

1	SQL	담당자																
문항	30076번 객실의 대여 일정을 확인한다. (대여 접수 번호, 대여 회원 이름, 대여 시작일, 대여 종료일)	김희진																
<pre>select rr.rental_receive_no "대여 접수 번호", m.member_name "회원 이름", rr.rental_start_date "대여 시작일", rr.rental_finish_date "대여 종료일" from member_info m, rental_receive rr where m.member_no = rr.member_no and rr.rental_receive_no in (select rental_receive_no from room_rental_record where camp_room_info_no = 30076);</pre>																		
결과																		
<table><tr><th>대여 접수 번호</th><th>회원 이름</th><th>대여 시작일</th><th>대여 종료일</th></tr><tr><td>70065</td><td>서지현</td><td>21/08/15</td><td>21/08/17</td></tr><tr><td>70066</td><td>한아름</td><td>21/09/02</td><td>21/09/03</td></tr><tr><td>70067</td><td>함용</td><td>21/08/23</td><td>21/08/25</td></tr></table>			대여 접수 번호	회원 이름	대여 시작일	대여 종료일	70065	서지현	21/08/15	21/08/17	70066	한아름	21/09/02	21/09/03	70067	함용	21/08/23	21/08/25
대여 접수 번호	회원 이름	대여 시작일	대여 종료일															
70065	서지현	21/08/15	21/08/17															
70066	한아름	21/09/02	21/09/03															
70067	함용	21/08/23	21/08/25															

2	SQL	담당자
문항	현재 대여 중인 용품 목록을 출력한다.	서동현
<pre>select pro_rentable_code from camp_product_management where pro_rentable_possibility='N';</pre>		
결과		
<pre>SQL> select pro_rentable_code 2 from camp_product_management 3 where pro_rentable_possibility='N'; PRO_RENTABLE_CODE ----- 101451 100020 100023 100574 100592 100622 101450 7 개의 행이 선택되었습니다.</pre>		

3	SQL	담당자
문항	캠핑 대여 용품 중 현재 대여 가능한 품목들을 확인한다.	서동현
select * from camp_product where camp_product_rentable_amount>0;		
결과		
<pre>CAMP_PRODUCT_NO CAMP_PRODUCT_CATEGORY_NO ----- CAMP_PRODUCT_NAME ----- CAMP_PRODUCT_RENTAL_FEE CAMP_PRODUCT_HOLD_AMOUNT CAMP_PRODUCT_RENTABLE_AMOUNT ----- CAMP_PRODUCT_DETAIL ----- 10185 1006 STAGER 포인트 해먹 40000 9 재질: 스테인리스, 알루미늄 173 개의 행이 선택되었습니다.</pre>		

4	SQL	담당자
문항	현재 대여 접수 중인 용품의 이름과 개수, 접수 날짜 및 대여한 회원의 번호와 이름을 출력한다.	김희진

```

select  rr.rental_receive_no "접수 번호",
        m.member_no "회원 번호",
        m.member_name "회원 이름",
        cp.camp_product_name "대여한 용품 이름",
        count(*) "개수",
        rr.rental_receive_date "대여 접수일"

from    member_info m,
        rental_receive rr,
        product_rental_record prr,
        camp_product_management cpm,
        camp_product cp

where   m.member_no = rr.member_no and
        rr.rental_receive_no = prr.rental_receive_no and
        prr.pro_rentable_code = cpm.pro_rentable_code and
        cpm.camp_product_no = cp.camp_product_no and
        m.member_no in (select member_no from rental_receive)

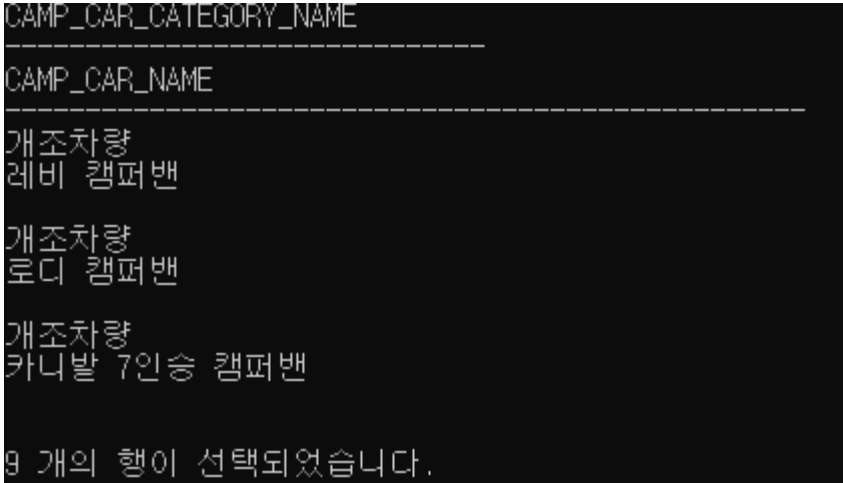
group by rr.rental_receive_no, m.member_no, m.member_name,
        cp.camp_product_name, rr.rental_receive_date

order by "대여 접수일";

