Where Every Slice is a Taste of Perfection

SQL PROJECT: PIZZA SALES

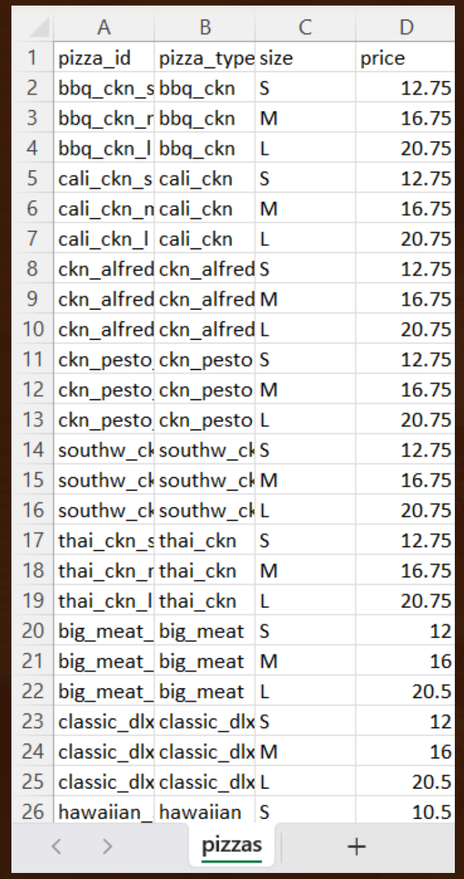
PROJECT APPROACH

Hello, I’m Sanika Ambre. I’ll be sharing my SQL project, where I analyzed pizza sales data to extract meaningful insights.

To achieve this, I:

- **Implemented various SQL queries to filter, sort, and aggregate the data.**
- **Utilized key SQL functions like JOIN, GROUP BY, and ORDER BY to answer business-relevant questions.**
- **Focused on identifying trends, popular pizza types, and sales performance.**

