

DB분석설계

동서울터미널 DB분석

<https://www.ti21.co.kr/index.aspx>

2025-11-25

김성중, 박찬하, 이령경, 정해연

뒷조

Contents

분석 및 수집

01. 요구사항 명세서

E-R Diagram 설계

02. 개념모델

데이터 요소의 구조와 데이터 요소간의 관계 설계

03. 논리모델

04. 테이블 명세서

SQL 테이블 생성

05. 물리 모델

06. SQL 조회

예매관리

1. 예매관리는 가입을 하지 않아도 가능하다.
2. 예매관리는 예매코드(PK), 예매날짜, 휴대폰번호, 생년월일, 카드번호, 운행코드(FK), 좌석번호, 승객유형코드(FK)의 정보를 가지며, 예매코드로 식별한다.
3. 예매관리는 버스등급, 승객타입, 인원수를 선택해야 하며 금액산정 세부내역은 각 테이블의 명세를 참고한다.
4. 예매관리와 운행정보는 1:N 관계를 갖는다.

운행정보

1. 운행정보는 운행코드(PK), 날짜, 운수사코드(FK), 출발시간, 도착시간, 예상소요시간, 출발터미널코드(FK), 도착터미널코드(FK), 버스등급코드(FK), 버스유형코드(FK), 총 좌석, 잔여좌석 정보를 가진다.
2. 운행정보는 운행코드로 식별한다.
3. 운행정보와 운수사, 도착지, 출발지, 버스유형, 버스등급 과 1:N 관계를 갖는다.
4. 거리에 따른 금액차등은 고려하지 않는다.

출발터미널

1. 출발터미널은 출발터미널코드, 출발지역, 출발터미널이름을 가진다.
2. 출발터미널은 출발터미널코드로 식별한다.
3. 경유되는 터미널은 고려하지 않는다.
4. 출발 터미널과 도착 터미널은 중복될 수 없다.

도착터미널

1. 도착터미널은 도착터미널코드, 도착지역, 도착터미널이름을 가진다.
2. 도착터미널은 도착터미널코드로 식별한다.
3. 경유되는 터미널은 고려하지 않는다.
4. 출발 터미널과 도착 터미널은 중복될 수 없다.

버스유형

1. 버스유형은 버스유형코드, 버스유형, 기본요금을 가진다.
2. 버스유형은 버스유형코드로 식별한다.
3. 버스유형은 시외버스와 고속버스로 구분된다.
4. 시외버스 일반은 20,000원, 고속버스 일반은 22,000원 이다.

승객유형

1. 운행정보는 운행코드(PK), 날짜, 운수사코드(FK), 출발시간, 도착시간, 예상소요시간, 출발터미널코드(FK), 도착터미널코드(FK), 버스(FK), 총 좌석, 잔여좌석 정보를 가진다.
2. 운행정보는 운행코드로 식별한다.
3. 승객유형은 아동, 중고생, 어른, 보훈으로 구분된다.
4. 어른을 기준으로 아동은 10,000원, 중고생은 5,000원, 보훈은 12,000원 할인된다.

운수사

1. 운수사코드(FK), 운수사이름, 운수사 연락처, 운수사주소의 정보를 가진다.
2. 운수사코드로 식별한다.
3. 운수사 주소는 운행지와 상관 없다.

버스등급

1. 버스등급은 버스등급코드(FK), 버스등급 , 버스등급별 할인율을 가진다.
2. 버스등급은 버스등급코드로 식별한다.
3. 버스등급은 일반, 우등, 프리미엄으로 구분한다.
4. 일반을 기준으로 우등은 5,000원 추가, 프리미엄은 10,000원이 추가된다.