```

결과

접수 번호	회원 번호	회원 이름	대여한 용품 이름	대여 접수일	개수
70002	800049	이소현	거실형 텐트	21/08/09	1
70002	800049	이소현	캠프타운 어디벤처5	21/08/09	1
70002	800049	이소현	폴라리스 실렌티움 렐렉스체어	21/08/09	2
70064	800038	강소라	거실형 텐트	21/08/11	1
70064	800038	강소라	버팔로 말라무3	21/08/11	1
70064	800038	강소라	비에프엘 버팔로 뉴 에코 렉타 타프	21/08/11	1

5	SQL	담당자
문항	'개조차량' 카테고리 내에서 대여 가능한 차량 목록을 확인한다.	서동현
<pre> select c.camp_car_category_name, s.camp_car_name from camp_car_category c, camp_car s where c.camp_car_category_no = s.camp_car_category_no and s.camp_car_rentable_amount>=1 and c.camp_car_category_no=2002; </pre>		
결과		
 <p>The screenshot shows the results of the SQL query in a dark-themed terminal. The output is as follows:</p> <pre> CAMP_CAR_CATEGORY_NAME ----- CAMP_CAR_NAME ----- 개조차량 레비 캠퍼밴 개조차량 로디 캠퍼밴 개조차량 카니발 7인승 캠퍼밴 9 개의 행이 선택되었습니다. </pre>		

6	SQL	담당자
문항	'구만산장 캠핑장'의 객실 목록을 확인한다.	서동현
<pre> select cri.camp_room_name,cri.camp_room_detail from camp_room_info cri, camp_ground cg where cri.camp_ground_no = cg.camp_ground_no and cg.camp_ground_name='구만산장 캠핑장'; </pre>		
결과		
<pre> CAMP_ROOM_NAME ----- CAMP_ROOM_DETAIL ----- 캠핑A동 트램폴린, 물놀이장, 운동시설, 화장실, 마트, 편의점 캠핑B동 트램폴린, 물놀이장, 운동시설, 화장실, 마트, 편의점 캠핑C동 트램폴린, 물놀이장, 운동시설, 화장실, 마트, 편의점 CAMP_ROOM_NAME ----- CAMP_ROOM_DETAIL ----- 황토방가족펜션 자락인 구만산에 이미 건강이 좋다고 잘알려진 황토방 </pre>		

7	SQL	담당자
문항	가구 카테고리의 대여 가능한 용품 목록을 확인한다.	서동현
<pre>select c.camp_product_category_name, s.camp_product_name from camp_product_category c, camp_product s where c.camp_product_category_no = s.camp_product_category_no and s.camp_product_rentable_amount>=1;</pre>		
결과		
<pre>CAMP_PRODUCT_CATEGORY_NAME ----- CAMP_PRODUCT_NAME ----- 가구 노마드 인디오 야전침대 가구 STAGER 포인트 해먹 173 개의 행이 선택되었습니다.</pre>		

8	SQL	담당자
문항	태안군에 위치한 캠핑장 목록을 확인한다.	서동현
select * from camp_ground where camp_ground_address_area like'%태안군%';		
결과		
<div>CAMP_GROUND_NO</div> <div>CAMP_GROUND_NAME</div> <div>CAMP_GROUND_ADDRESS_AREA</div> <div>CAMP_GROUND_DETAIL</div> <div>CAMP_GROUND_ADDRESS_DETAIL</div> <div>전기시설+하수도시설</div> <div>충청남도 태안군 원북면 방갈리 535</div> <div>8 개의 행이 선택되었습니다.</div>		

