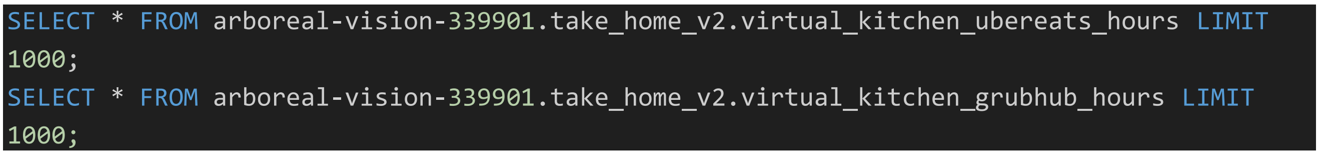
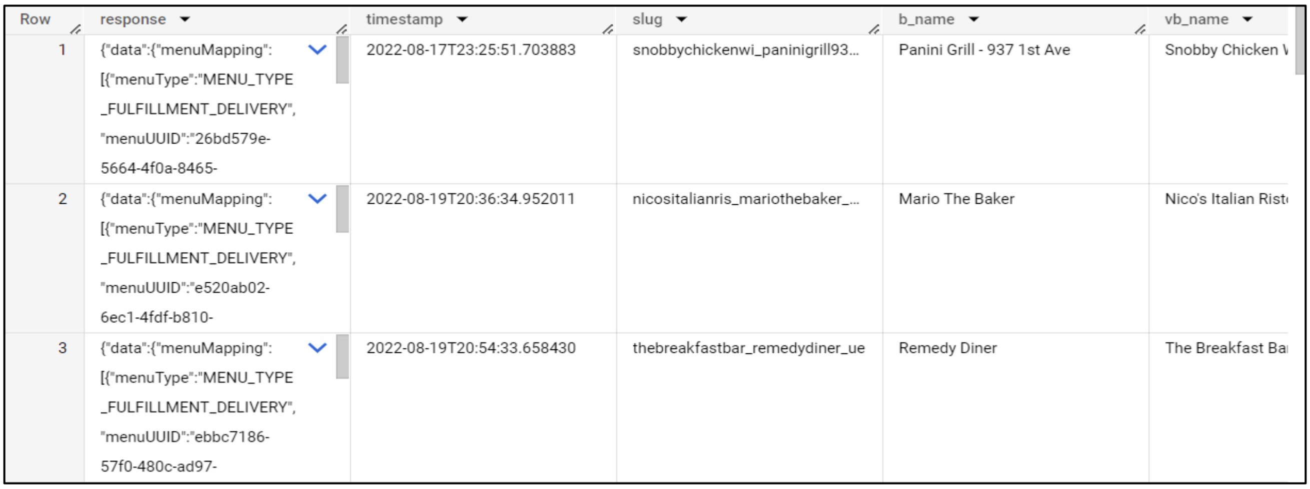
Uber Eats and GRUBHUB analysis

1. Running the first command to retrieve the data on Big Query to get data.

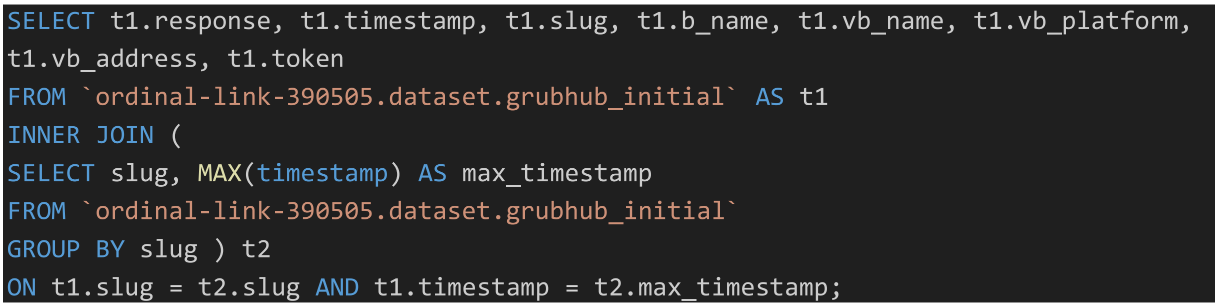


```
SELECT * FROM arboreal-vision-339901.take_home_v2.virtual_kitchen_ubereats_hours LIMIT
1000;
SELECT * FROM arboreal-vision-339901.take_home_v2.virtual_kitchen_grubhub_hours LIMIT
1000;
```

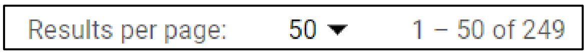


Row	response	timestamp	slug	b_name	vb_name
1	{"data":{"menuMapping": [{"menuType":"MENU_TYPE _FULFILLMENT_DELIVERY", "menuUUID":"26bd579e- 5664-4f0a-8465-	2022-08-17T23:25:51.703883	snobbychickenwi_paninigrill93...	Panini Grill - 937 1st Ave	Snobby Chicken W
2	{"data":{"menuMapping": [{"menuType":"MENU_TYPE _FULFILLMENT_DELIVERY", "menuUUID":"e520ab02- 6ec1-4fdf-b810-	2022-08-19T20:36:34.952011	nicositalianris_mariothebaker_...	Mario The Baker	Nico's Italian Rist
3	{"data":{"menuMapping": [{"menuType":"MENU_TYPE _FULFILLMENT_DELIVERY", "menuUUID":"ebbc7186- 57f0-480c-ad97-	2022-08-19T20:54:33.658430	thebreakfastbar_remedydiner_ue	Remedy Diner	The Breakfast Ba

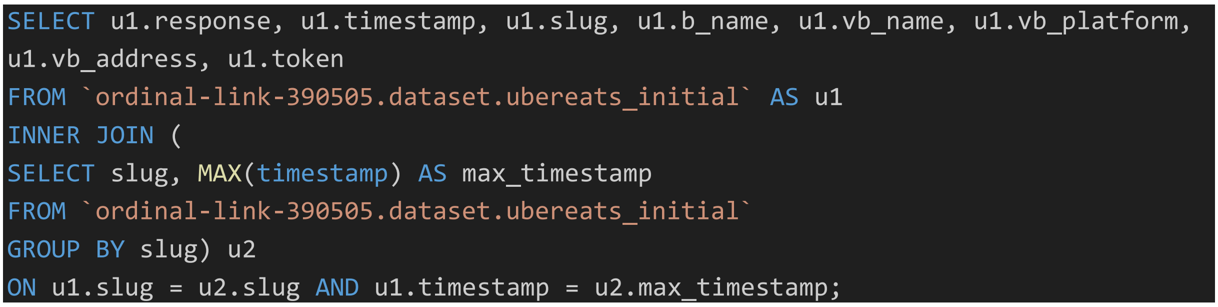
2. Deleting duplicate rows by keeping only the most recent timestamps.