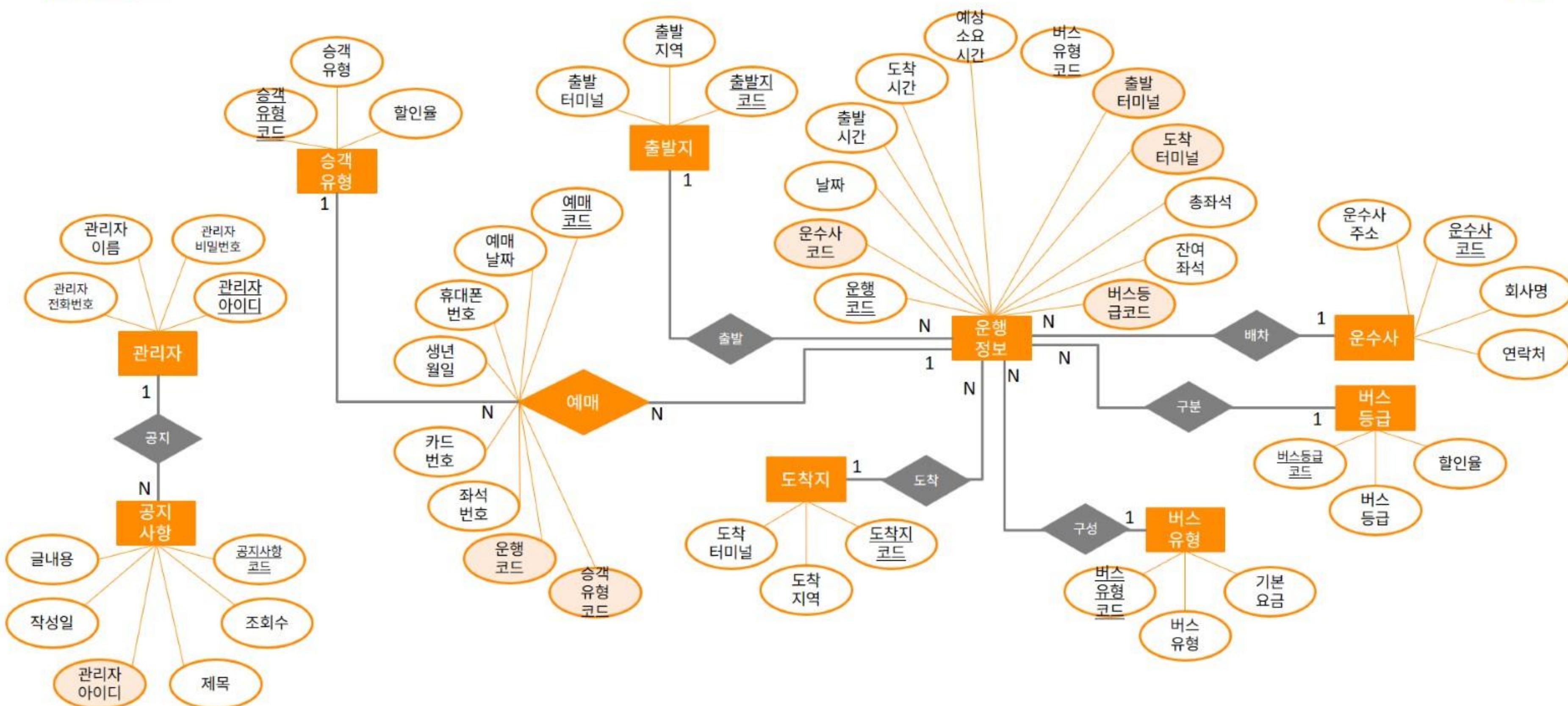
관리자

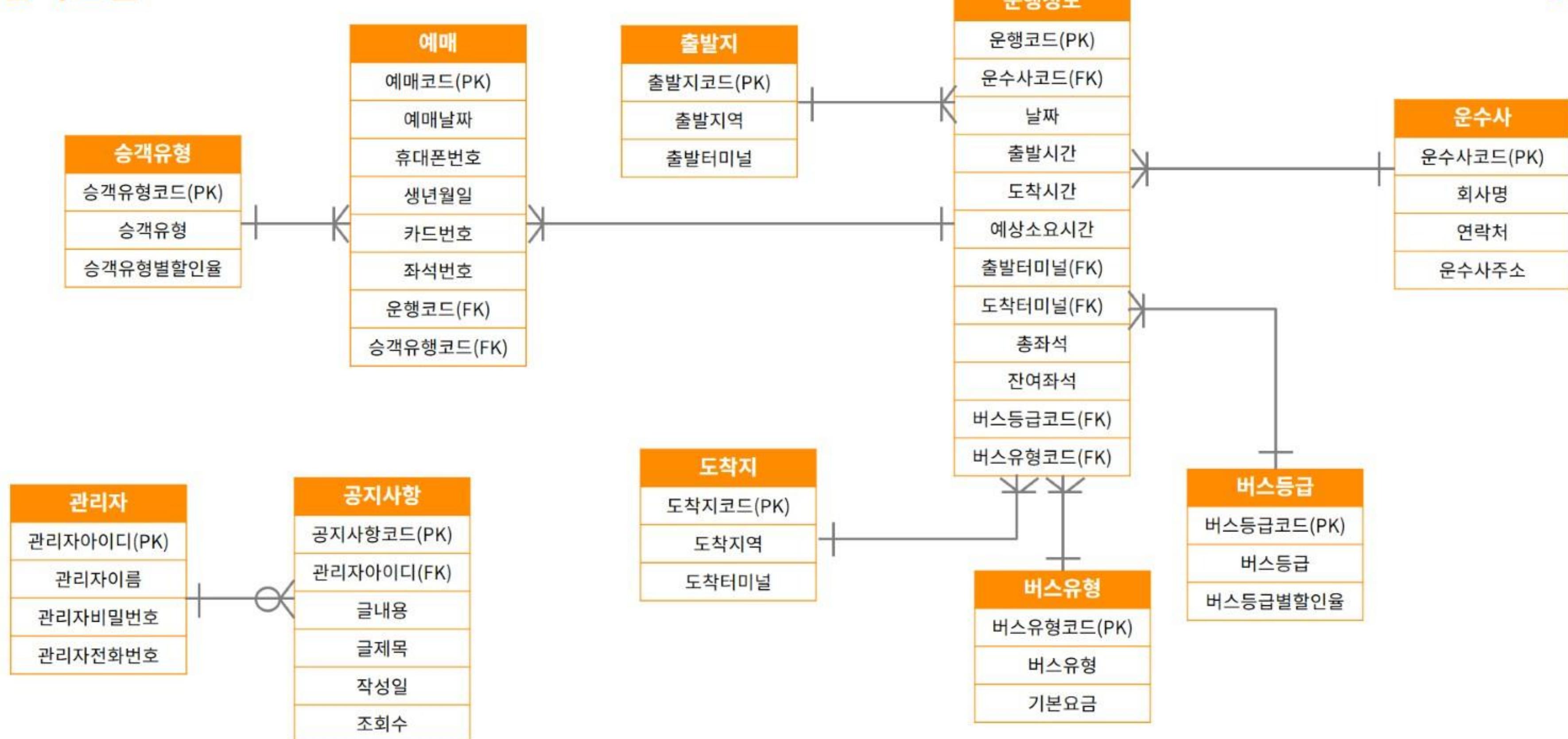
1. 관리자는 관리자아이디, 관리자비밀번호, 관리자이름, 관리자전화번호의 정보를 가지며, 관리자 아이디로 식별한다.
2. 관리자는 모든 정보를 추가, 삭제, 수정, 검색할 수 있다.
3. 관리자는 공지사항과 1:N 관계를 갖는다.

공지사항

1. 공지사항은 공지사항코드, 제목, 작성자, 작성일, 조회수를 가진다.
2. 공지사항은 관리자만이 접근 가능하며, 고객들은 열람만 가능하다.

개념모델





테이블 명세서

04

테이블 영문명	routelInfo	테이블 한글명	운행정보			
한글 컬럼명	Field	Type	Null	Key	Default	Extra
운행코드	r_id	Int		PRI		Auto-Incre
날짜	r_date	Date	NO			
운수사코드	c_id	Int	NO			
출발시간	r_depature	Time	NO			
도착시간	r_arrival	Time	NO			
소요시간	r_takeTime	Time				
출발지코드	d_id	Int	NO			
도착지코드	a_id	Int	NO			
버스등급코드	bc_id	Tinyint	NO			
버스유형코드	bt_id	Tinyint	NO			
총좌석	r_totalseat	Tinyint	NO		0	Unsigned
잔여좌석	r_available	Tinyint				Unsigned

테이블 명세서

테이블 영문명	departure	테이블 한글명	출발지			
한글 컬럼명	Field	Type	Null	Key	Default	Extra
출발지코드	d_id	Int		PRI		Auto-Incre
출발지역	d_region	Varchar(20)	NO			
출발터미널	d_terminal	Varchar(20)	NO			

테이블 영문명	arrival	테이블 한글명	도착지			
한글 컬럼명	Field	Type	Null	Key	Default	Extra
도착지코드	a_id	Int		PRI		Auto-Incre
도착지역	a_region	Varchar(20)	NO			
도착터미널	a_terminal	Varchar(20)	NO			

