



**DATABASE MANAGEMENT SYSTEM**

**Individual project**

**Project title: “Magnum(company)”**

**Student name:** Alimkulov Sarvarbek

**Group name:** BDA1904

**Tutor name:** Abylay Salimzhanov

**Nur-Sultan 2020**

## Introduction

I created a managed database system for the Magnum supermarket chain using information I gathered from employees and online sources. Also created functions and indexes, triggers, and views. I made requests and used them joins, window and aggregate functions.

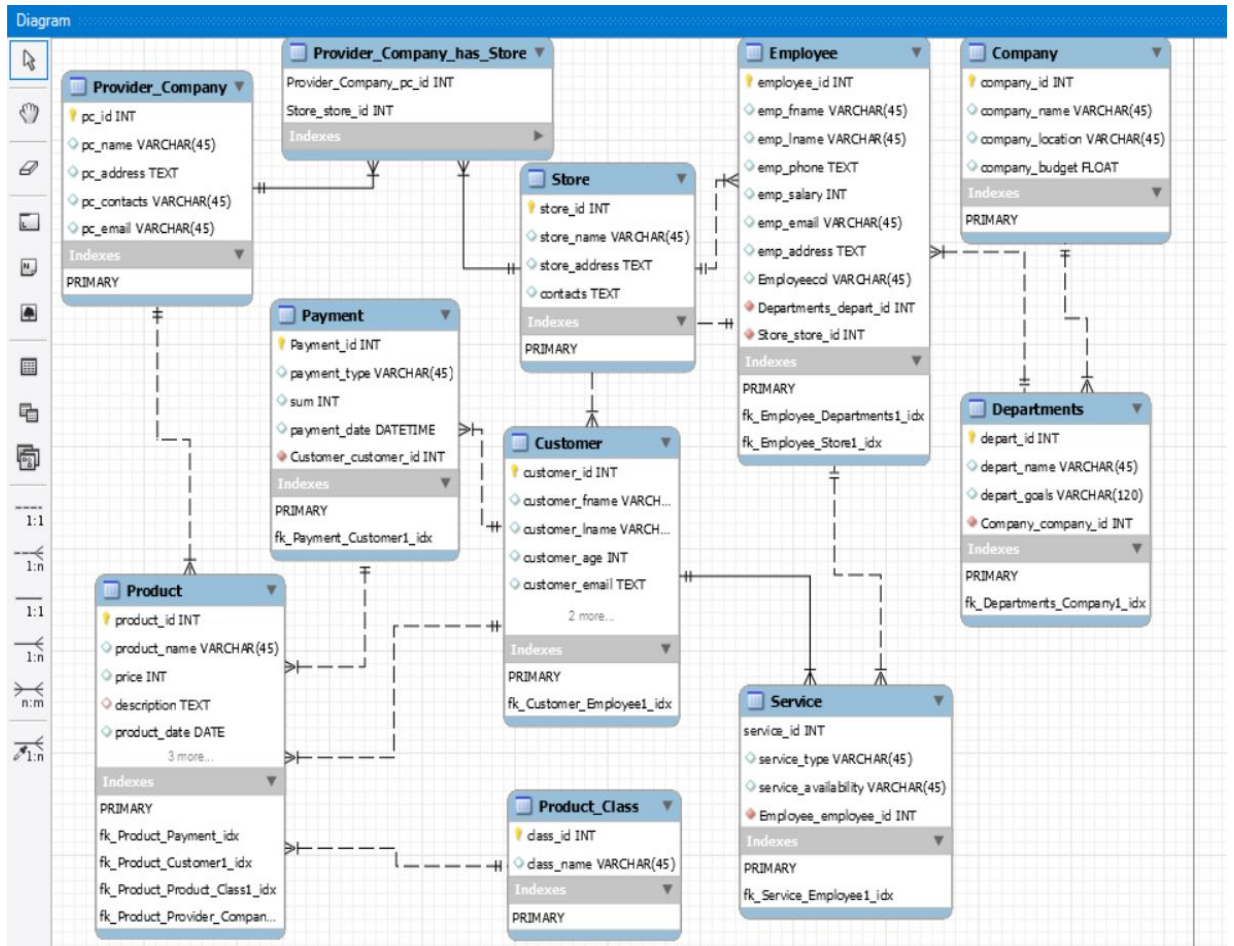
**Description:** Magnum is one of the largest supermarket chains in Kazakhstan. The company traces its history back to September 14, 2007, when the first Magnum store was opened in Alma-ATA. There is a fairly wide range of consumer goods from many supplier companies that Magnum does business with. In addition, Magnum owns quite a variety of stores in different cities of Kazakhstan with a large number of employees and offers customers various types of services.