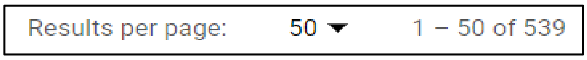
```
SELECT t1.response, t1.timestamp, t1.slug, t1.b_name, t1.vb_name, t1.vb_platform,
t1.vb_address, t1.token
FROM `ordinal-link-390505.dataset.grubhub_initial` AS t1
INNER JOIN (
SELECT slug, MAX(timestamp) AS max_timestamp
FROM `ordinal-link-390505.dataset.grubhub_initial`
GROUP BY slug ) t2
ON t1.slug = t2.slug AND t1.timestamp = t2.max_timestamp;
```



Results per page: 50 ▾ 1 – 50 of 249

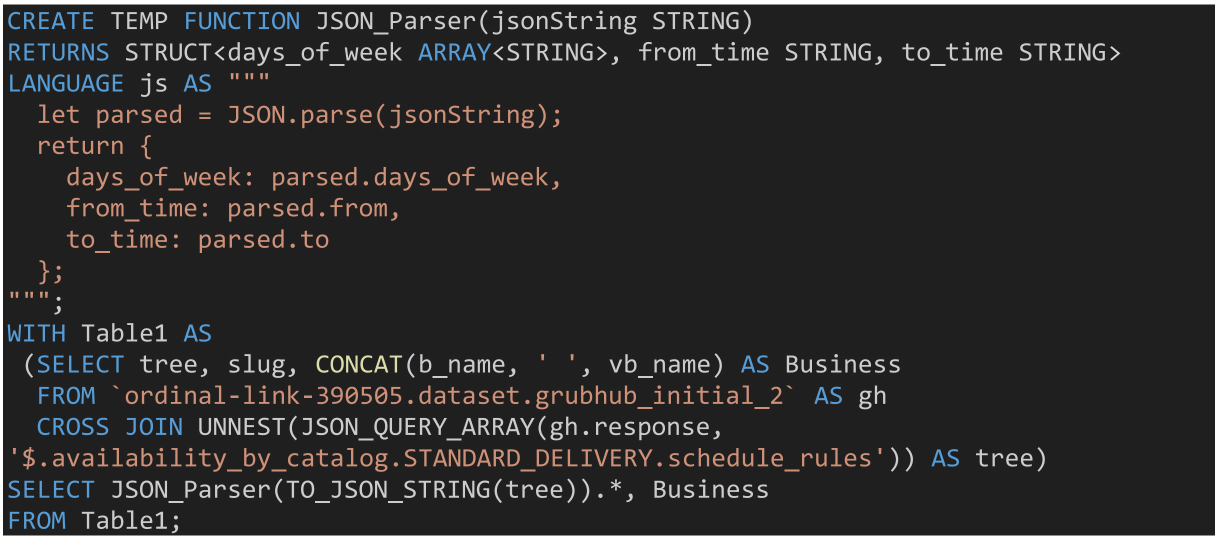


```
SELECT u1.response, u1.timestamp, u1.slug, u1.b_name, u1.vb_name, u1.vb_platform,
u1.vb_address, u1.token
FROM `ordinal-link-390505.dataset.ubereats_initial` AS u1
INNER JOIN (
SELECT slug, MAX(timestamp) AS max_timestamp
FROM `ordinal-link-390505.dataset.ubereats_initial`
GROUP BY slug) u2
ON u1.slug = u2.slug AND u1.timestamp = u2.max_timestamp;
```

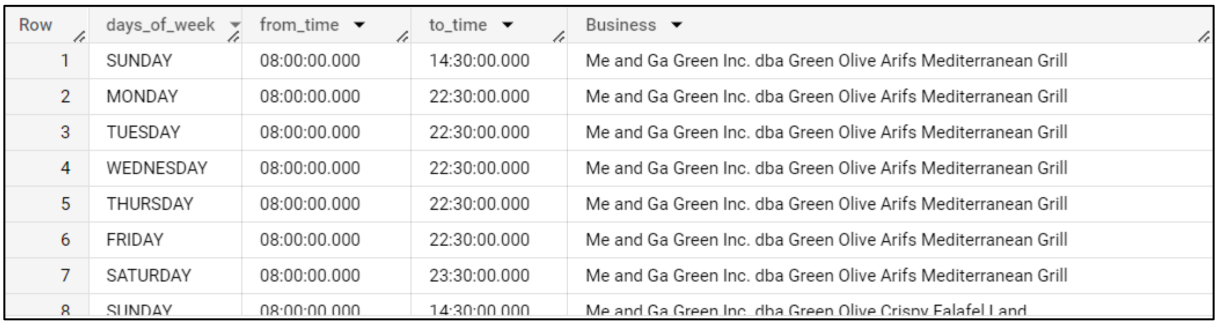


Results per page: 50 ▾ 1 – 50 of 539

3. Extracting the required JSON Data from the grubhub_initial_1 table.

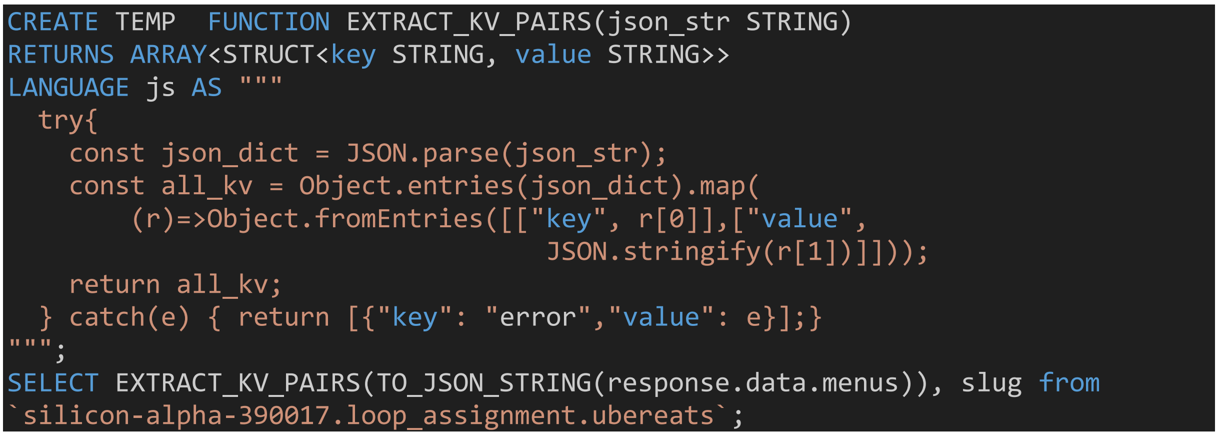


```
CREATE TEMP FUNCTION JSON_Parser(jsonString STRING)
RETURNS STRUCT<days_of_week ARRAY<STRING>, from_time STRING, to_time STRING>
LANGUAGE js AS """
  let parsed = JSON.parse(jsonString);
  return {
    days_of_week: parsed.days_of_week,
    from_time: parsed.from,
    to_time: parsed.to
  };
""";
WITH Table1 AS
 (SELECT tree, slug, CONCAT(b_name, ' ', vb_name) AS Business
  FROM `ordinal-link-390505.dataset.grubhub_initial_2` AS gh
  CROSS JOIN UNNEST(JSON_QUERY_ARRAY(gh.response,
'$.availability_by_catalog.STANDARD_DELIVERY.schedule_rules')) AS tree)
SELECT JSON_Parser(TO_JSON_STRING(tree)).*, Business
FROM Table1;
```

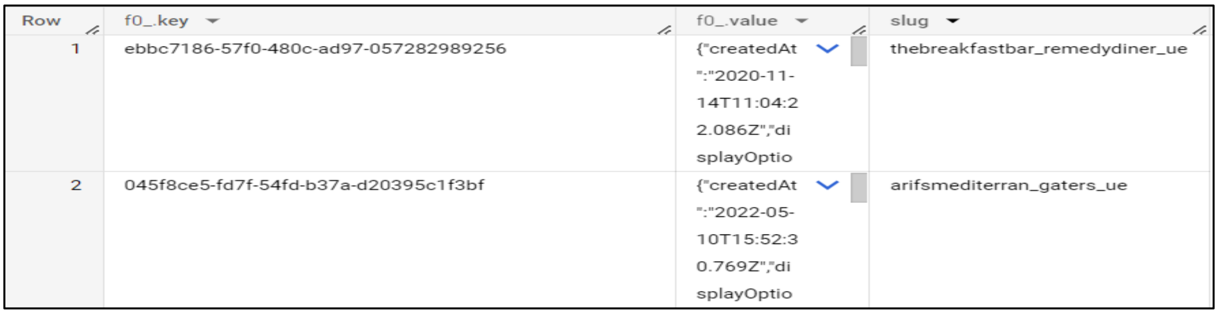


Row	days_of_week	from_time	to_time	Business
1	SUNDAY	08:00:00.000	14:30:00.000	Me and Ga Green Inc. dba Green Olive Arifs Mediterranean Grill
2	MONDAY	08:00:00.000	22:30:00.000	Me and Ga Green Inc. dba Green Olive Arifs Mediterranean Grill
3	TUESDAY	08:00:00.000	22:30:00.000	Me and Ga Green Inc. dba Green Olive Arifs Mediterranean Grill
4	WEDNESDAY	08:00:00.000	22:30:00.000	Me and Ga Green Inc. dba Green Olive Arifs Mediterranean Grill
5	THURSDAY	08:00:00.000	22:30:00.000	Me and Ga Green Inc. dba Green Olive Arifs Mediterranean Grill
6	FRIDAY	08:00:00.000	22:30:00.000	Me and Ga Green Inc. dba Green Olive Arifs Mediterranean Grill
7	SATURDAY	08:00:00.000	23:30:00.000	Me and Ga Green Inc. dba Green Olive Arifs Mediterranean Grill
8	SUNDAY	08:00:00.000	14:30:00.000	Me and Ga Green Inc. dba Green Olive Crispy Falafel Land