9	SQL	담당자
문항	가격이 저렴한 순으로 정렬한 용품 목록을 추출한다.	서동현
select * from camp_product order by camp_product_rental_fee;		
결과		
<pre>CAMP_PRODUCT_NO CAMP_PRODUCT_CATEGORY_NO ----- CAMP_PRODUCT_NAME ----- CAMP_PRODUCT_RENTAL_FEE CAMP_PRODUCT_HOLD_AMOUNT CAMP_PRODUCT_RENTABLE_AMOUNT ----- CAMP_PRODUCT_DETAIL ----- 10046 1002 대형 함마톤 접이식 캐노피 타프 80000 10 10 재질: 폴리에스터 173 개의 행이 선택되었습니다.</pre>		

10	SQL	담당자
문항	캠핑장 객실 상세 설명에 '카라반'을 포함하고 있는 객실 목록을 확인한다.	서동현
select * from camp_room_info where camp_room_detail like '%카라반%';		
결과		
<pre>CAMP_ROOM_INFO_NO CAMP_GROUND_NO CAMP_ROOM_NAME ----- CAMP_ROOM_RENTAL_FEE CAMP_ROOM_MAX_PEOPLE ----- CAMP_ROOM_DETAIL ----- CAMP_ROOM ----- 30063 3016 프리미엄1 140000 5 아름다운 자연경관,카라반 21/08/09</pre>		

11	SQL	담당자
문항	3인 이상 탑승 가능한 차량 목록(차 이름)을 확인한다.	서동현
<pre> select camp_car_name, camp_car_boarding_amount from camp_car where camp_car_boarding_amount>=3; </pre>		
결과		
<pre> SQL> select camp_car_name, camp_car_boarding_amount from camp_car where camp_car_boarding_a mount>=3; CAMP_CAR_NAME CAMP_CAR_BOARDING_AMOUNT ----- 바인스버그 모터홈 4 코반 마스터690 5 코반 마스터 3 스머프1호 캠퍼밴 4 캠봉씨 모터홈 4 그랜드 스타렉스 캠퍼밴 5 풀라티 모터홈 4 스텔스 모터홈 7 카니발 9인승 캠퍼밴 9 카니발 7인승 캠퍼밴 7 10 개의 행이 선택되었습니다. </pre>		

12	SQL	담당자
문항	<p>전화번호가 010-8825-8507인 회원의 반납 여부를 확인한다. (대여 접수 번호, 대여 종료일, 반납 여부, 반납일)</p> <pre> select rl.rental_receive_no "대여 접수 번호", rr.rental_finish_date "대여 종료일", f_return_ck(rl.rental_receive_no) "반납 여부", nvl(to_char(rl.return_date) , '-') "반납일" from return_list rl, rental_receive rr where rl.rental_receive_no = rr.rental_receive_no and rl.rental_receive_no = (select rr2.rental_receive_no from member_info mi, rental_receive rr2 where rr2.member_no = mi.member_no and mi.member_tel = '010-8825-8507');</pre>	김희진
결과		
<div> <div>대여 접수 번호</div> <div>대여 종료일</div> <div>반납</div> <div>반납일</div> </div> <div> <div>-----</div> <div>-----</div> <div>-----</div> <div>-----</div> </div> <div> <div>70033</div> <div>21/08/10</div> <div>Y</div> <div>21/08/11</div> </div>		

13	SQL	담당자
문항	<p>반납 테이블에 존재하지 않으면서 대여 종료일이 지난 회원의 회원 ID, 회원 이름, 주소, 전화번호를 확인한다.</p> <pre> select rr.rental_receive_no "대여 접수 번호", m.member_id "회원 ID" , m.member_name "회원 이름" , m.member_address "회원 주소" , m.member_tel "전화 번호" from member_info m, rental_receive rr where m.member_no = rr.member_no and not exists (select 1 from return_list where rental_receive_no = rr.rental_receive_no) and rr.rental_finish_date < sysdate; </pre>	김희진
결과		
<div> <div>대여 접수 번호</div> <div>회원 ID</div> <div>회원 이름</div> <div>회원 주소</div> <div>전화 번호</div> </div> <div> <div>-----</div> <div>-----</div> <div>-----</div> <div>-----</div> <div>-----</div> </div> <div> <div>70008</div> <div>T87102928</div> <div>최현정</div> <div>제주</div> <div>010-9321-2928</div> </div>		