**Aims:** Make the database system more accessible and simple.

**Information about Magnum company:** Magnum Cash & Carry is the largest retail chain in Kazakhstan, one of the ten largest private companies in the country. The main activity is the sale of consumer goods. The Chain owns 77 shopping complexes of various formats (hypermarkets, supermarkets and small Magnum-express stores) in nine cities of Kazakhstan: Alma-ATA, Nur-Sultan, Kaskelen, Karaganda, Taldykorgan, Shymkent, Petropavlovsk, Kyzylorda and Taraz. The total number of customers of the Magnum Cash & Carry retail chain exceeds 300,000 people per day. The company employs more than 11,000 people.

**Entities:** 1. Company 2. Departments 3. Stores 4. Employee 5. Service 6. Customer 7. Product 8. Product class 9. Payment 10. Provider companies

## Entity relationship diagram (ERD)



## Business rules:

- 1. Company can own many departments. One-many**
- 2. Each Department has one or more employees. One-many**
- 3. There are one or many employees in one store. One-many**
- 4. Each employee can perform one type of service. One-many**
- 5. Each customer can order one or many services. One-many**
- 6. Each customer can make one or many payments. One-many**
- 7. Each employee can serve one or many customers. One-many**
- 8. Each store can be provided with products from many companies. Many-many (used bridge entity)**
- 9. Each company can provide one or many products. One-many**
- 10. Each product belongs to one product type. Many-one**
- 11. Each customer can take or buy one or many products. One-many**

## DDL and DML:

select \* from company

Data Output Explain Messages Notifications				
	company_id [PK] integer	company_name character varying (32)	company_location character varying (32)	company_budget numeric
1	1	Magnum	Almaty,Auezov 1/10	380000000000

select \* from Departments

Data Output Explain Messages Notifications				
	depart_id [PK] integer	depart_name character varying (32)	depart_goals character varying (120)	company_id integer
1	100	Management	control	1
2	101	HR	manage personal	1
3	102	Sales	sale product	1
4	103	Purchasing	contracts with product compa...	1
5	104	Advertisement	advertise company	1
6	105	Finance	income and expense reports	1
7	106	IT	software	1

select \* from Product\_class

Data Output Explain Messages Notifications			
	class_id [PK] integer	class_name character varying (32)	
1	1	Bakery	
2	2	Drinks	
3	3	Chemistry	
4	4	Alcohol	
5	5	Fruits	
6	6	Vegetables	
7	7	Meat_prod	
8	8	Dairy_prod	
9	9	Pastry_prod	
10	10	Household_prod	

select \* from Provider\_company

Data Output		Explain	Messages	Notifications	
	pc_id [PK] integer	pc_name character varying (32)	pc_address text	pc_contacts text	pc_email character varying (32)
1	10	Foodmaster company	Shymkent Al-Far...	777-000	foodmaster@gmail.com
2	11	Coca-cola company	Almaty Abay,69	111-222	cocacola@gmail.com
3	12	Yashkino company	Tomsk Mira,20	333-444	kdv@gmail.com
4	13	Asu company	Nur-Sultan Petro...	221-888	info@asu.kz
5	14	Colgate company	Moscow Krylats...	898-777	contact_Colgate@colpal.com
6	15	head&shoulders company	Moscow 67	666-555	shoulders@gmail.com
7	16	Jack Daniels company	USA Lem Motlo...	070-898	info@jackden.ru
8	17	Tsesna company	Nur-Sultan Al-Fa...	999-654	info@tsesnagroup.kz
9	18	Nestle company	Veve	087-087	contact@ru.nestle.com
10	19	FruitWorld	Moscow Zemlya...	977-666	fruitworld@gmail.com

select \* from store

Data Output	Explain	Messages	Notifications	
	store_id [PK] integer	store_name character varying (32)	store_address text	store_contacts text
1	20	Magnum ATAK	Kenesary st.	65 Editable column
2	21	Magnum Express	Karagandy highway	434-242
3	22	Magnum Cash-Carry	Imanov st.	888-000
4	23	Magnum Cash-Carry	Aitmatova st.	909-707
5	24	Magnum Express	Ryskulova st.	303-202
6	25	Magnum ATAK	Zhubanova st.	254-678
7	26	Magnum Express	Abay st.	900-850
8	27	Magnum Express	Abylai khan st.	600-575
9	28	Magnum Cash-Carry	Republic avenue	888-676
10	29	Magnum ATAK	Mangilik el avenue	565-980