4. The JSON path for the UberEATS JSON object contains a variable. Since JSON path query cannot be dynamic, we will drill down on the node we require up until the variant.

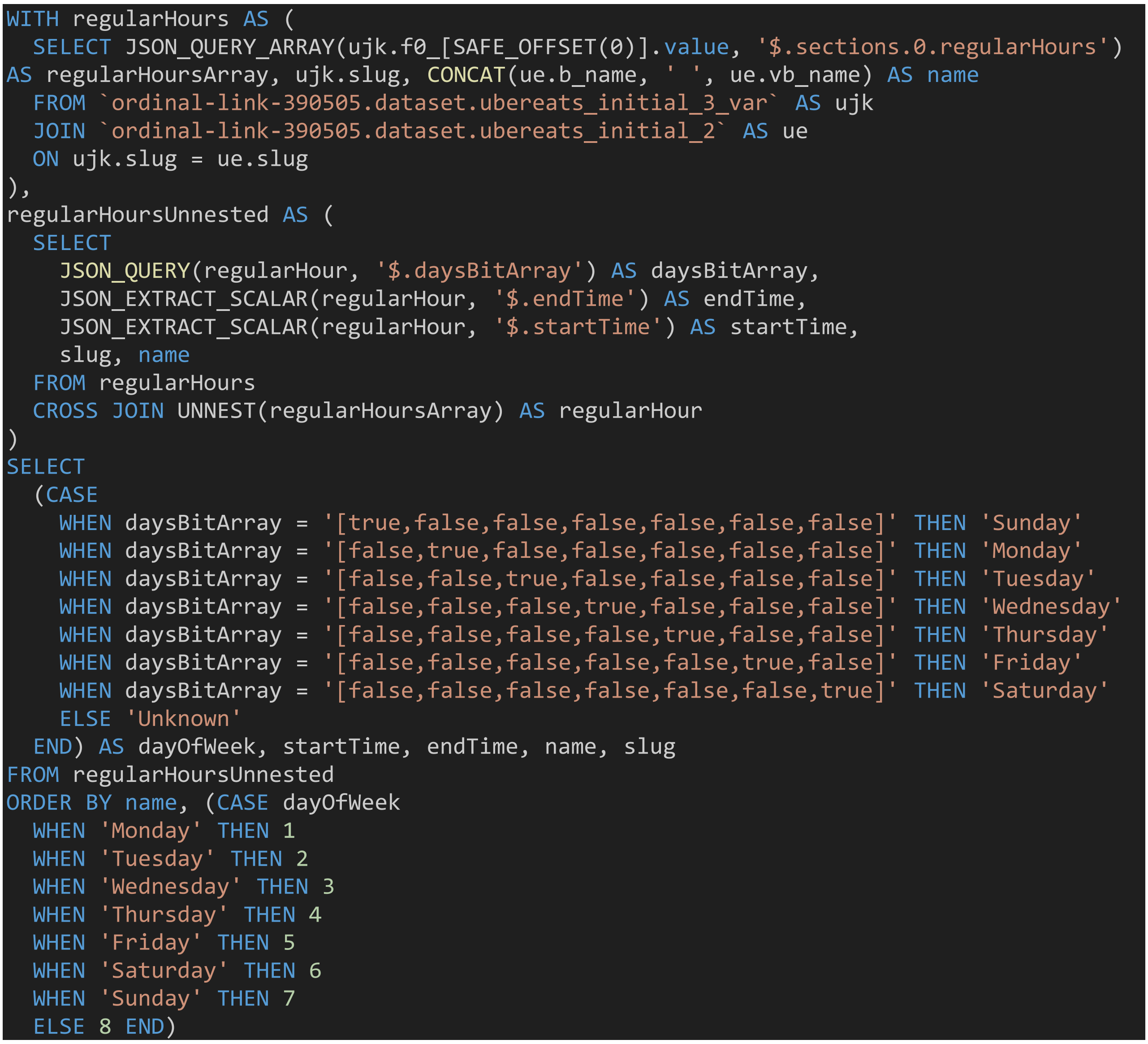


```
CREATE TEMP  FUNCTION EXTRACT_KV_PAIRS(json_str STRING)
RETURNS ARRAY<STRUCT<key STRING, value STRING>>
LANGUAGE js AS """
  try{
    const json_dict = JSON.parse(json_str);
    const all_kv = Object.entries(json_dict).map(
        (r)=>Object.fromEntries([["key", r[0]],["value",
                                JSON.stringify(r[1])]]));
    return all_kv;
  } catch(e) { return [{"key": "error","value": e}];}
""";
SELECT EXTRACT_KV_PAIRS(TO_JSON_STRING(response.data.menus)), slug from
`silicon-alpha-390017.loop_assignment.ubereats`;
```

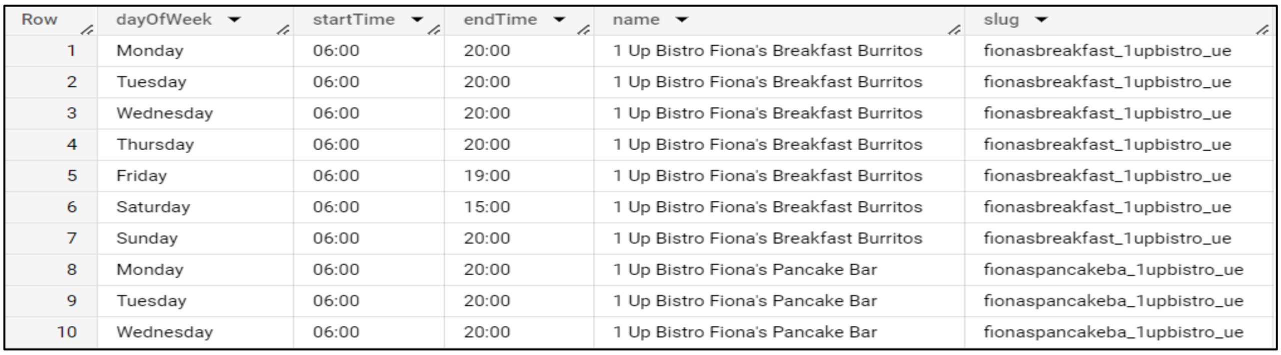


Row	f0_.key	f0_.value	slug
1	ebbc7186-57f0-480c-ad97-057282989256	{"createdAt":"2020-11-14T11:04:22.086Z","displayOptio	thebreakfastbar_remedydiner_ue
2	045f8ce5-fd7f-54fd-b37a-d20395c1f3bf	{"createdAt":"2022-05-10T15:52:30.769Z","displayOptio	arifsmediterran_gaters_ue