14	SQL	담당자								
문항	800038번 회원에게 발급된 가상계좌 번호를 확인한다.	김희진								
<pre>select (select member_name from member_info where member_no = 800038) "회원 이름", bank_name "은행명", virtual_account_number "가상계좌번호", depositor "예금주" from cash_deposit_payment where payment_info_no = (select pi.payment_info_no from payment_info pi, rental_receive rr where pi.rental_receive_no = rr.rental_receive_no and rr.member_no = 800038);</pre>										
결과										
<table><tr><th>회원 이름</th><th>은행명</th><th>가상계좌번호</th><th>예금주</th></tr><tr><td>강소라</td><td>하나</td><td>10026548019</td><td>강소라</td></tr></table>			회원 이름	은행명	가상계좌번호	예금주	강소라	하나	10026548019	강소라
회원 이름	은행명	가상계좌번호	예금주							
강소라	하나	10026548019	강소라							

1	PL/SQL	담당자
문항	<p>대여 접수 번호와 결제 수단을 받아서 모든 대여 물품의 대여료 합계를 계산한 후, 대여 접수 번호와 결제 코드, 총 대여료를 결제 정보 테이블에 저장한다. 단, 휴대폰 소액 결제인 경우 총 결제 금액이 100만원 이상이면 예외 처리</p> <pre> create or replace procedure p_payment_in (v_rr_no rental_receive.rental_receive_no%type, v_pay payment_way.payment_type%type) is v_pay_code payment_way.payment_code%type; v_pro_fee payment_info.payment_total_fee%type; v_car_fee payment_info.payment_total_fee%type; v_room_fee payment_info.payment_total_fee%type; v_total_fee payment_info.payment_total_fee%type; phone_limit_fee EXCEPTION; begin select payment_code into v_pay_code from payment_way where payment_type = v_pay; select nvl((select sum(pro_total_rental_fee) from product_rental_record where rental_receive_no = v_rr_no) ,0) into v_pro_fee from dual; select nvl((select sum(car_total_rental_fee) from car_rental_record where rental_receive_no = v_rr_no) ,0) into v_car_fee from dual; select nvl((select sum(room_total_rental_fee) from room_rental_record where rental_receive_no = v_rr_no) ,0) into v_room_fee from dual; v_total_fee := v_pro_fee + v_car_fee + v_room_fee; if v_pay_code = 3333 and v_total_fee > 1000000 then raise phone_limit_fee; end if; insert into payment_info (payment_info_no, rental_receive_no, payment_code, payment_total_fee) values (s_payment_info_no.nextval, v_rr_no, v_pay_code, v_total_fee); EXCEPTION when phone_limit_fee then dbms_output.put_line('휴대폰 소액 결제는 100만원 이하의 금액만 가능합니다.');</pre> <p>end; /</p>	김희진

2	PL/SQL	담당자										
문항	결제 정보 번호, 카드사, 카드 번호를 받고, 카드 결제 테이블에 저장한다. 단, 해당 결제 정보의 결제 코드가 카드 결제가 아닐 경우 예외처리	최건호										
<pre>create or replace procedure p_card_payment_tab (v_card_number card_payment.card_number%type, v_card_company card_payment.card_company%type, v_payment_info_no payment_info.payment_info_no%type) is v_payment_code payment_info.payment_code%type; system_check_insert_fail exception; begin select payment_code into v_payment_code from payment_info where payment_info_no=v_payment_info_no; if v_payment_code != 1111 then raise system_check_insert_fail; else insert into card_payment (card_payment_no,card_payment_date,card_number, card_company,payment_info_no) values (s_card_payment_no.nextval,sysdate,v_card_number, v_card_company,v_payment_info_no); end if; exception when system_check_insert_fail then dbms_output.put_line('결제 코드가 다릅니다.'); end; /</pre>												
결과												
<pre>SQL> exec p_card_payment_tab (4016075514942875, 'BC', 90002) PL/SQL 처리가 정상적으로 완료되었습니다. SQL> select *from card_payment;</pre> <table><tr><th>CARD_PAYMENT_NO</th><th>CARD_PAY</th><th>CARD_NUMBER</th><th>CARD_COMPANY</th><th>PAYMENT_INFO_NO</th></tr><tr><td>91030</td><td>21/08/11</td><td>4016075514942875</td><td>BC</td><td>90002</td></tr></table> <pre>SQL> exec p_card_payment_tab (4016075514942875, 'BC', 90020); 결제 코드가 다릅니다.</pre>			CARD_PAYMENT_NO	CARD_PAY	CARD_NUMBER	CARD_COMPANY	PAYMENT_INFO_NO	91030	21/08/11	4016075514942875	BC	90002
CARD_PAYMENT_NO	CARD_PAY	CARD_NUMBER	CARD_COMPANY	PAYMENT_INFO_NO								
91030	21/08/11	4016075514942875	BC	90002								