select \* from service

Data Output	Explain	Messages	Notifications
	service_id [PK] integer	service_type character varying (45)	service_availability character varying (32)
1	31	Consultation	Yes
2	32	Receiving orders	Yes
3	33	Delivery	No
4	34	Return goods	No
5	35	Aftersales service	No

select \* from employee

Data Output Explain Messages Notifications

	emp_id [PK] integer	emp_fname character varying (32)	emp_lname character varying (32)	job_name character varying (32)	emp_phone text	emp_salary bigint	emp_email text	emp_address text	store_id integer	depart_id integer
1	40	Alya	Askerova	Manager	8-707-520-90-83	500000	a.alya@mail.ru	Shymkent,Vostok 67	20	100
2	41	Janibek	Smanov	Security	8-777-700-77-90	100000	j.smanov@mail...	Shymkent,Sever 41	21	101
3	42	Danarys	Targarien	Cashier	8-702-714-74-14	170000	danarys01@ga...	Nur-Sultan,Bokeyk...	28	103
4	43	Maksat	Abilov	Consultant	8-771-648-96-47	110000	abilov424@gm...	Karaganda,Seifulin...	26	102
5	44	Baurzhan	Smanov	Programmer	8-771-698-96-47	200000	sm.bauka@mail...	Almaty,Al-Farabi 42...	28	106
6	45	Stas	Alekseev	Security	8-705-648-96-47	160000	astan@mail.ru	Shymkent, Samal-2	24	101
7	46	Askar	Askerov	Accountant	8-747-648-96-47	125000	askerov@gamil...	Nur-Sultan, Mangli...	25	105
8	47	Aset	Marlenov	Marketer	8-701-648-96-47	170000	aset@mail.ru	Almaty,Abay 24	22	104
9	48	Musa	Lesov	Cashier	8-771-648-95-47	135000	nurles@gmail.c...	Karaganda,Zibek-Z...	29	103
10	49	Timur	Rakmanov	Programmer	8-771-648-96-46	230000	rakman@gmail...	Shymkent, Baityrsi...	21	106

select \* from product

Data Output Explain Messages Notifications

	prod_id [PK] integer	prod_name character varying (32)	price integer	description text	prod_date date	class_id integer	pc_id integer	store_id integer
1	70	Milk	320	Natural product ...	2020-12-08	8	10	23
2	71	Fanta	180	carbonated ora...	2021-12-08	2	11	28
3	72	Kit-kat	150	Milk chocolate ...	2021-02-09	9	18	25
4	73	bread	100	flour product	2020-11-10	1	17	29
5	74	Whiski	7500	high quality alco...	2021-12-19	4	16	21
6	75	Water	100	fresh clear drink	2021-08-09	2	13	25
7	76	Apple	350	Natural fruits fr...	2021-01-09	5	19	26
8	77	Tooth paste	400	Helps you to sa...	2021-03-18	10	14	27
9	78	Shampoo	1200	Against brittle a...	2021-03-17	3	15	20
10	79	Tomatos	800	Delicious and ripe	2021-03-18	6	19	25

select \* from customer

Data Output Explain Messages Notifications

	customer_id [PK] integer	customer_fname character varying (32)	customer_lname character varying (32)	customer_phone text	customer_address text	store_id integer	prod_id integer
1	101	Bob	Sander	778-222-111	Kudaiberdiuly 36	23	70
2	102	Sam	Cook	778-111-111	Kabanbai batyr 60a/6	24	73
3	103	Mia	Jonson	223-456-165	Kunaev 34	25	74
4	104	Jake	Hall	778-243-444	Ryskulbekov 45	25	75
5	105	Jose	Sapata	777-232-187	Sharl De Gol 21	26	71
6	106	Maria	Sar	228-265-197	Kunanbayev 78	27	72
7	107	Jon	Blake	788-122-178	Republic 35	21	70
8	108	Susan	Jonson	647-432-761	Mangiik el 61	28	71
9	109	Dan	Vober	869-764-985	Uly dala 75	22	70
10	110	Harry	Potter	678-242-198	Turkistan 1	26	75
11	111	Lionel	Messi	768-879-541	Akmeshit 60	24	76
12	112	Rick	Kick	778-334-675	Almaty 23	25	79
13	113	Ben	Sammat	778-264-111	Kerey and Zhanibek 24	25	74
14	114	Jessica	Sonnen	778-276-653	Koshkarbayev 8	27	77
15	115	Alex	Boges	778-753-111	Musyrepov 8	29	78