5. Extracting the day of the week, opening, and closing times of the restaurants registered with Ubereats.

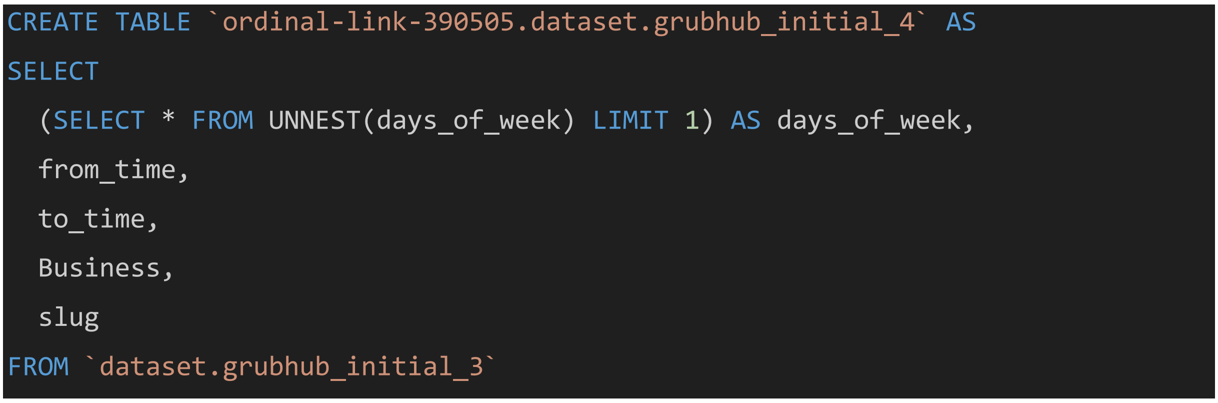


```sql
WITH regularHours AS (
  SELECT JSON_QUERY_ARRAY(ujk.f0_[SAFE_OFFSET(0)].value, '$.sections.0.regularHours')
AS regularHoursArray, ujk.slug, CONCAT(ue.b_name, ' ', ue.vb_name) AS name
  FROM `ordinal-link-390505.dataset.ubereats_initial_3_var` AS ujk
  JOIN `ordinal-link-390505.dataset.ubereats_initial_2` AS ue
  ON ujk.slug = ue.slug
),
regularHoursUnnested AS (
  SELECT
    JSON_QUERY(regularHour, '$.daysBitArray') AS daysBitArray,
    JSON_EXTRACT_SCALAR(regularHour, '$.endTime') AS endTime,
    JSON_EXTRACT_SCALAR(regularHour, '$.startTime') AS startTime,
    slug, name
  FROM regularHours
  CROSS JOIN UNNEST(regularHoursArray) AS regularHour
)
SELECT
  (CASE
    WHEN daysBitArray = '[true,false,false,false,false,false,false]' THEN 'Sunday'
    WHEN daysBitArray = '[false,true,false,false,false,false,false]' THEN 'Monday'
    WHEN daysBitArray = '[false,false,true,false,false,false,false]' THEN 'Tuesday'
    WHEN daysBitArray = '[false,false,false,true,false,false,false]' THEN 'Wednesday'
    WHEN daysBitArray = '[false,false,false,false,true,false,false]' THEN 'Thursday'
    WHEN daysBitArray = '[false,false,false,false,false,true,false]' THEN 'Friday'
    WHEN daysBitArray = '[false,false,false,false,false,false,true]' THEN 'Saturday'
    ELSE 'Unknown'
  END) AS dayOfWeek, startTime, endTime, name, slug
FROM regularHoursUnnested
ORDER BY name, (CASE dayOfWeek
  WHEN 'Monday' THEN 1
  WHEN 'Tuesday' THEN 2
  WHEN 'Wednesday' THEN 3
  WHEN 'Thursday' THEN 4
  WHEN 'Friday' THEN 5
  WHEN 'Saturday' THEN 6
  WHEN 'Sunday' THEN 7
  ELSE 8 END)
```

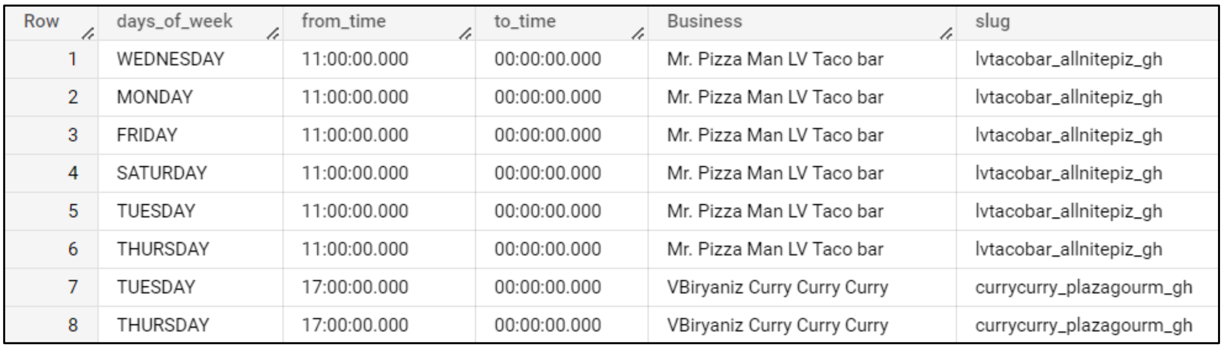


Row	dayOfWeek	startTime	endTime	name	slug
1	Monday	06:00	20:00	1 Up Bistro Fiona's Breakfast Burritos	fionasbreakfast_1upbistro_ue
2	Tuesday	06:00	20:00	1 Up Bistro Fiona's Breakfast Burritos	fionasbreakfast_1upbistro_ue
3	Wednesday	06:00	20:00	1 Up Bistro Fiona's Breakfast Burritos	fionasbreakfast_1upbistro_ue
4	Thursday	06:00	20:00	1 Up Bistro Fiona's Breakfast Burritos	fionasbreakfast_1upbistro_ue
5	Friday	06:00	19:00	1 Up Bistro Fiona's Breakfast Burritos	fionasbreakfast_1upbistro_ue
6	Saturday	06:00	15:00	1 Up Bistro Fiona's Breakfast Burritos	fionasbreakfast_1upbistro_ue
7	Sunday	06:00	20:00	1 Up Bistro Fiona's Breakfast Burritos	fionasbreakfast_1upbistro_ue
8	Monday	06:00	20:00	1 Up Bistro Fiona's Pancake Bar	fionaspancakeba_1upbistro_ue
9	Tuesday	06:00	20:00	1 Up Bistro Fiona's Pancake Bar	fionaspancakeba_1upbistro_ue
10	Wednesday	06:00	20:00	1 Up Bistro Fiona's Pancake Bar	fionaspancakeba_1upbistro_ue

6. Unnesting the day of the week column in the Grubhub table so it can be analyzed further.

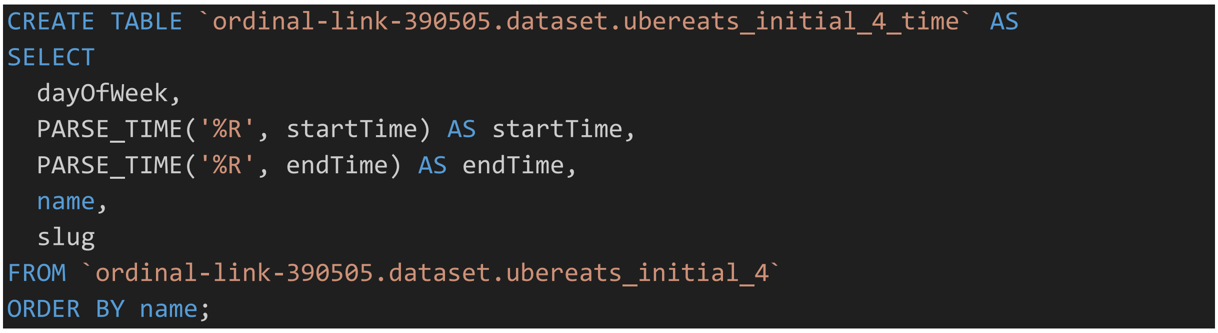


```
CREATE TABLE `ordinal-link-390505.dataset.grubhub_initial_4` AS
SELECT
  (SELECT * FROM UNNEST(days_of_week) LIMIT 1) AS days_of_week,
  from_time,
  to_time,
  Business,
  slug
FROM `dataset.grubhub_initial_3`
```