테이블 영문명	busClass	테이블 한글명	버스등급			
한글 컬럼명	Field	Type	Null	Key	Default	Extra
버스등급코드	bc_id	Tinyint		PRI		Auto-Incre
등급	bc_class	Varchar(20)	NO		'일반'	
할인율	bc_discount	Int	NO		0	

테이블 영문명	company	테이블 한글명	운수사			
한글 컬럼명	Field	Type	Null	Key	Default	Extra
운수사코드	c_id	Int		PRI		Auto-Incre
회사명	c_name	Varchar(20)	NO			
연락처	c_phone	Varchar(20)				
주소	c_address	Varchar(50)				

테이블 영문명	busType	테이블 한글명	버스유형			
한글 컬럼명	Field	Type	Null	Key	Default	Extra
버스유형코드	bt_id	Tinyint		PRI		Auto-Incre
유형	bt_type	Varchar(10)	NO			
기본요금	bt_price	Varchar(20)	NO		0	Unsigned

테이블 명세서

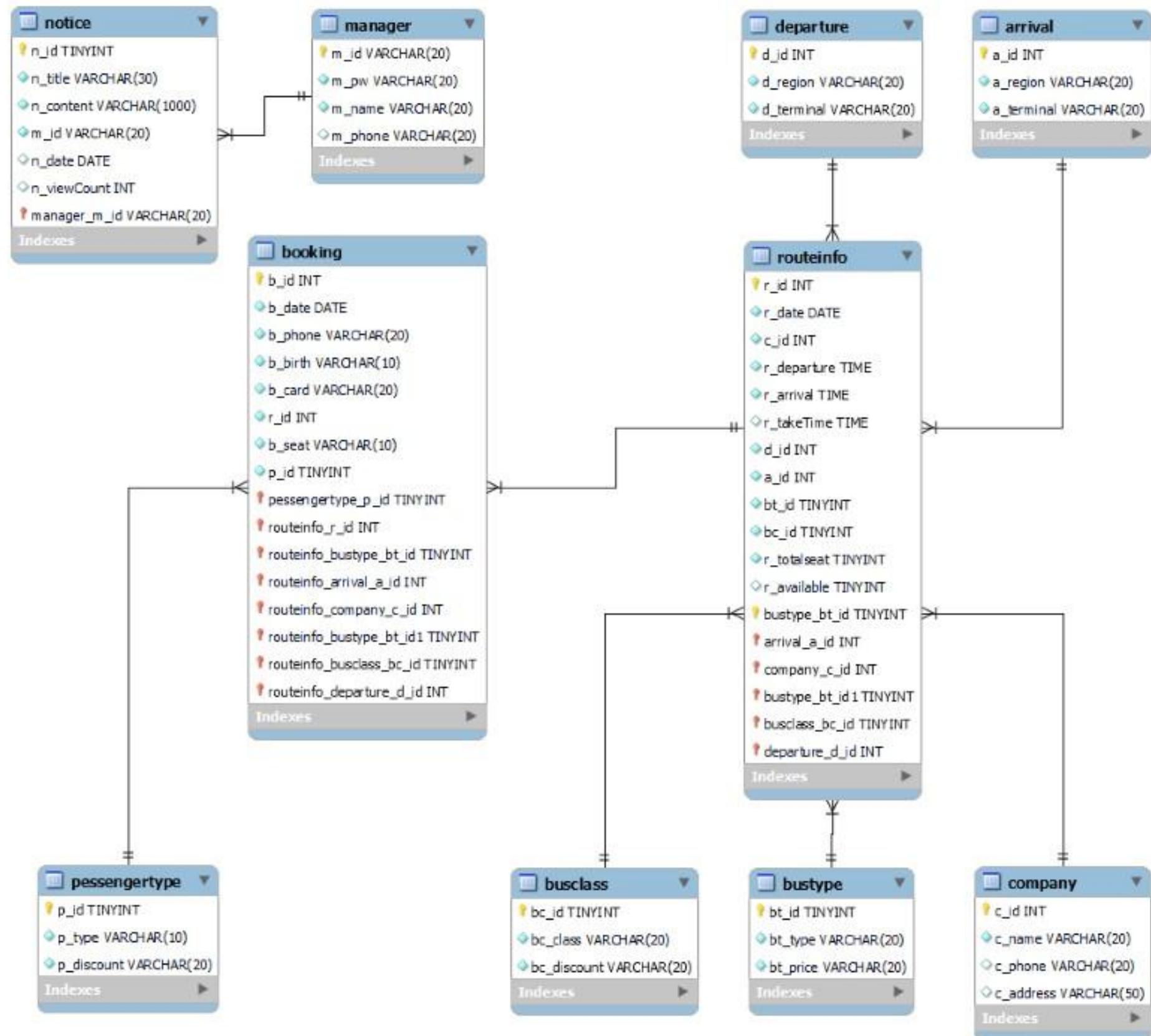
테이블 영문명	booking	테이블 한글명	예매			
한글 컬럼명	Field	Type	Null	Key	Default	Extra
예매코드	b_id	Int		PRI		Auto-Incre
예매날짜	b_date	Date	NO		Currdate	
휴대폰번호	b_phone	Varchar(20)	NO			
생년월일	b_birth	Varchar(10)	NO			
카드번호	b_card	Varchar(20)	NO			Unsigned
운행코드	r_id	Int	NO			
좌석번호	b_seat	Varchar(10)	NO			
승객유형코드	p_id	Tinyint	NO			