select \* from payment

Data Output		Explain	Messages	Notifications			
	pay_id [PK] integer	payment_type character varying (32)	p_sum numeric	p_date date	customer_id integer	prod_id integer	
1	90	card	350	2020-11-06	101	76	
2	91	cash	100	2020-11-06	103	75	
3	92	card	7500	2020-11-06	107	74	
4	93	cash	1200	2020-11-06	109	78	
5	94	cash	100	2020-11-06	108	73	
6	95	card	400	2020-11-06	106	77	
7	96	cash	180	2020-11-06	110	71	
8	97	card	800	2020-11-06	111	79	
9	98	card	150	2020-11-06	112	72	
10	99	cash	320	2020-11-06	102	70	

select \* from provider\_store

	Data Output	Explain	Messages	Notifications
	pc_id integer	store_id integer		
1	10	23		
2	11	28		
3	18	25		
4	17	29		
5	16	21		
6	13	25		
7	19	26		
8	14	27		
9	15	20		
10	19	25		

- **Filtering data**

1. In this query, we can see information about the employee where the names have the letter “a” and the first 5 high salary

SELECT emp\_fname || ' ' || emp\_lname as full\_name, emp\_salary from employee where emp\_fname like '%a%' order by emp\_salary limit 5

	Data Output	Explain	Messages	Notifications
	full_name text	emp_salary bigint		
1	Janibek Sman...	100000		
2	Maksat Abilov	110000		
3	Askar Askerov	125000		
4	Musa Lesov	135000		
5	Stas Alekseev	160000		

2.Retrieve payment sum more than 180 , payment with card,and time when doing payment before now

select p\_sum,payment\_type,now()-p\_date as time\_before\_now from payment where payment\_type in('card') or p\_sum>180 fetch first 3 rows only

	Data Output	Explain	Messages	Notifications
	p_sum numeric	payment_type character varying (32)	time_before_now interval	
1	350	card	1 day 13:35:16.583818	
2	7500	card	1 day 13:35:16.583818	
3	1200	cash	1 day 13:35:16.583818	

3.Retrieve product name is not bread and description where description has letter - t and product date between 08-11-2020 and 08-12-2020

select prod\_name,description from product where description like '%t%' and prod\_name<>'bread' and prod\_date between '08-11-2020' and '08-12-2020' order by prod\_id

	Data Output	Explain	Messages	Notifications
	prod_name character varying (32)	description text		
1	Milk	Natural product ...		

- **Joins**



1.Retrieve full name,product name is bread,store address and where store name is Magnum Express

```
select customer_fname || ' ' || customer_lname as full_name , prod_name, store_address
```

```
from customer
```

```
inner join product on customer.prod_id=product.prod_id
```

```
left join store on customer.store_id=store.store_id
```

```
where customer_fname='Sam' and store.store_name='Magnum Express' and prod_name='bread'
```

Data Output	Explain	Messages	Notifications
	<b>full_name</b> text	<b>prod_name</b> character varying (32)	<b>store_address</b> text
1	Sam Cook	bread	Ryskulova st.

2.Retrieve sum employee salary,job name,departments name is IT,count employee in this departments

```
select sum(emp_salary) as depart_salary,count(emp_id),job_name,depart_name from employee
```

```
inner join departments on employee.depart_id=departments.depart_id
```

```
where depart_name='IT' group by job_name,depart_name
```

Data Output		Explain	Messages	Notifications	
	<b>depart_salary</b> numeric	<b>count</b> bigint	<b>job_name</b> character varying (32)	<b>depart_name</b> character varying (32)	
1	430000	2	Programmer	IT	

### ● Window and aggregate functions

1.Use aggregate functions for define sum of payment with cards

```
select payment_type, sum(p_sum) over (partition by payment_type),count(customer_id) over (partition by payment_type) from payment
```

```
Full join customer using (customer_id)
```

```
where payment_type='card' and customer_fname='Lionel'
```

	Data Output	Explain	Messages	Notifications
	<b>payment_type</b> character varying (32)		<b>sum</b> numeric	<b>count</b> bigint
1	card		800	1

2. Use window functions for sum of salary with job names

select sum(emp\_salary), job\_name from employee group by job\_name order by job\_name