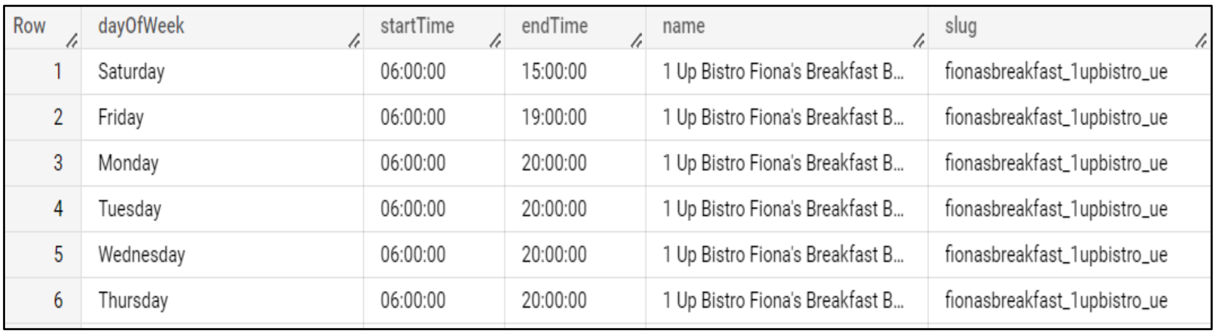


Row	days_of_week	from_time	to_time	Business	slug
1	WEDNESDAY	11:00:00.000	00:00:00.000	Mr. Pizza Man LV Taco bar	lvtacobar_allnitepiz_gh
2	MONDAY	11:00:00.000	00:00:00.000	Mr. Pizza Man LV Taco bar	lvtacobar_allnitepiz_gh
3	FRIDAY	11:00:00.000	00:00:00.000	Mr. Pizza Man LV Taco bar	lvtacobar_allnitepiz_gh
4	SATURDAY	11:00:00.000	00:00:00.000	Mr. Pizza Man LV Taco bar	lvtacobar_allnitepiz_gh
5	TUESDAY	11:00:00.000	00:00:00.000	Mr. Pizza Man LV Taco bar	lvtacobar_allnitepiz_gh
6	THURSDAY	11:00:00.000	00:00:00.000	Mr. Pizza Man LV Taco bar	lvtacobar_allnitepiz_gh
7	TUESDAY	17:00:00.000	00:00:00.000	VBiryaniz Curry Curry Curry	currycurry_plazagourm_gh
8	THURSDAY	17:00:00.000	00:00:00.000	VBiryaniz Curry Curry Curry	currycurry_plazagourm_gh

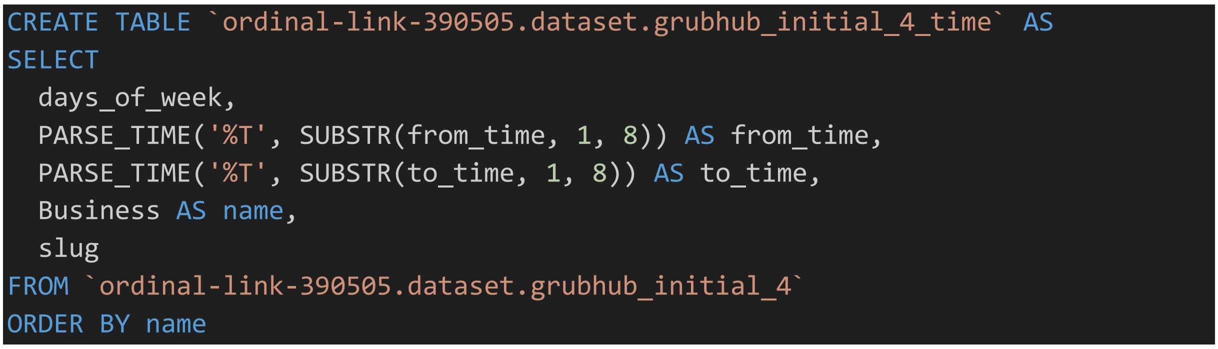
7. Converting the startTime and endTime columns in the Ubereats and Grubhub table from dtype string to dtype time and creating a new table.



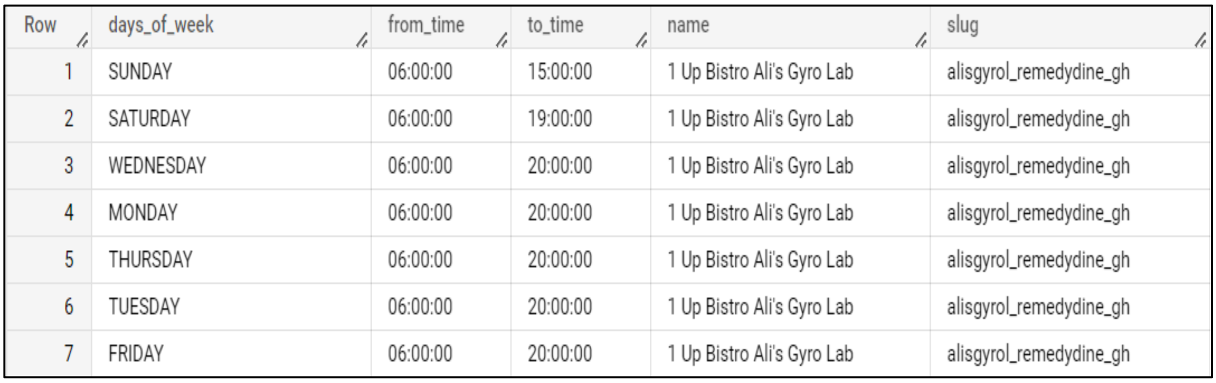
```sql
CREATE TABLE `ordinal-link-390505.dataset.ubereats_initial_4_time` AS
SELECT
  dayOfWeek,
  PARSE_TIME('%R', startTime) AS startTime,
  PARSE_TIME('%R', endTime) AS endTime,
  name,
  slug
FROM `ordinal-link-390505.dataset.ubereats_initial_4`
ORDER BY name;
```



Row	dayOfWeek	startTime	endTime	name	slug
1	Saturday	06:00:00	15:00:00	1 Up Bistro Fiona's Breakfast B...	fionasbreakfast_1upbistro_ue
2	Friday	06:00:00	19:00:00	1 Up Bistro Fiona's Breakfast B...	fionasbreakfast_1upbistro_ue
3	Monday	06:00:00	20:00:00	1 Up Bistro Fiona's Breakfast B...	fionasbreakfast_1upbistro_ue
4	Tuesday	06:00:00	20:00:00	1 Up Bistro Fiona's Breakfast B...	fionasbreakfast_1upbistro_ue
5	Wednesday	06:00:00	20:00:00	1 Up Bistro Fiona's Breakfast B...	fionasbreakfast_1upbistro_ue
6	Thursday	06:00:00	20:00:00	1 Up Bistro Fiona's Breakfast B...	fionasbreakfast_1upbistro_ue