테이블 영문명	pessengerType	테이블 한글명	승객유형			
한글 컬럼명	Field	Type	Null	Key	Default	Extra
승객유형코드	p_id	Tinyint		PRI		Auto-Incre
승객유형	p_type	Varchar(10)	NO			
할인율	p_discount	Int	NO			

테이블 명세서

테이블 영문명	manager	테이블 한글명	관리자			
한글 컬럼명	Field	Type	Null	Key	Default	Extra
아이디	m_id	Varchar(20)		PRI		
비밀번호	m_pw	Varchar(20)	NO			
이름	m_name	Varchar(20)	NO			
전화번호	m_phone	Varchar(20)				

테이블 영문명	notice	테이블 한글명	공지사항			
한글 컬럼명	Field	Type	Null	Key	Default	Extra
공지사항코드	n_id	Tinyint		PRI		Auto-Incre
제목	n_title	Varchar(30)	NO			
내용	n_content	Varchar(1000)	NO			
관리자아이디	m_id	Varchar(20)	NO			
작성일	n_date	Date			Currentdate	
조회수	n_viewCount	Int			0	Unsigned



전체 승객 중 아동 승객 예매 정보 출력

```
i30  
i31 • select  
i32     p_type as '승객유형', b_date as '예약일자',  
i33     bt_type as '버스종류', bc_class as '버스등급',  
i34     r_date as '운행날짜', d_terminal as '출발터미널',  
i35     a_terminal as '도착터미널', r_departure as '출발시간',  
i36     r_takeTime as '예상소요시간', r_arrival as '예상도착시간',  
i37     (bt.bt_price + bc.bc_class + p.p_discount) as '결제금액' from booking b  
i38     inner join routeinfo r on r.r_id = b.r_id  
i39     inner join departure d on d.d_id = r.d_id  
i40     inner join arrival a on a.a_id = r.a_id  
i41     inner join busclass bc on bc.bc_id = r.bc_id  
i42     inner join bustype bt on bt.bt_id = r.bt_id  
i43     inner join pessengertype p on p.p_id = b.p_id  
i44     where b.p_id = 2;
```

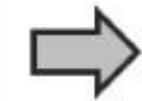
Result Grid | Filter Rows: | Export: | Wrap Cell Content:

승객 유형	예약일자	버스 종류	버스 등급	운행날짜	출발터미널	도착터미널	출발시간	예상소요시간	예상도착시간	결제금액
아동	2025-11-19	시외	일반	2025-12-06	의정부	강릉	12:00:00	01:45:00	13:45:00	10000
아동	2025-11-19	고속	우등	2025-12-04	성남	주문진	17:45:00	01:30:00	19:15:00	12000
아동	2025-11-25	고속	프리미엄	2025-12-02	정주대	속초	21:00:00	02:15:00	23:15:00	12000
아동	2025-11-25	고속	일반	2025-12-03	의정부	남원	14:30:00	02:15:00	16:45:00	12000
아동	2025-11-26	시외	프리미엄	2025-12-04	대전정사	수원	22:15:00	01:15:00	23:30:00	10000