	Data Output	Explain	Messages	Notifications
	<b>sum</b> numeric		<b>job_name</b> character varying (32)	
1	125000	Accountant		
2	305000	Cashier		
3	110000	Consultant		
4	500000	Manager		
5	170000	Marketer		
6	430000	Programmer		
7	260000	Security		

### • Conditional Expressions & Operators

1. We use case for definition level salary

```
select emp_salary, job_name, case when emp_salary <= 100000 then 'low salary'
when emp_salary between 100000 and 200000 then 'medium salary'
else 'High salary'
end as level_salary
from employee
```

Data Output Explain Messages Notifications

	emp_salary bigint	job_name character varying (32)	level_salary text
1	500000	Manager	High salary
2	100000	Security	low salary
3	170000	Cashier	medium salary
4	110000	Consultant	medium salary
5	200000	Programmer	medium salary
6	160000	Security	medium salary
7	125000	Accountant	medium salary
8	170000	Marketer	medium salary
9	135000	Cashier	medium salary
10	230000	Programmer	High salary

2. We use condition operators for define feedback for service

do \$\$

declare

serviceid service.service\_id%type:=31;

type service.service\_type%type;

avialability service.service\_availability%type;

service\_feedback integer;

begin

select serviceid,service\_type,service\_availability from service

into serviceid,type,avialability where serviceid=service\_id;

if not found then

raise notice '% is availability-%',type,service\_feedback;

else

if type='Consultation' then service\_feedback:=8;

elseif type='Receiving orders' then service\_feedback:=10;

else

service\_feedback:=0;

end if;

raise notice '% is availability-%',type,service\_feedback;

end if;

```
end  
$$
```

Data Output	Explain	Messages	Notifications
-------------	---------	----------	---------------

ЗAMEЧAHИE: Consultation is availability-8  
D0

Query returned successfully in 219 msec.

- **Functions**

1. We check count of employee with employee full name

create function emp\_info(full\_name varchar(32))

returns integer

language plpgsql

as \$\$

declare count\_emp integer;

begin

select count(\*) into count\_emp from employee where emp\_fname ||


'||emp\_lname =full\_name;

return count\_emp;

end;

\$\$

select emp\_info('Alya Askerova')

Data Output	Explain	Messages	Notifications
 emp_info integer 			
1	1		

2. Selected all payment sum and count of payment

create or replace function get\_sum(

out pay\_sum int,

out p\_count int

)



language plpgsql

as \$\$

begin

select count(pay\_id),avg(p\_sum) into p\_count,pay\_sum from payment;

end;\$\$

Data Output	Explain	Messages	Notifications
 get_sum record 			
1	(1110,10)		

- **Rollup and cube**

Use cube grouping,employee full name,store\_name where work and these sum of salary

select emp\_fname || ' '|| emp\_lname as full\_name,

store\_name,sum(emp\_salary)

from employee

inner join store on employee.store\_id=store.store\_id

group by cube(full\_name,store\_name)

	Data Output	Explain	Messages	Notifications
	<b>full_name</b> text	<b>store_name</b> character varying (32)	<b>sum</b> numeric	
1	[null]	[null]	1900000	
2	Askar Askerov	Magnum ATAК	125000	
3	Danarys Targ...	Magnum Cash-Carry	170000	
4	Baurzhan Sma...	Magnum Cash-Carry	200000	
5	Maksat Abilov	Magnum Express	110000	
6	Janibek Sman...	Magnum Express	100000	
7	Stas Alekseev	Magnum Express	160000	
8	Aset Marlenov	Magnum Cash-Carry	170000	
9	Alya Askerova	Magnum ATAК	500000	
10	Timur Rakman...	Magnum Express	230000	
11	Musa Lesov	Magnum ATAК	135000	
12	Askar Askerov	[null]	125000	
13	Musa Lesov	[null]	135000	
14	Danarys Targ...	[null]	170000	
15	Alya Askerova	[null]	500000	
16	Baurzhan Sma...	[null]	200000	
17	Stas Alekseev	[null]	160000	
18	Janibek Sman...	[null]	100000	
19	Aset Marlenov	[null]	170000	
20	Maksat Abilov	[null]	110000	
21	Timur Rakman...	[null]	230000	
22	[null]	Magnum ATAК	760000	
23	[null]	Magnum Cash-Carry	540000	
24	[null]	Magnum Express	600000	

1. In this query use rolup grouping, count products with column prod\_name, class\_name

```

select prod_name, class_name, count(prod_id)
from product
inner join Product_class on
product.class_id=Product_class.class_id
group by rollup(prod_name, class_name)