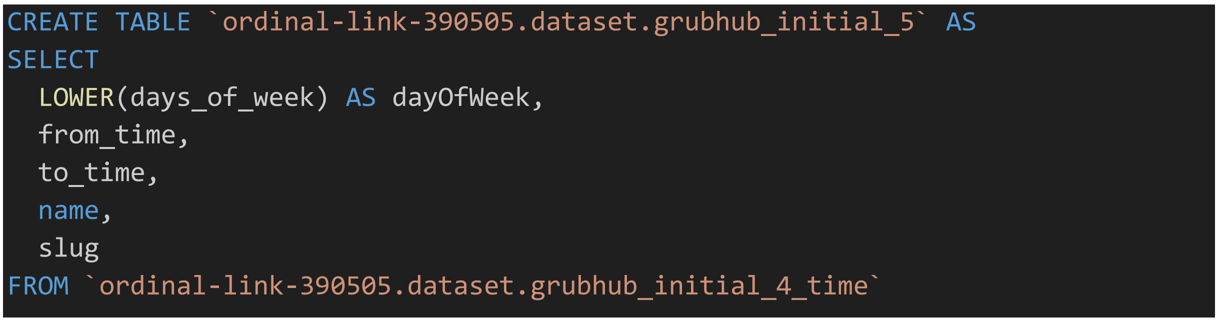


```sql
CREATE TABLE `ordinal-link-390505.dataset.grubhub_initial_4_time` AS
SELECT
  days_of_week,
  PARSE_TIME('%T', SUBSTR(from_time, 1, 8)) AS from_time,
  PARSE_TIME('%T', SUBSTR(to_time, 1, 8)) AS to_time,
  Business AS name,
  slug
FROM `ordinal-link-390505.dataset.grubhub_initial_4`
ORDER BY name
```

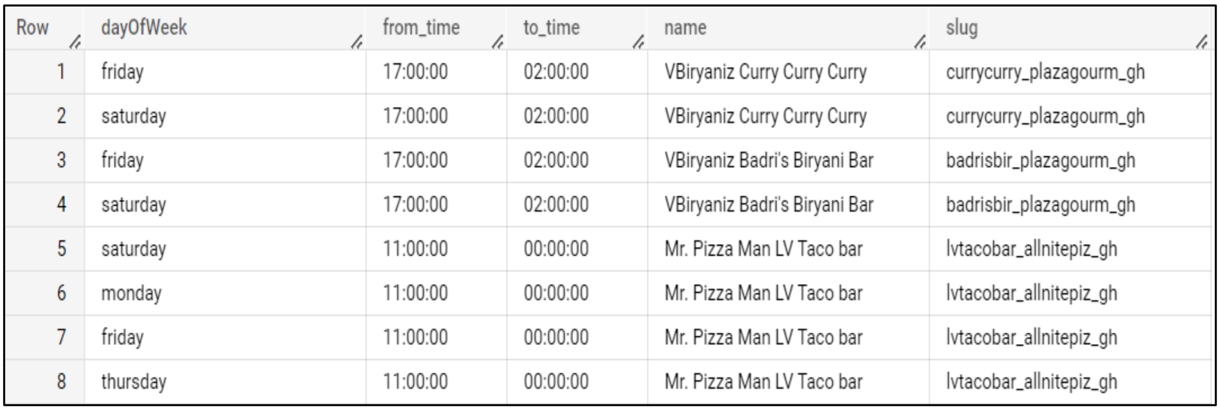


Row	days_of_week	from_time	to_time	name	slug
1	SUNDAY	06:00:00	15:00:00	1 Up Bistro Ali's Gyro Lab	alisgyrol_remedydine_gh
2	SATURDAY	06:00:00	19:00:00	1 Up Bistro Ali's Gyro Lab	alisgyrol_remedydine_gh
3	WEDNESDAY	06:00:00	20:00:00	1 Up Bistro Ali's Gyro Lab	alisgyrol_remedydine_gh
4	MONDAY	06:00:00	20:00:00	1 Up Bistro Ali's Gyro Lab	alisgyrol_remedydine_gh
5	THURSDAY	06:00:00	20:00:00	1 Up Bistro Ali's Gyro Lab	alisgyrol_remedydine_gh
6	TUESDAY	06:00:00	20:00:00	1 Up Bistro Ali's Gyro Lab	alisgyrol_remedydine_gh
7	FRIDAY	06:00:00	20:00:00	1 Up Bistro Ali's Gyro Lab	alisgyrol_remedydine_gh

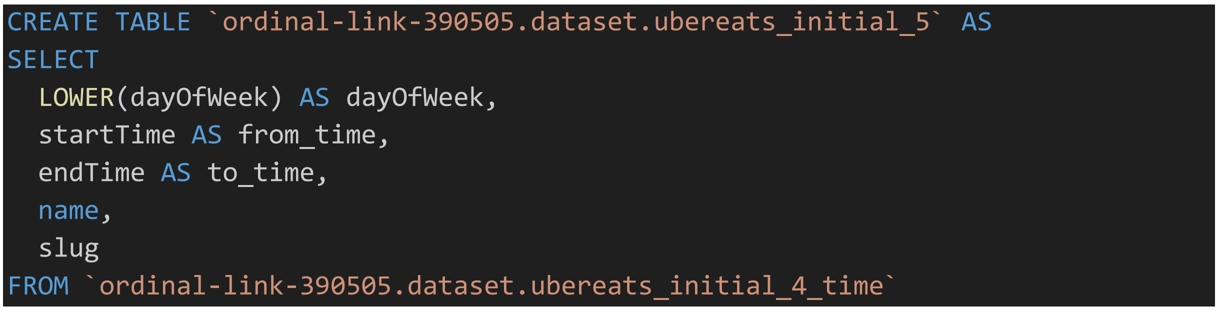
8. Converting day of week column in Ubereats and Grubhub tables to lowercase.



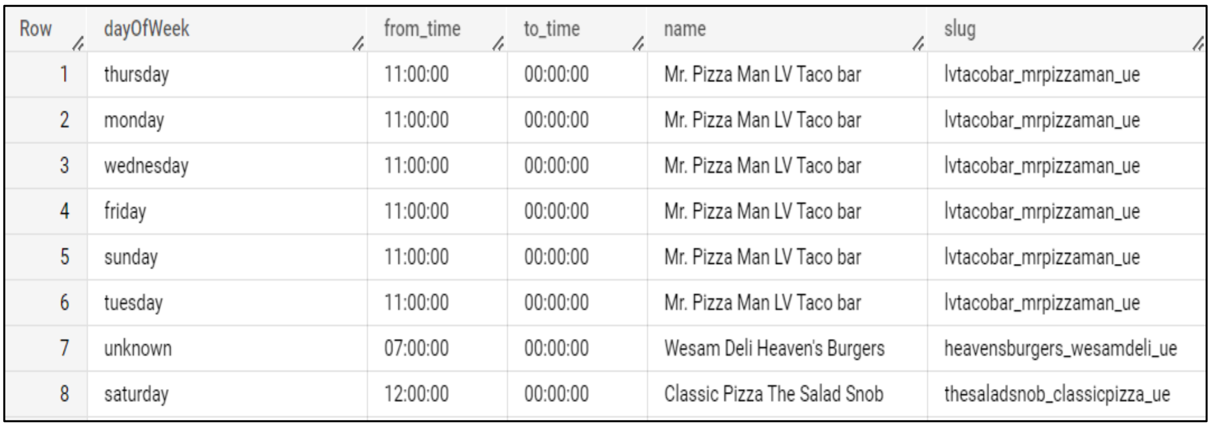
```
CREATE TABLE `ordinal-link-390505.dataset.grubhub_initial_5` AS
SELECT
  LOWER(days_of_week) AS dayOfWeek,
  from_time,
  to_time,
  name,
  slug
FROM `ordinal-link-390505.dataset.grubhub_initial_4_time`
```