3	PL/SQL	담당자
문항	결제 정보 번호, 은행명, 예금주를 받고, 무통장 입금 결제 테이블에 저장한다. 단, 해당 결제 정보의 결제 코드가 무통장 입금 결제가 아닐 경우 예외 처리	김희진
	<pre> create or replace procedure p_cash_in (v_pay_no payment_info.payment_info_no%type, v_bank_name cash_deposit_payment.bank_name%type, v_depositor cash_deposit_payment.depositor%type) is pay_code_match EXCEPTION; v_p_code payment_info.payment_code%type; begin select payment_code into v_p_code from payment_info where payment_info_no = v_pay_no; if v_p_code != 2222 then raise pay_code_match; else insert into cash_deposit_payment (cash_deposit_payment_no, bank_name, virtual_account_number, depositor, payment_info_no) values (s_cash_deposit_payment_no.nextval, v_bank_name, s_virtual_account_number.nextval, v_depositor, v_pay_no); end if; EXCEPTION when pay_code_match then dbms_output.put_line('선택한 결제 방식이 아닙니다.');</pre> <p>end;</p> <p>/</p>	

4	PL/SQL	담당자
문항	<p>결제 정보 번호, 통신사, 휴대폰 번호를 받고, 휴대폰 소액 결제 테이블에 저장한다. 단, 해당 결제 정보의 결제 코드가 휴대폰 소액 결제가 아닐 경우 예외처리</p>	최건호
	<pre> create or replace procedure p_phone_payment_tab (v_phone_number phone_payment.phone_number%type, v_telecom phone_payment.telecom%type, v_payment_info_no payment_info.payment_info_no%type) is v_payment_code payment_way.payment_code%type; system_check_insert_fail exception; begin select payment_code into v_payment_code from payment_info where payment_info_no=v_payment_info_no ; if v_payment_code != 3333 then raise system_check_insert_fail; else insert into phone_payment (phone_payment_no,phone_payment_date,telecom, phone_number,payment_info_no) values (s_phone_payment_no.nextval,sysdate,v_telecom, v_phone_number,v_payment_info_no); end if; exception when system_check_insert_fail then dbms_output.put_line(' 결제 코드가 다릅니다.'); commit; end; / </pre>	
	결과	
	<pre> SQL> exec p_phone_payment_tab('010-8484-1062','KT',90021); PL/SQL 처리가 정상적으로 완료되었습니다. SQL> select *from phone_payment; PHONE_PAYMENT_NO PHONE_PA TELECOM PHONE_NUMBER PAYMENT_INFO_NO ----- 93020 21/08/11 KT 010-8484-1062 90021 SQL> exec p_phone_payment_tab('010-8484-1062','KT',90002) 결제 코드가 다릅니다. </pre>	

5	PL/SQL	담당자
문항	회원 아이디와 대여 시작일, 종료일을 받아서 대여 접수 테이블에 저장한다.	김희진
<pre> create or replace procedure p_rental_receive_in (v_mem_id member_info.member_id%type, v_rent_start rental_receive.rental_start_date%type, v_rent_finish rental_receive.rental_finish_date%type) is v_mem_no member_info.member_no%type; begin if v_rent_start <= sysdate then dbms_output.put_line('대여 시작일에는 ' sysdate ' 이후의 날짜만 입력이 가능합니다.');</pre> <pre> elsif v_rent_finish >= (v_rent_start + 6) then dbms_output.put_line('6일 이상 대여할 수 없습니다.');</pre> <pre> elsif v_rent_start >= v_rent_finish then dbms_output.put_line('대여 종료일을 다시 선택해주세요.');</pre> <pre> else select member_no into v_mem_no from member_info where member_id = v_mem_id; insert into rental_receive (rental_receive_no, member_no, rental_start_date, rental_finish_date) values (s_rental_receive_no.nextval, v_mem_no, v_rent_start, v_rent_finish); end if; commit; end; / </pre>		