운행정보가 없는 예약내역 삭제하기

```
delete from booking b
where r_id > 51;
```

b_id	b_date	b_phone	b_birth	b_card	r_id	b_seat	p_id
1	2025-11-19	010-4124-1000	571125	126487789453	55	12	1
2	2025-11-19	010-1242-8882	680223	128465489484	32	13	1
3	2025-11-19	010-2532-1992	790311	125456789485	53	14	1
4	2025-11-19	010-1112-1002	840402	127456789486	2	15	1
5	2025-11-19	010-2222-1232	960503	225456789487	65	16	1
6	2025-11-19	010-4262-1262	040604	333456789488	143	17	1
7	2025-11-19	010-4262-1242	120705	553445979489	11	18	2
8	2025-11-19	010-9962-1182	230804	447558789480	33	19	2
9	2025-11-19	010-8862-2222	940903	668757789484	100	29	1
10	2025-11-19	010-4266-3332	860102	553986789485	32	30	1
11	2025-11-19	010-4222-4442	770119	344565719486	17	31	1
12	2025-11-19	010-4444-5552	930401	978434729487	72	36	1
13	2025-11-19	010-4555-6662	821128	863233739485	123	38	1
14	2025-11-19	010-4666-7772	700231	755122759443	98	1	1
15	2025-11-19	010-4777-8882	670428	655091769453	98	18	4



b_id	b_date	b_phone	b_birth	b_card	r_id	b_seat	p_id
2	2025-11-19	010-1242-8882	680223	128465489484	32	13	1
4	2025-11-19	010-1112-1002	840402	127456789486	2	15	1
7	2025-11-19	010-4262-1242	120705	553445979489	11	18	2
8	2025-11-19	010-9962-1182	230804	447558789480	33	19	2
10	2025-11-19	010-4266-3332	860102	553986789485	32	30	1
11	2025-11-19	010-4222-4442	770119	344565719486	17	31	1
16	2025-11-19	010-4888-9992	541118	111097729463	17	11	4
17	2025-11-19	010-4111-1772	420517	992085719483	4	30	4
18	2025-11-19	010-4222-1882	750614	888086729493	6	12	1
20	2025-11-20	010-3332-1112	741211	666876789433	23	29	1
21	2025-11-20	010-4442-1332	821223	555096759484	23	5	1
22	2025-11-20	010-5552-1211	720312	444006524483	23	4	1
24	2025-11-20	010-5772-1222	400512	222466789413	44	2	4
25	2025-11-20	010-4232-1233	320511	111896789233	30	1	4
26	2025-11-20	010-4262-1244	530423	122496789313	20	3	1

구리에서 공주가는 노선의 승객정보

```
511
512 • select b.b_phone as '전화번호', p.p_type as '승객유형', (bt.bt_price + bc.bc_class + p.p_discount) as '결제금액' from routeinfo r
      inner join booking b on b.r_id = r.r_id
      inner join passenger p on p.p_id = b.p_id
      inner join busclass bc on bc.bc_id = r.bc_id
```

<

Result Grid | Filter Rows: Export: Wrap Cell Content:

전화번호	승객유형	결제금액
010-1322-1252	보통	10000
010-4213-8858	어른	22000

11/25부터 예매한 정보 출력

```

1
2 • select b.b_id as '예매코드',
3   concat_ws(' ', b.b_date, r.r_arrival) as '예매일',
4   concat_ws(' / ', b.b_seat, p.p_type) as '좌석번호 / 매수'
5   from booking b
6   inner join routeinfo r on r.r_id = b.r_id
7   inner join passenger type p on p.p_id = b.p_id
8   where b.b_date >= '2025-11-25';

```

Result Grid | Filter Rows: [] | Export: [] | Wrap Cell Content: []

	예매코드	예매일	좌석번호 / 매수
95	2025-11-25 23:15:00	32 / 아동	
96	2025-11-25 00:00:00	31 / 어른	
97	2025-11-25 00:45:00	1 / 어른	
98	2025-11-25 23:15:00	2 / 어른	
99	2025-11-25 18:45:00	7 / 어른	
100	2025-11-25 16:15:00	22 / 어른	
101	2025-11-25 16:45:00	29 / 아동	
102	2025-11-25 17:30:00	19 / 어른	
103	2025-11-25 18:30:00	17 / 중고상	
104	2025-11-25 21:15:00	15 / 보호	
105	2025-11-25 21:15:00	14 / 어른	
106	2025-11-25 23:15:00	11 / 어른	
107	2025-11-26 00:00:00	12 / 어른	
108	2025-11-26 23:30:00	22 / 아동	
113	2025-11-26 21:30:00	4 / 어른	
114	2025-11-26 19:30:00	5 / 어른	
115	2025-11-26 18:45:00	6 / 어른	
116	2025-11-26 22:00:00	7 / 어른	