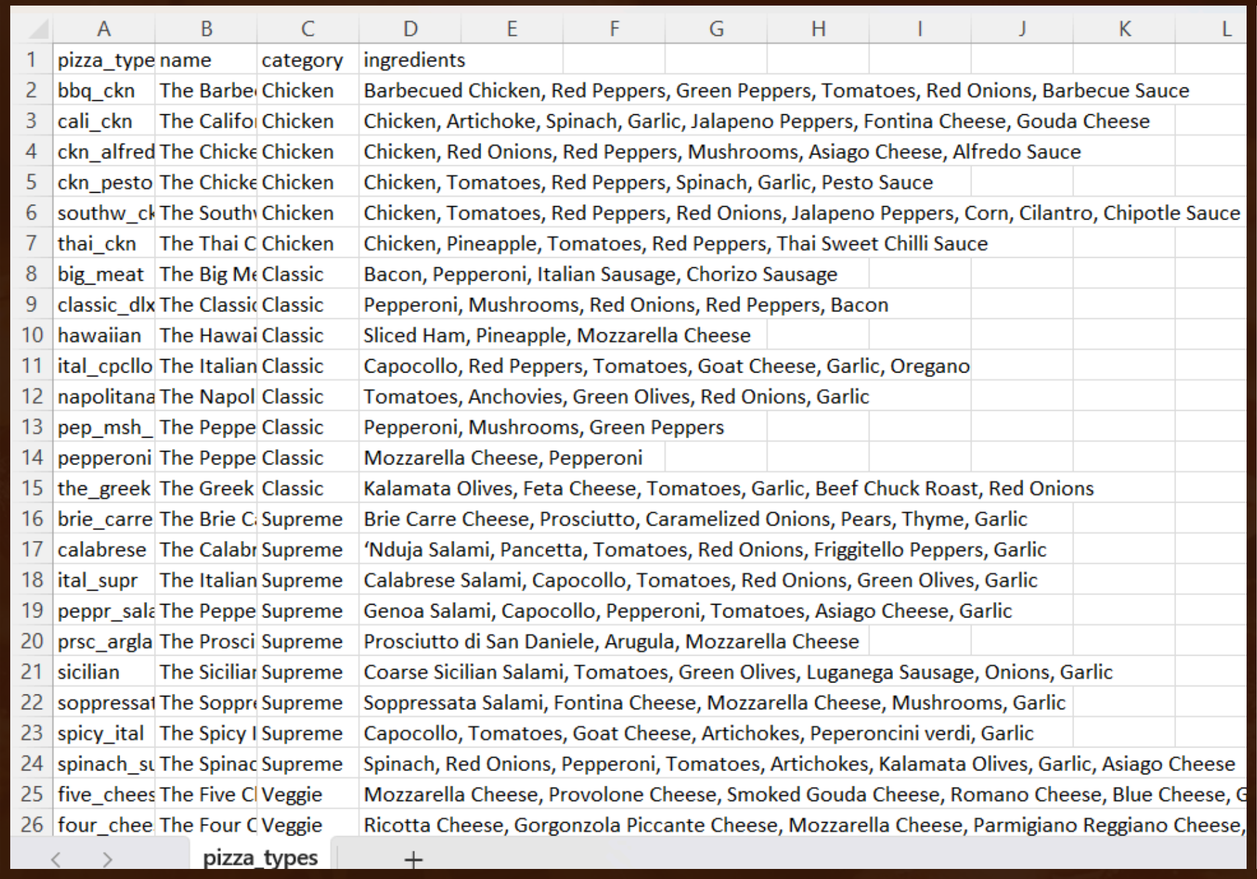
DATASETS



	A	B	C	D
1	pizza_id	pizza_type	size	price
2	bbq_ckn_s	bbq_ckn	S	12.75
3	bbq_ckn_r	bbq_ckn	M	16.75
4	bbq_ckn_l	bbq_ckn	L	20.75
5	cali_ckn_s	cali_ckn	S	12.75
6	cali_ckn_n	cali_ckn	M	16.75
7	cali_ckn_l	cali_ckn	L	20.75
8	ckn_alfred	ckn_alfred	S	12.75
9	ckn_alfred	ckn_alfred	M	16.75
10	ckn_alfred	ckn_alfred	L	20.75
11	ckn_pesto	ckn_pesto	S	12.75
12	ckn_pesto	ckn_pesto	M	16.75
13	ckn_pesto	ckn_pesto	L	20.75
14	southw_ck	southw_ck	S	12.75
15	southw_ck	southw_ck	M	16.75
16	southw_ck	southw_ck	L	20.75
17	thai_ckn_s	thai_ckn	S	12.75
18	thai_ckn_r	thai_ckn	M	16.75
19	thai_ckn_l	thai_ckn	L	20.75
20	big_meat_	big_meat	S	12
21	big_meat_	big_meat	M	16
22	big_meat_	big_meat	L	20.5
23	classic_dlx	classic_dlx	S	12
24	classic_dlx	classic_dlx	M	16
25	classic_dlx	classic_dlx	L	20.5
26	hawaiian_	hawaiian	S	10.5