6	PL/SQL	담당자
문항	<p>접수 번호와 대여할 용품 정보를 받고, 수량 및 기간에 따른 총 대여료를 계산한 후 용품 대여 내역 테이블에 저장한다.</p> <pre> create or replace procedure p_pro_rental_record_in (v_rr_no rental_receive.rental_receive_no%type, v_pro_code camp_product_management.pro_rentable_code%type) is v_rent_day number(1); v_total_fee product_rental_record.pro_total_rental_fee%type; v_pro_fee camp_product.camp_product_rental_fee%type; begin select to_number(rental_finish_date-rental_start_date) into v_rent_day from rental_receive where rental_receive_no = v_rr_no; select camp_product_rental_fee into v_pro_fee from camp_product where camp_product_no = (select camp_product_no from camp_product_management where pro_rentable_code = v_pro_code); v_total_fee := v_pro_fee * v_rent_day; insert into product_rental_record (product_rental_record_no, rental_receive_no, pro_total_rental_fee, pro_rentable_code) values (s_product_rental_record_no.nextval, v_rr_no, v_total_fee, v_pro_code); commit; end; / </pre>	김희진

7	PL/SQL	담당자
문항	접수 번호와 대여할 차량 정보를 받고 대여 기간에 따른 총 대여료를 계산한 후 차량 대여 내역 테이블에 저장한다.	최건호
	<pre> create or replace procedure p_car_rental_record_tab (v_car_rentable_code car_rental_record.car_rentable_code%type, v_rental_receive_no car_rental_record.rental_receive_no%type) is v_rental_day number(1); v_car_total_rental_fee car_rental_record.car_total_rental_fee%type; v_camp_car_rental_fee camp_car.camp_car_rental_fee%type; begin select to_number(rental_finish_date - rental_start_date) into v_rental_day from rental_receive where rental_receive_no=v_rental_receive_no; select camp_car_rental_fee into v_camp_car_rental_fee from camp_car where camp_car_no = (select camp_car_no from camp_car_management where car_rentable_code=v_car_rentable_code); v_car_total_rental_fee :=v_camp_car_rental_fee*v_rental_day; insert into car_rental_record (car_rental_record_no,car_total_rental_fee,car_rentable_code,rental_receive_no) values (s_car_rental_record_no.nextval,v_car_total_rental_fee, v_car_rentable_code,v_rental_receive_no); commit; end; / </pre>	
결과		
	<pre> SQL> exec p_car_rental_record_tab(200100,70033) PL/SQL 처리가 정상적으로 완료되었습니다. SQL> select *from car_rental_record; CAR_RENTAL_RECORD_NO CAR_TOTAL_RENTAL_FEE CAR_RENTABLE_CODE RENTAL_RECEIVE_NO ----- 50020 498000 200100 70033 </pre>	

8	PL/SQL	담당자
문항	접수 번호와 대여할 객실, 사용할 인원 수 정보를 받고 대여 기간에 따른 총 대여료를 계산한 후 캠핑장 대여 내역 테이블에 저장한다.	최건호

```

create or replace procedure p_room_rental_record_tab
(
    v_camp_room_info_no      room_rental_record.camp_room_info_no%type,
    v_rental_receive_no      room_rental_record.rental_receive_no%type,
    v_room_use_people_number room_rental_record.room_use_people_number%type
)
is
    v_rental_day              number(1);
    v_room_total_rental_fee   room_rental_record.room_total_rental_fee%type;
    v_camp_room_rental_fee    camp_room_info.camp_room_rental_fee%type;

begin
    select to_number(rental_finish_date - rental_start_date) into v_rental_day
        from rental_receive
        where rental_receive_no=v_rental_receive_no;

    select camp_room_rental_fee into v_camp_room_rental_fee
        from camp_room_info
        where camp_room_info_no=v_camp_room_info_no;

    v_room_total_rental_fee :=v_camp_room_rental_fee * v_rental_day;

    insert into room_rental_record
        (room_rental_record_no,room_total_rental_fee,camp_room_info_no,
        rental_receive_no,room_use_people_number)
    values
        (s_room_rental_record_no.nextval,v_room_total_rental_fee,
        v_camp_room_info_no,v_rental_receive_no,v_room_use_people_number);

end;
/

```

결과

```

SQL> exec p_room_rental_record_tab(30075,70033,2)
PL/SQL 처리가 정상적으로 완료되었습니다.
SQL> select *from room_rental_record;
ROOM_RENTAL_RECORD_NO ROOM_TOTAL_RENTAL_FE CAMP_ROOM_INFO_NO RENTAL_RECEIVE_NO ROOM_USE_PEOPLE_NUMBER
-----
60001 120000                                30075                                70033                                2