전주에서 출발해서 인천에 도착하는 버스의 시간표

```

15
16 • select d.d_terminal as '출발터미널',
17   a.a_terminal as '도착터미널',
18   r.r_departure as '출발시간',
19   r.r_arrival as '도착시간'
20   from departure d
21   inner join routeinfo r on r.d_id = d.d_id
22   inner join arrival a on a.a_id = r.a_id
23   where d.d_terminal = '군산' and a.a_terminal = '인천';

```

Result Grid | Filter Rows: [] | Export: [] | Wrap Cell Content: []

	출발터미널	도착터 미널	출발시간	도착시 간
	군산	인천	22:15:00	00:45:00

관리자가 작성한 글 제목과 글 내용 중 '시외'를 포함하고 있는
게시글의 글번호와 글제목, 글내용(20글자)을 출력

```
26  
27 • select n_id as '글번호', n_title as '글제목', substr(n_content, 1, 20) as '글내용' from notice  
28   where n_title like '%시외%' or n_content like '%시외%' ;  
29  
30
```

글번호	글제목	글내용
1	시외버스 운행중단노선 안내	2025년 12월 01일부터 장리-상
2	시외버스 신규노선 안내	-시외버스 신규노선 안내 - 시...
3	출주노선 프리미엄 자량운행 안내	- 프리미엄 자량 운행 안내 - 시
5	2025년 추석 특별수송 예매 안내	* 2025년 추석 특별수송 예매 안
6	시외버스 신규 노선 안내 (광양, 동광양)	- 시외버스 신규 노선 안내 -

예매 정보에서 승객유형이 어른인 사람 / 총 예매 수 출력

```
18  
19 • select concat((truncate((count(*)/85),2)*100),'%')  
20   as '어른 예매자 비율' from booking b  
21   inner join passenger_type p on b.p_id = p.p_id  
22   where p.p_type = '어른';
```

어른 예매자 비율
75.00%

좌석이 프리미엄인 고객의 전화번호 가운데 4자리만 ****로 수정하여 출력

```
27  
28 • select b.b_id, b.b_seat, replace(b_phone,substr(b_phone,5,4),'****'), b.b_birth, r.r_date,bc.bc_class from booking b  
29   inner join routeinfo r on r.r_id = b.r_id  
30   inner join busclass bc on bc.bc_id = r.bc_id  
31 where bc.bc_class='프리미엄';
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: //

	b_id	b_seat	replace(b_phone,substr(b_phone,5,4),'****')	b_birth	r_date	bc_class
▶	2	13	010-****-8882	680223	2025-12-03	프리미엄
	10	30	010-****-3332	860102	2025-12-03	프리미엄
	17	30	010-****-1772	420517	2025-12-05	프리미엄
	20	29	010-****-1112	741211	2025-11-30	프리미엄
	21	5	010-****-1332	821223	2025-11-30	프리미엄
	22	4	010-****-1211	720312	2025-11-30	프리미엄
	24	2	010-****-1222	400512	2025-12-03	프리미엄
	42	40	010-****-1445	441202	2025-12-05	프리미엄
	62	39	010-****-1114	750908	2025-12-05	프리미엄
	63	1	010-****-1265	961011	2025-12-05	프리미엄
	68	12	010-****-1876	770318	2025-12-02	프리미엄
	73	28	010-****-1264	710811	2025-12-02	프리미엄
	74	10	010-****-1225	490902	2025-12-02	프리미엄
	79	11	010-****-8822	751202	2025-12-02	프리미엄
	80	15	010-****-1992	661102	2025-12-02	프리미엄
	95	32	010-****-2422	211019	2025-12-02	프리미엄
	97	1	010-****-6654	831017	2025-12-02	프리미엄
	98	2	010-****-5435	751016	2025-12-02	프리미엄
	100	22	010-****-4512	861014	2025-11-30	프리미엄
	102	19	010-****-1765	681211	2025-12-02	프리미엄
	106	11	010-****-1323	801201	2025-12-02	프리미엄
	108	22	010-****-9522	220501	2025-12-04	프리미엄
	114	5	010-****-1582	050812	2025-12-03	프리미엄

동서울 -> 동동서울로 수정하여 출력