```



Data Output Explain Messages Notifications

	prod_name character varying (32)	class_name character varying (32)	count bigint
1	[null]	[null]	10
2	Fanta	Drinks	1
3	Whiski	Alcohol	1
4	Tooth paste	Household_prod	1
5	Milk	Dairy_prod	1
6	Shampoo	Chemistry	1
7	Apple	Fruits	1
8	Kit-kat	Pastry_prod	1
9	bread	Bakery	1
10	Water	Drinks	1
11	Tomatos	Vegetables	1
12	Tomatos	[null]	1
13	bread	[null]	1
14	Whiski	[null]	1
15	Water	[null]	1
16	Kit-kat	[null]	1
17	Fanta	[null]	1
18	Milk	[null]	1
19	Apple	[null]	1
20	Shampoo	[null]	1
21	Tooth paste	[null]	1

- Views

1. Creating view prod\_info with columns prod\_name, class\_name, remains\_time

Create view prod\_info as select prod\_name,  
class\_name, prod\_date-now() as remains\_time  
from product

inner join Product\_class on product.class\_id=Product\_class.class\_id  
group by (prod\_name, class\_name, remains\_time)

select \* from prod\_info

	Data Output	Explain	Messages	Notifications
	prod_name character varying (32)		class_name character varying (32)	remains_time interval
1	Fanta		Drinks	395 days 09:26:40....
2	Whiski		Alcohol	406 days 09:26:40....
3	Milk		Dairy_prod	30 days 09:26:40.3...
4	Apple		Fruits	62 days 09:26:40.3...
5	Water		Drinks	274 days 09:26:40....
6	Tomatos		Vegetables	130 days 09:26:40....
7	Kit-kat		Pastry_prod	93 days 09:26:40.3...
8	Tooth paste		Household_prod	130 days 09:26:40....
9	Shampoo		Chemistry	129 days 09:26:40....
10	bread		Bakery	2 days 09:26:40.30...

2.Create view cust\_check with columns p\_sum,customer full\_name,times for payment

Create view cust\_check as select p\_sum,prod\_name,customer\_fname ||  
'||customer\_lname as full\_name,now()-p\_date as times

from payment

full join customer on customer.customer\_id=payment.customer\_id

inner join product on product.prod\_id=payment.prod\_id

select \* from cust\_check

	Data Output	Explain	Messages	Notifications
	p_sum numeric	prod_name character varying (32)	full_name text	times interval
1	320	Milk	Sam Cook	1 day 14:37...
2	180	Fanta	Harry Potter	1 day 14:37...
3	150	Kit-kat	Rick Kick	1 day 14:37...
4	100	bread	Susan Jonson	1 day 14:37...
5	7500	Whiski	Jon Blake	1 day 14:37...
6	100	Water	Mia Jonson	1 day 14:37...
7	350	Apple	Bob Sander	1 day 14:37...
8	400	Tooth paste	Maria Sar	1 day 14:37...
9	1200	Shampoo	Dan Vober	1 day 14:37...
10	800	Tomatos	Lionel Messi	1 day 14:37...

## Triggers

Create or replace function salary\_change()

returns trigger

language plpgsql

```

as $$
begin if NEW.emp_salary>OLD.emp_salary Then
raise 'Salary increase';
end if;
return new.emp_salary;
end;
$$

```

Create trigger changing before update on employee for each row execute procedure salary\_change();

Data Output	Explain	Messages	Notifications
CREATE TRIGGER			
Query returned successfully in 126 msec.			



- **Indexes**

1.Create index for the p\_sum column table payment and p\_sum not equal to 200

```

create index idx_psum on payment(p_sum) where p_sum!=200
explain select * from payment where payment_type='Card'

```


Data Output	Explain	Messages	Notifications
<div> <div>  </div> <div> <b>QUERY PLAN</b> </div> <div> text </div> <div>  </div> </div>			
1	Seq Scan on paym...		
2	Filter: ((payment_...		

1.Create index for the job\_name table employee

```

create index idx_job_name on employee(job_name)
explain select * from employee where depart_id>102

```

Data Output	Explain	Messages	Notifications
	<div> <div>QUERY PLAN</div> <div>text</div> <div>  </div> </div>		
1	Seq Scan on empl...		
2	Filter: (depart_id ...		

**Summary:** During the work on this project I reminded and repeated all materials of Database Management courses and tried my practice skills. Furthermore, I developed my skills in creating a management database system for the company, in my case it is Magnum, and working with SQL language.