```

9	PL/SQL	담당자
문항	<p>결제 이전에 대여를 취소할 경우, 대여 접수 번호를 받아 모든 물품의 대여 내역을 삭제한다.</p> <pre> create or replace procedure p_recieve_candle (v_rr_no rental_receive.rental_receive_no%type) is v_p_ck number(1); begin select nvl((select 1 from payment_info where rental_receive_no = v_rr_no) , 0) into v_p_ck from dual; if v_p_ck = 0 then delete rental_receive where rental_receive_no = v_rr_no; delete product_rental_record where rental_receive_no = v_rr_no; delete car_rental_record where rental_receive_no = v_rr_no; delete room_rental_record where rental_receive_no = v_rr_no; else dbms_output.put_line('결제된 대여 내역은 삭제할 수 없습니다.');</pre>	김희진

10	PL/SQL	담당자
문항	물품 등록 / 삭제 시 보유 개수와 대여 가능 개수를 변경한다.	최건호

```

create or replace trigger t_product_de_in
before insert or delete on camp_product_management
for each row
begin
    if deleting then
        update camp_product
        set camp_product_rentable_amount = camp_product_rentable_amount-1,
            camp_product_hold_amount = camp_product_hold_amount-1
        where camp_product_no = :old.camp_product_no;
    else
        update camp_product
        set camp_product_rentable_amount = camp_product_rentable_amount+1,
            camp_product_hold_amount = camp_product_hold_amount+1
        where camp_product_no = :new.camp_product_no;
    end if;
end;
/

```

결과

```

SQL> update camp_product set camp_product_rentable_amount = camp_product_rentable_amount+1,
2   camp_product_hold_amount = camp_product_hold_amount+1
3   where camp_product_no=10164;

1 행이 갱신되었습니다.

```

```

SQL> update camp_product set camp_product_rentable_amount = camp_product_rentable_amount-1,
2   camp_product_hold_amount = camp_product_hold_amount-1
3   where camp_product_no=10165;

1 행이 갱신되었습니다.

```

CAMP_PRODUCT_NO	CAMP_PRODUCT_CATEGORY_NO	CAMP_PRODUCT_NAME	CAMP_PRODUCT_RENTAL_FEE	CAMP_PRODUCT_HOLD_AMOUNT	CAMP_PRODUCT_RENTABLE_AMOUNT
CAMP_PRODUCT_DETAIL					
10164	1006	노마드 인디오 야전침대	34000	11	11
재질 : 스테인리스, 알루미늄					
10165	1006	STAGER 포인트 해먹	40000	9	9

11	PL/SQL	담당자
문항	<p>품목 대여 시 해당 품목의 대여 가능 여부를 변경한다.</p> <pre> create or replace trigger t_pro_rental_psb after insert or update or delete of pro_rentable_code on product_rental_record for each row begin if inserting then update camp_product_management set pro_rentable_possibility = 'N' where pro_rentable_code = :new.pro_rentable_code; elsif deleting then update camp_product_management set pro_rentable_possibility = 'Y' where pro_rentable_code = :old.pro_rentable_code; elsif updating then update camp_product_management set pro_rentable_possibility = 'Y' where pro_rentable_code = :old.pro_rentable_code; update camp_product_management set pro_rentable_possibility = 'N' where pro_rentable_code = :new.pro_rentable_code; end if; end; / </pre>	김희진

12	PL/SQL	담당자
문항	반납이 확인된 물품의 경우 대여 가능 여부를 변경한다.	최건호
<pre> create or replace trigger t_return_in after insert on return_list for each row begin update camp_product_management set pro_rentable_possibility='Y' where pro_rentable_code = nvl((select pro_rentable_code from product_rental_record where rental_receive_no = :new.rental_receive_no),0); update camp_car_management set car_rentable_possibility='Y' where car_rentable_code = nvl((select car_rentable_code from car_rental_record where rental_receive_no = :new.rental_receive_no),0); end; / </pre>		
결과		
<pre> PRO_RENTABLE_CODE P CAMP_PRODUCT_NO PRO_REGI ----- 101448 Y 10145 21/08/09 101449 Y 10145 21/08/09 101450 N 10146 21/08/09 </pre> <pre> RETURN_LIST_NO RENTAL_RECEIVE_NO RETURN_D ----- 900023 70033 21/08/11 </pre> <pre> PRO_RENTABLE_CODE P CAMP_PRODUCT_NO PRO_REGI ----- 101448 Y 10145 21/08/09 101449 Y 10145 21/08/09 101450 Y 10146 21/08/09 </pre>		