pizzas

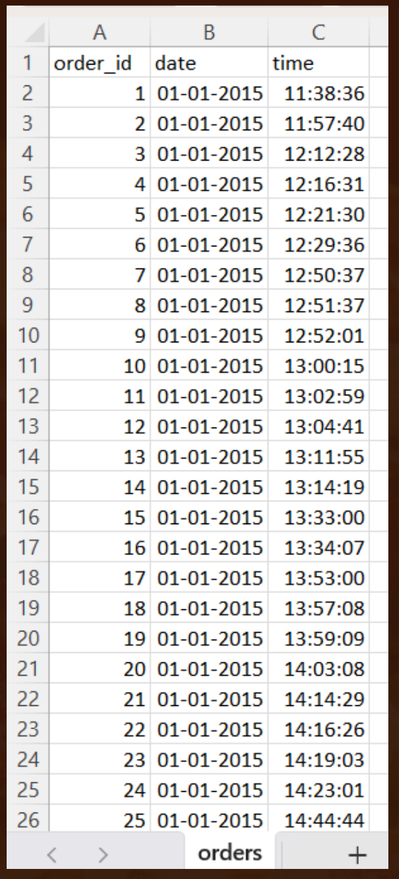
Pizzas



	A	B	C	D
1	pizza_type	name	category	ingredients
2	bbq_ckn	The Barbe	Chicken	Barbecued Chicken, Red Peppers, Green Peppers, Tomatoes, Red Onions, Barbecue Sauce
3	cali_ckn	The Califo	Chicken	Chicken, Artichoke, Spinach, Garlic, Jalapeno Peppers, Fontina Cheese, Gouda Cheese
4	ckn_alfred	The Chicke	Chicken	Chicken, Red Onions, Red Peppers, Mushrooms, Asiago Cheese, Alfredo Sauce
5	ckn_pesto	The Chicke	Chicken	Chicken, Tomatoes, Red Peppers, Spinach, Garlic, Pesto Sauce
6	southw_ck	The South	Chicken	Chicken, Tomatoes, Red Peppers, Red Onions, Jalapeno Peppers, Corn, Cilantro, Chipotle Sauce
7	thai_ckn	The Thai C	Chicken	Chicken, Pineapple, Tomatoes, Red Peppers, Thai Sweet Chilli Sauce
8	big_meat	The Big Me	Classic	Bacon, Pepperoni, Italian Sausage, Chorizo Sausage
9	classic_dlx	The Classic	Classic	Pepperoni, Mushrooms, Red Onions, Red Peppers, Bacon
10	hawaiian	The Hawai	Classic	Sliced Ham, Pineapple, Mozzarella Cheese
11	ital_cpcllo	The Italian	Classic	Capocollo, Red Peppers, Tomatoes, Goat Cheese, Garlic, Oregano
12	napolitana	The Napol	Classic	Tomatoes, Anchovies, Green Olives, Red Onions, Garlic
13	pep_msh_	The Peppe	Classic	Pepperoni, Mushrooms, Green Peppers
14	pepperoni	The Peppe	Classic	Mozzarella Cheese, Pepperoni
15	the_greek	The Greek	Classic	Kalamata Olives, Feta Cheese, Tomatoes, Garlic, Beef Chuck Roast, Red Onions
16	brie_carre	The Brie C	Supreme	Brie Carre Cheese, Prosciutto, Caramelized Onions, Pears, Thyme, Garlic
17	calabrese	The Calabr	Supreme	'Nduja Salami, Pancetta, Tomatoes, Red Onions, Friggitello Peppers, Garlic
18	ital_supr	The Italian	Supreme	Calabrese Salami, Capocollo, Tomatoes, Red Onions, Green Olives, Garlic
19	peppr_sala	The Peppe	Supreme	Genoa Salami, Capocollo, Pepperoni, Tomatoes, Asiago Cheese, Garlic
20	prsc_argla	The Prosci	Supreme	Prosciutto di San Daniele, Arugula, Mozzarella Cheese
21	sicilian	The Sicilia	Supreme	Coarse Sicilian Salami, Tomatoes, Green Olives, Luganega Sausage, Onions, Garlic
22	soppressat	The Soppr	Supreme	Soppressata Salami, Fontina Cheese, Mozzarella Cheese, Mushrooms, Garlic
23	spicy_ital	The Spicy I	Supreme	Capocollo, Tomatoes, Goat Cheese, Artichokes, Peperoncini verdi, Garlic
24	spinach_su	The Spinac	Supreme	Spinach, Red Onions, Pepperoni, Tomatoes, Artichokes, Kalamata Olives, Garlic, Asiago Cheese
25	five_chees	The Five C	Veggie	Mozzarella Cheese, Provolone Cheese, Smoked Gouda Cheese, Romano Cheese, Blue Cheese, G
26	four_chee	The Four C	Veggie	Ricotta Cheese, Gorgonzola Piccante Cheese, Mozzarella Cheese, Parmigiano Reggiano Cheese,

