINCT** customer\_id, **count**(amount) **AS** pmt\_count

**FROM** payment

**WHERE** amount > 6.99

**GROUP** **BY** payment.customer\_id) **AS** paying\_custs

**ON** customer.customer\_id = paying\_custs.customer\_id

**ORDER** **BY** pmt\_count **DESC**

Returns:

customer\_id|last\_name|first\_name|pmt\_count|

-----------+---------+----------+---------+

341|Menard |Peter | 22|

1|Smith |Mary | 5|

343|Graf |Douglas | 1|

467|Deloach |Alvin | 1|

567|Mcadams |Alfredo | 1|

342|Martino |Harold | 1|

183|Andrews |Ida | 1|

8. How many free rentals did our stores give away?

--Final question assuming you want me to create a function

**CREATE** **OR** **REPLACE** **FUNCTION** calc\_free\_rental\_count()

**RETURNS** **INTEGER** **AS**

**$$**

**DECLARE**

x **INTEGER**;

y **INTEGER**;

**RESULT** **INTEGER**;

**BEGIN**

--Query to fetch total rental count and assign into x

**SELECT** **count**(\*) **INTO** x

**FROM** payment

**LEFT** **JOIN** rental **ON** payment.rental\_id = rental.rental\_id

**LEFT** **JOIN** inventory **ON** rental.inventory\_id = inventory.inventory\_id;

--Query to fetch rental counts greater than zero and assign into y

**SELECT** **count**(\*) **INTO** y

**FROM** payment

**LEFT** **JOIN** rental **ON** payment.rental\_id = rental.rental\_id

**LEFT** **JOIN** inventory **ON** rental.inventory\_id = inventory.inventory\_id

**WHERE** amount > 0;

--Calculate the difference between total and paid amount rentals

**RESULT** := x - y;

--Return the result

**RETURN** **RESULT**;

**END**;

**$$**

**LANGUAGE** plpgsql;

**SELECT** calc\_free\_rental\_count();

Returns:

calc\_free\_rental\_count|

----------------------+

14564|

I could have written a query to fetch that value, but it seemed like that’s the value you were looking for, and I figured you wanted me to write a function at some point! Please let me know if I misunderstood any of the questions; some were a bit vague for my understanding, but I did the best I could. John-