Row	dayOfWeek	from_time	to_time	name	slug
1	friday	17:00:00	02:00:00	VBiryaniz Curry Curry Curry	currycurry_plazagourm_gh
2	saturday	17:00:00	02:00:00	VBiryaniz Curry Curry Curry	currycurry_plazagourm_gh
3	friday	17:00:00	02:00:00	VBiryaniz Badri's Biryani Bar	badrisbir_plazagourm_gh
4	saturday	17:00:00	02:00:00	VBiryaniz Badri's Biryani Bar	badrisbir_plazagourm_gh
5	saturday	11:00:00	00:00:00	Mr. Pizza Man LV Taco bar	lvtacobar_allnitepiz_gh
6	monday	11:00:00	00:00:00	Mr. Pizza Man LV Taco bar	lvtacobar_allnitepiz_gh
7	friday	11:00:00	00:00:00	Mr. Pizza Man LV Taco bar	lvtacobar_allnitepiz_gh
8	thursday	11:00:00	00:00:00	Mr. Pizza Man LV Taco bar	lvtacobar_allnitepiz_gh

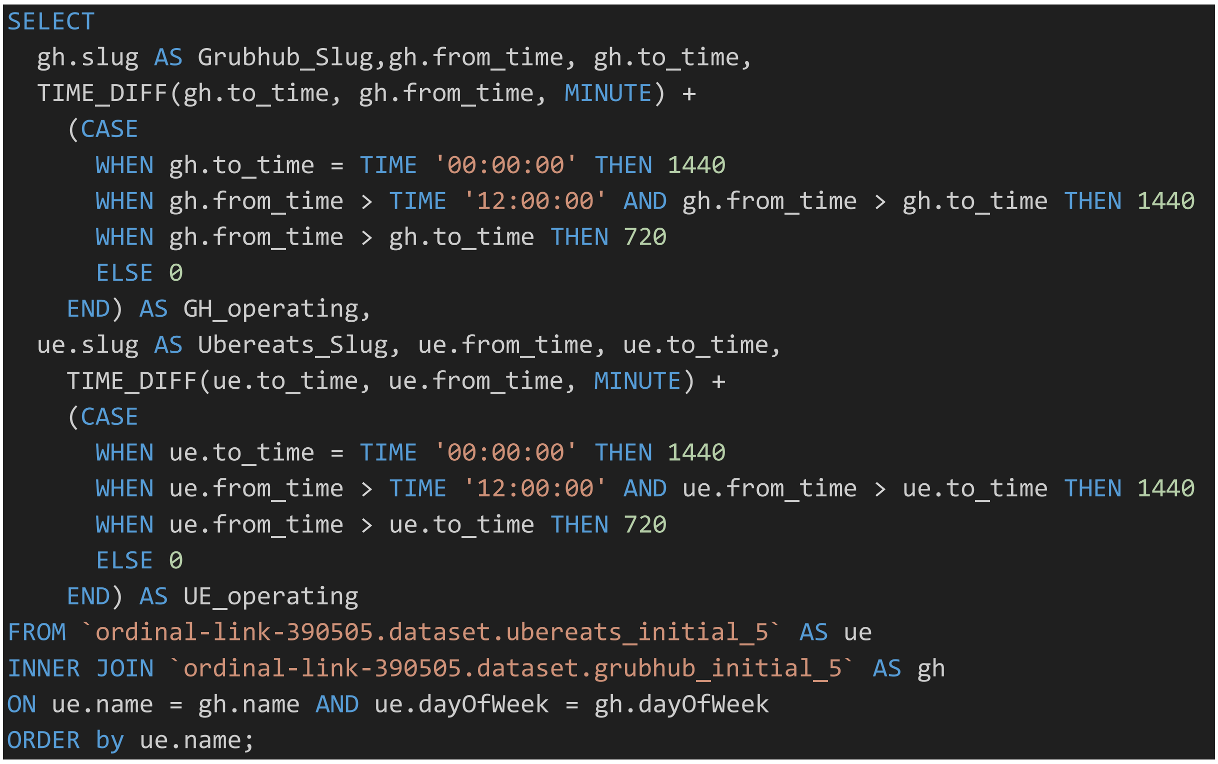


```
CREATE TABLE `ordinal-link-390505.dataset.ubereats_initial_5` AS
SELECT
  LOWER(dayOfWeek) AS dayOfWeek,
  startTime AS from_time,
  endTime AS to_time,
  name,
  slug
FROM `ordinal-link-390505.dataset.ubereats_initial_4_time`
```

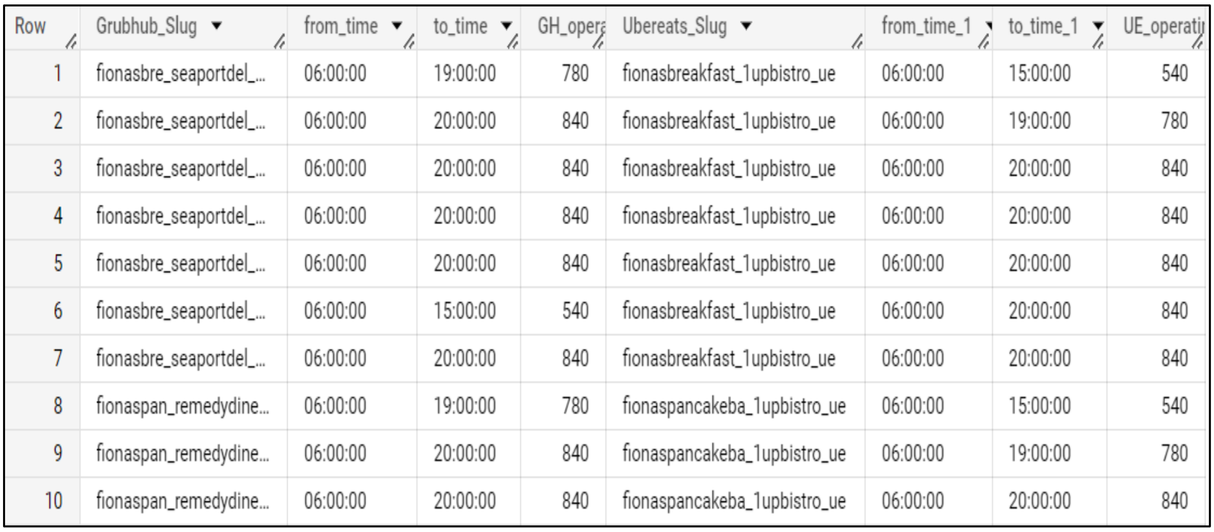


Row	dayOfWeek	from_time	to_time	name	slug
1	thursday	11:00:00	00:00:00	Mr. Pizza Man LV Taco bar	lvtacobar_mrpizzaman_ue
2	monday	11:00:00	00:00:00	Mr. Pizza Man LV Taco bar	lvtacobar_mrpizzaman_ue
3	wednesday	11:00:00	00:00:00	Mr. Pizza Man LV Taco bar	lvtacobar_mrpizzaman_ue
4	friday	11:00:00	00:00:00	Mr. Pizza Man LV Taco bar	lvtacobar_mrpizzaman_ue
5	sunday	11:00:00	00:00:00	Mr. Pizza Man LV Taco bar	lvtacobar_mrpizzaman_ue
6	tuesday	11:00:00	00:00:00	Mr. Pizza Man LV Taco bar	lvtacobar_mrpizzaman_ue
7	unknown	07:00:00	00:00:00	Wesam Deli Heaven's Burgers	heavensburgers_wesamdeli_ue
8	saturday	12:00:00	00:00:00	Classic Pizza The Salad Snob	thesaladsnob_classicpizza_ue

9. Creating a table with matched Business names and a time difference between the opening and closing times calculated in minutes.