pizza_types

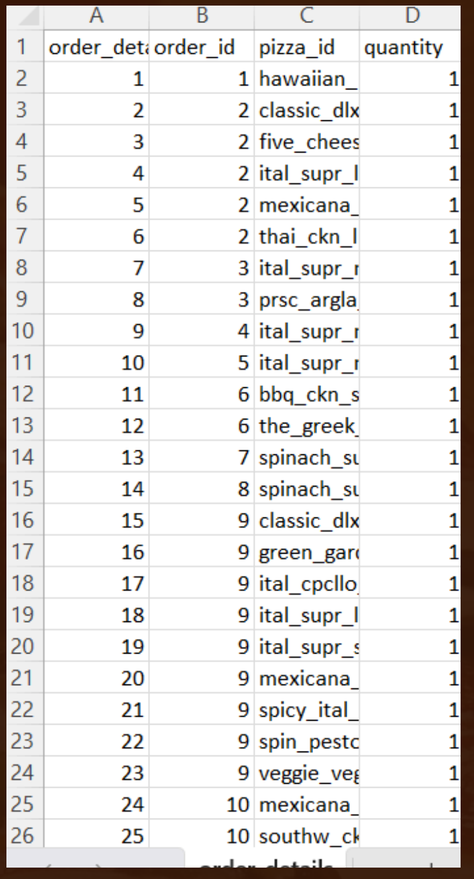
Pizza_types



	A	B	C
1	order_id	date	time
2	1	01-01-2015	11:38:36
3	2	01-01-2015	11:57:40
4	3	01-01-2015	12:12:28
5	4	01-01-2015	12:16:31
6	5	01-01-2015	12:21:30
7	6	01-01-2015	12:29:36
8	7	01-01-2015	12:50:37
9	8	01-01-2015	12:51:37
10	9	01-01-2015	12:52:01
11	10	01-01-2015	13:00:15
12	11	01-01-2015	13:02:59
13	12	01-01-2015	13:04:41
14	13	01-01-2015	13:11:55
15	14	01-01-2015	13:14:19
16	15	01-01-2015	13:33:00
17	16	01-01-2015	13:34:07
18	17	01-01-2015	13:53:00
19	18	01-01-2015	13:57:08
20	19	01-01-2015	13:59:09
21	20	01-01-2015	14:03:08
22	21	01-01-2015	14:14:29
23	22	01-01-2015	14:16:26
24	23	01-01-2015	14:19:03
25	24	01-01-2015	14:23:01
26	25	01-01-2015	14:44:44

orders

Orders



	A	B	C	D
1	order_deta	order_id	pizza_id	quantity
2	1	1	hawaiian_	1
3	2	2	classic_dlx	1
4	3	2	five_chees	1
5	4	2	ital_supr_l	1
6	5	2	mexicana_	1
7	6	2	thai_ckn_l	1
8	7	3	ital_supr_r	1
9	8	3	prsc_argla	1
10	9	4	ital_supr_r	1
11	10	5	ital_supr_r	1
12	11	6	bbq_ckn_s	1
13	12	6	the_greek_	1
14	13	7	spinach_su	1
15	14	8	spinach_su	1
16	15	9	classic_dlx	1
17	16	9	green_garc	1
18	17	9	ital_cpcllo	1
19	18	9	ital_supr_l	1
20	19	9	ital_supr_s	1
21	20	9	mexicana_	1
22	21	9	spicy_ital_	1
23	22	9	spin_pesto	1
24	23	9	veggie_veg	1
25	24	10	mexicana_	1
26	25	10	southw_ck	1

Orders_details

RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED.

```
SELECT
    COUNT(order_id) AS total_orders
FROM
    orders;
```

Result Grid

	total_orders
▶	21350

CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES.

```
SELECT
    ROUND(SUM(order_details.quantity * pizzas.price),
            2) AS total_sales
FROM
    order_details
        JOIN
    pizzas ON pizzas.pizza_id = order_details.pizza_id;
```