13	PL/SQL	담당자
문항	대여 가능 여부의 변동에 따라 대여 가능 개수를 변경한다.	김희진
<pre> create or replace trigger t_pro_rental_pa after update of pro_rentable_possibility on camp_product_management for each row begin if :new.pro_rentable_possibility = 'Y' then update camp_product set camp_product_rentable_amount = camp_product_rentable_amount+1 where camp_product_no = :new.camp_product_no; else update camp_product set camp_product_rentable_amount = camp_product_rentable_amount-1 where camp_product_no = :old.camp_product_no; end if; end; / </pre>		

14	PL/SQL	담당자
문항	캠핑장 삭제 시 연관된 객실 정보도 삭제한다.	최건호
<pre> create or replace trigger t_camp_ground_de after delete on camp_ground for each row begin delete camp_room_info where camp_ground_no= :old.camp_ground_no; end; / </pre>		
결과		
<pre> SQL> delete camp_ground where camp_ground_no=3020 2 ; 1 행이 삭제되었습니다. SQL> select *from camp_room_info where camp_room_info_no=3020; 선택된 레코드가 없습니다. </pre>		

15	PL/SQL	담당자
문항	대여 종료일이 오늘 날짜인 대여 내역을 추출해 반납 테이블에 저장한다.	최건호

```

create or replace procedure p_return_in
(
    v_rental_receive_no      rental_receive.rental_receive_no%type
)
is
    v_rental_finish_date      rental_receive.rental_finish_date%type;
    system_check_insert_fail exception;
begin
    if v_rental_finish_date = sysdate then

        insert into return_list (return_list_no,rental_receive_no)
        values (s_return_list_no.nextval,v_rental_receive_no);
        end if;

end;
/

```

결과

```

RENTAL_RECEIVE_NO  MEMBER_NO  RENTAL_R  RENTAL_S  RENTAL_F
-----
          70039      800051  21/08/10  21/08/11  21/08/13
          70002      800049  21/08/09  21/08/10  21/08/13
          70003      800016  21/08/09  21/08/10  21/08/14
          70064      800038  21/08/11  21/08/15  21/08/19
          70008      800044  21/08/05  21/08/07  21/08/09
          70032      800011  21/08/04  21/08/07  21/08/09
          70033      800021  21/08/04  21/08/07  21/08/11

```

7 개의 행이 선택되었습니다.

SQL> exec p_return_in (70033)

PL/SQL 처리가 정상적으로 완료되었습니다.

SQL> select *from return_list;

```

RETURN_LIST_NO  RENTAL_RECEIVE_NO  RETURN_D
-----
          900023              70033  21/08/11

```

16	PL/SQL	담당자
문항	회원 전화번호를 받아서 해당 회원의 대여 내역을 반납 테이블에 저장한다.	김희진
<pre> create or replace procedure p_return_in (v_re_phone member_info.member_tel%type) is v_rr_no rental_receive.rental_receive_no%type; begin select rr.rental_receive_no into v_rr_no from rental_receive rr, member_info m where m.member_no = rr.member_no and m.member_tel = v_re_phone; insert into return_list (return_list_no, rental_receive_no) values (s_return_list_no.nextval, v_rr_no); EXCEPTION when no_data_found then dbms_output.put_line('대여 내역이 없습니다.');</pre> <pre> commit; end; /</pre>		

17	PL/SQL	담당자
문항	대여 접수 시 이미 대여한 기록이 있고, 반납 기록은 없을 경우 예외처리 한다.	김희진
<pre> create or replace trigger t_receive_ck before insert on rental_receive for each row declare v_receive_ck number(1); v_return_ck number(1); begin select nvl((select 1 from rental_receive where member_no = :new.member_no) , 0) into v_receive_ck from dual; select nvl((select 1 from return_list where rental_receive_no = (select rental_receive_no from rental_receive where member_no = :new.member_no)) ,0) into v_return_ck from dual; if v_receive_ck = 1 and v_return_ck = 0 then raise_application_error(-20001, '기존 대여 내역이 존재합니다.');</pre> <p>end if;</p> <p>end;</p> <p>/</p>		

18	PL/SQL	담당자
문항	대여 내역 번호를 입력하면 반납 여부를 확인하여 'Y' 또는 'N' 값을 리턴해주는 함수	김희진
<pre> create or replace function f_return_ck (v_rr_no number) return varchar2 is v_ck number(1); begin select nvl((select 1 from return_list where rental_receive_no = v_rr_no) , 0) into v_ck from dual; if v_ck = 1 then return 'Y'; else return 'N'; end if; end; / </pre>		