```

8 • update arrival
9   set a_terminal=concat('동',a_terminal)
10  where a_terminal='동서울';
11
12
13 • select * from arrival;

```

Result Grid | Filter Rows: [] | Edit: []

a_id	a_region	a_terminal
1	서울	동동서울
2	서울	서울경부
3	서울	센트럴시티(서울)
4	경기	고양백석
5	경기	구리
6	경기	부천
7	경기	수원
8	경기	안양
9	경기	의정부
10	경기	성남
11	경기	안산
12	경기	동인
13	인천	인천
14	인천	인천공항T1
15	인천	인천Air2
16	강원도	강릉
17	강원도	동해
18	강원도	삼척
19	강원도	속초
20	강원도	주문진
21	강원도	춘천
22	강원도	횡성
23	대전	대전복합

예매내역의 승객유형, 버스유형, 버스등급을 통한 요금 산출

```

25
26 • select p.p_type as '승객유형', bt.bt_type as '버스유형', bc.bc_class as '버스등급',
27   concat(bt_price+p.p_discount+bc.bc_discount,'원') as '요금' from booking b
28   inner join passengertype p on b.p_id = p.p_id
29   inner join routeinfo r on b.r_id = r.r_id
30   inner join busclass bc on bc.bc_id = r.bc_id
31   inner join bustype bt on bt.bt_id = r.bt_id;

```

Result Grid | Filter Rows: [] | Export: [] | Wrap Cell Content: []

승객 유형	버스 유형	버스 등급	요금
어른	고속	프리미엄	32000 원
어른	고속	우등	27000 원
아동	시외	일반	10000 원
아동	고속	우등	17000 원
어른	고속	프리미엄	32000 원
어른	고속	우등	27000 원
보통	고속	우등	15000 원
보통	고속	프리미엄	20000 원
어른	고속	일반	22000 원
어른	시외	프리미엄	30000 원
어른	시외	프리미엄	30000 원
어른	시외	프리미엄	30000 원
보통	시외	프리미엄	18000 원
보통	고속	일반	10000 원
어른	고속	일반	22000 원
보통	고속	일반	10000 원
보통	시외	일반	8000 원
보통	고속	우등	15000 원
어른	시외	일반	20000 원
어른	고속	일반	22000 원
어른	고속	우등	27000 원
어른	고속	일반	22000 원
어른	고속	우등	???

예매정보 기반 운수사 별 운행횟수

```

1
2 • SELECT c.c_id as '운수사코드', c.c_name as '회사명', COUNT(distinct r.r_id) as '운행횟수' FROM booking b
3   inner join routeinfo r on r.r_id = b.r_id
4   inner join company c on r.c_id = c.c_id
5   group by c.c_id
6   order by c.c_id;

```

<

Result Grid | Filter Rows: [] Export: [] Wrap Cell Content: []

운수사 코드	회사 명	운행 횟수
1	동부고속	6
2	동양고속	9
3	상화고속	6
4	경전여객	3
5	천일여객	2
6	충남고속	4
7	강원고속	1
8	대원고속	1
10	아성고속	1

예매정보 기반 운수사 별 총매출

```

8
9 • select c.c_id as '운수사코드', c.c_name as '회사명', sum(bt.bt_price + bc.bc_discount + p.p_discount) as '총 매출액' from routeinfo r
10  inner join company c on r.c_id = c.c_id
11  inner join busclass bc on bc.bc_id = r.bc_id
12  inner join bustype bt on bt.bt_id = r.bt_id
13  inner join booking b on b.r_id = r.r_id
14  inner join passenger p on b.p_id = p.p_id
15  group by c.c_id
16  order by c.c_id;

```

<

Result Grid | Filter Rows: [] Export: [] Wrap Cell Content: []

운수사 코드	회사 명	총 매 출 액
1	동부고속	255000
2	동양고속	525000
3	상화고속	384000
4	경전여객	104000
5	천일여객	229000
6	충남고속	318000
7	강원고속	25000
8	대원고속	20000
10	아성고속	44000