IDENTIFY THE HIGHEST-PRICED PIZZA.

```sql
SELECT
    pizza_types.name, pizzas.price
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
ORDER BY pizzas.price DESC
LIMIT 1;
```

	name	price
▶	The Greek Pizza	35.95

IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED.

```
SELECT
    pizzas.size,
    COUNT(order_details.order_details_id) AS order_count
FROM
    pizzas
        JOIN
    order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY pizzas.size
ORDER BY order_count DESC;
```

Result Grid

size	order_count
L	18526
M	15385
S	14137
XL	544
XXL	28

LIST THE TOP 5 MOST ORDERED PIZZA TYPES ALONG WITH THEIR QUANTITIES.

```sql
SELECT
    pizza_types.name, SUM(order_details.quantity) AS quantity
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.name
ORDER BY quantity DESC
LIMIT 5;
```

Result Grid | Filter Rows:

name	quantity
The Classic Deluxe Pizza	2453
The Barbecue Chicken Pizza	2432
The Hawaiian Pizza	2422
The Pepperoni Pizza	2418
The Thai Chicken Pizza	2371

JOIN THE NECESSARY TABLES TO FIND THE TOTAL QUANTITY OF EACH PIZZA CATEGORY ORDERED.

```sql
SELECT
    pizza_types.category,
    SUM(order_details.quantity) AS quantity
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.category
ORDER BY quantity DESC;
```

Result Grid | Filt

category	quantity
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY.

```sql
SELECT
    HOUR(order_time) AS hour, COUNT(order_id) AS order_count
FROM
    orders
GROUP BY HOUR(order_time);
```

Result Grid

hour	order_count
11	1231
12	2520
13	2455
14	1472
15	1468
16	1920
17	2336
18	2399
19	2009
20	1642
21	1198
22	663
23	28
10	8
9	1

JOIN RELEVANT TABLES TO FIND THE CATEGORY-WISE DISTRIBUTION OF PIZZAS.

```
select category, count(name) from pizza_types
group by category;
```

Result Grid | Filter R

	category	count(name)
▶	Chicken	6
	Classic	8
	Supreme	9
	Veggie	9

GROUP THE ORDERS BY DATE AND CALCULATE THE AVERAGE NUMBER OF PIZZAS ORDERED PER DAY.

```sql
SELECT
    ROUND(AVG(quantity), 0)
FROM
    (SELECT
        orders.order_date, SUM(order_details.quantity) AS quantity
    FROM
        orders
    JOIN order_details ON orders.order_id = order_details.order_id
    GROUP BY orders.order_date) AS order_quantity;
```

Result Grid | Filter Rows

	ROUND(AVG(quantity), 0)
▶	138

DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE.

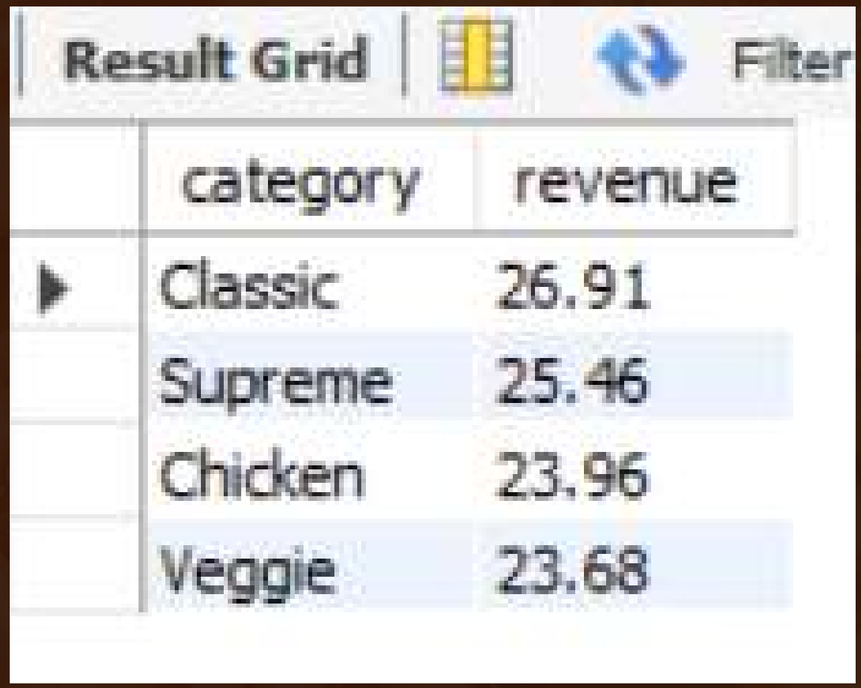
```sql
SELECT
    pizza_types.name,
    SUM(order_details.quantity * pizzas.price) AS revenue
FROM
    pizza_types
        JOIN
    pizzas ON pizzas.pizza_type_id = pizza_types.pizza_type_id
        JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.name
ORDER BY revenue DESC
LIMIT 3;
```