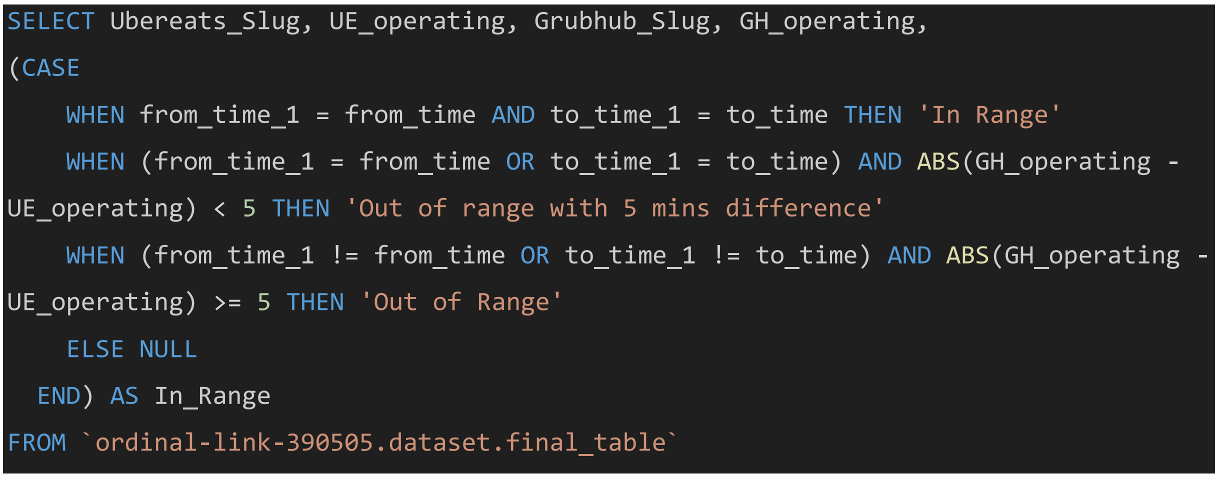


```
SELECT
  gh.slug AS Grubhub_Slug,gh.from_time, gh.to_time,
  TIME_DIFF(gh.to_time, gh.from_time, MINUTE) +
    (CASE
      WHEN gh.to_time = TIME '00:00:00' THEN 1440
      WHEN gh.from_time > TIME '12:00:00' AND gh.from_time > gh.to_time THEN 1440
      WHEN gh.from_time > gh.to_time THEN 720
      ELSE 0
    END) AS GH_operating,
  ue.slug AS Ubereats_Slug, ue.from_time, ue.to_time,
    TIME_DIFF(ue.to_time, ue.from_time, MINUTE) +
    (CASE
      WHEN ue.to_time = TIME '00:00:00' THEN 1440
      WHEN ue.from_time > TIME '12:00:00' AND ue.from_time > ue.to_time THEN 1440
      WHEN ue.from_time > ue.to_time THEN 720
      ELSE 0
    END) AS UE_operating
FROM `ordinal-link-390505.dataset.ubereats_initial_5` AS ue
INNER JOIN `ordinal-link-390505.dataset.grubhub_initial_5` AS gh
ON ue.name = gh.name AND ue.dayOfWeek = gh.dayOfWeek
ORDER by ue.name;
```

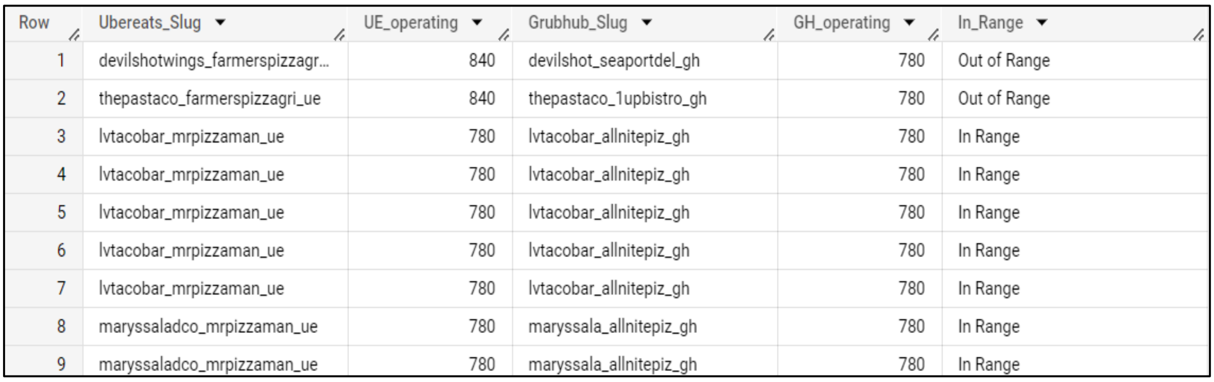


Row	Grubhub_Slug	from_time	to_time	GH_opera	Ubereats_Slug	from_time_1	to_time_1	UE_operati
1	fionasbre_seaportdel_...	06:00:00	19:00:00	780	fionasbreakfast_1upbistro_ue	06:00:00	15:00:00	540
2	fionasbre_seaportdel_...	06:00:00	20:00:00	840	fionasbreakfast_1upbistro_ue	06:00:00	19:00:00	780
3	fionasbre_seaportdel_...	06:00:00	20:00:00	840	fionasbreakfast_1upbistro_ue	06:00:00	20:00:00	840
4	fionasbre_seaportdel_...	06:00:00	20:00:00	840	fionasbreakfast_1upbistro_ue	06:00:00	20:00:00	840
5	fionasbre_seaportdel_...	06:00:00	20:00:00	840	fionasbreakfast_1upbistro_ue	06:00:00	20:00:00	840
6	fionasbre_seaportdel_...	06:00:00	15:00:00	540	fionasbreakfast_1upbistro_ue	06:00:00	20:00:00	840
7	fionasbre_seaportdel_...	06:00:00	20:00:00	840	fionasbreakfast_1upbistro_ue	06:00:00	20:00:00	840
8	fionaspan_remedydine...	06:00:00	19:00:00	780	fionaspancakeba_1upbistro_ue	06:00:00	15:00:00	540
9	fionaspan_remedydine...	06:00:00	20:00:00	840	fionaspancakeba_1upbistro_ue	06:00:00	19:00:00	780
10	fionaspan_remedydine...	06:00:00	20:00:00	840	fionaspancakeba_1upbistro_ue	06:00:00	20:00:00	840

10. Creating the final table in accordance with the desired output of the assignment.



```
SELECT Ubereats_Slug, UE_operating, Grubhub_Slug, GH_operating,
(CASE
    WHEN from_time_1 = from_time AND to_time_1 = to_time THEN 'In Range'
    WHEN (from_time_1 = from_time OR to_time_1 = to_time) AND ABS(GH_operating -
UE_operating) < 5 THEN 'Out of range with 5 mins difference'
    WHEN (from_time_1 != from_time OR to_time_1 != to_time) AND ABS(GH_operating -
UE_operating) >= 5 THEN 'Out of Range'
    ELSE NULL
  END) AS In_Range
FROM `ordinal-link-390505.dataset.final_table`
```



Row	Ubereats_Slug	UE_operating	Grubhub_Slug	GH_operating	In_Range
1	devilshotwings_farmerspizzagr...	840	devilshot_seaportdel_gh	780	Out of Range
2	thepastaco_farmerspizzagri_ue	840	thepastaco_1upbistro_gh	780	Out of Range
3	lvtacobar_mrpizzaman_ue	780	lvtacobar_allnitepiz_gh	780	In Range
4	lvtacobar_mrpizzaman_ue	780	lvtacobar_allnitepiz_gh	780	In Range
5	lvtacobar_mrpizzaman_ue	780	lvtacobar_allnitepiz_gh	780	In Range
6	lvtacobar_mrpizzaman_ue	780	lvtacobar_allnitepiz_gh	780	In Range
7	lvtacobar_mrpizzaman_ue	780	lvtacobar_allnitepiz_gh	780	In Range
8	maryssaladco_mrpizzaman_ue	780	maryssala_allnitepiz_gh	780	In Range
9	maryssaladco_mrpizzaman_ue	780	maryssala_allnitepiz_gh	780	In Range

Notes: -

Some businesses in the Uber Eats dataset had timings for days labelled ‘Unknown’. Such rows were dropped automatically when Inner Joining. Similarly, some businesses had multiple entries for the same days of the week. In this case, the first entry for the weekday was selected.