Result Grid | Filter Rows:

name	revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5

CALCULATE THE PERCENTAGE CONTRIBUTION OF EACH PIZZA TYPE TO TOTAL REVENUE

```sql
SELECT
    pizza_types.category,
    ROUND(SUM(order_details.quantity * pizzas.price) / (SELECT
                    ROUND(SUM(order_details.quantity * pizzas.price),
                                2) AS total_sales
                FROM
                    order_details
                        JOIN
                    pizzas ON pizzas.pizza_id = order_details.pizza_id) * 100,
            2) AS revenue
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.category
ORDER BY revenue DESC;
```



category	revenue
Classic	26.91
Supreme	25.46
Chicken	23.96
Veggie	23.68

ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME.

```sql
select order_date,
sum(revenue) over(order by order_date) as cum_revenue
from
(select orders.order_date,
sum(order_details.quantity * pizzas.price) as revenue
from order_details join pizzas
on order_details.pizza_id = pizzas.pizza_id
join orders
on orders.order_id = order_details.order_id
group by orders.order_date) as sales;
```

Result Grid | Filter Rows:

order_date	cum_revenue
2015-01-01	2713.8500000000004
2015-01-02	5445.75
2015-01-03	8108.15
2015-01-04	9863.6
2015-01-05	11929.55
2015-01-06	14358.5
2015-01-07	16560.7
2015-01-08	19399.05
2015-01-09	21526.4
2015-01-10	23990.350000000002
2015-01-11	25862.65
2015-01-12	27781.7
2015-01-13	29831.300000000003
2015-01-14	32358.700000000004
2015-01-15	34343.50000000001
2015-01-16	36937.65000000001
2015-01-17	39001.75000000001
2015-01-18	40978.600000000006
2015-01-19	43365.75000000001
2015-01-20	45763.65000000001
2015-01-21	47804.20000000001
2015-01-22	50300.90000000001
2015-01-23	52724.600000000006

DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE FOR EACH PIZZA CATEGORY

```sql
select name,revenue from
(select category,name,revenue,
rank() over(partition by category order by revenue desc) as rn
from
(select pizza_types.category, pizza_types.name,
sum((order_details.quantity) * pizzas.price) as revenue
from pizza_types join pizzas
on pizza_types.pizza_type_id = pizzas.pizza_type_id
join order_details
on order_details.pizza_id = pizzas.pizza_id
group by pizza_types.category, pizza_types.name) as a) as b
where rn <= 3;
```

Result Grid | Filter Rows:

name	revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5
The Classic Deluxe Pizza	38180.5
The Hawaiian Pizza	32273.25
The Pepperoni Pizza	30161.75
The Spicy Italian Pizza	34831.25
The Italian Supreme Pizza	33476.75
The Sicilian Pizza	30940.5
The Four Cheese Pizza	32265.70000000065
The Mexicana Pizza	26780.75
The Five Cheese Pizza	26066.5

THANK YOU