```
------------------------------------------------
--Exception Örnek
------------------------------------------------

declare
    v_name varchar2(25);
begin
    select ad into v_name from personel
    where unvan = 'UZMAN';

    dbms_output.put_line(v_name);
end;


------------------------------------------------
--Exception Handling (Hata Yönetme)
------------------------------------------------

set serveroutput on;
declare
    v_name varchar2(25);
begin
    select ad into v_name from personel
    where unvan = 'UZMAN';

    dbms_output.put_line(v_name);

    exception
        when too_many_rows then
            dbms_output.put_line('Birden fazla satır geldi');
end;


------------------------------------------------
--Hata Yakalama Predefined
------------------------------------------------

declare
    v_name varchar2(25);
begin
    select ad into v_name from personel
--    where unvan = 'UZMAN';
    where personel_id = 5010;

    dbms_output.put_line(v_name);

    dbms_output.put_line('Bölme İşlemi: '|| 1/0);

    exception
        when TOO_MANY_ROWS then
            dbms_output.put_line('Birden fazla satır geldi');
        when OTHERS then
            dbms_output.put_line('Başka bir hata oluştu');
end;
```

```
--------------------------------------------
--Hata Yakalama Internally
--------------------------------------------

set serveroutput on;
declare

    e_null_insert exception;

    pragma exception_init(e_null_insert, -01400);

begin

    insert into bolgeler (bolge_kodu, bolge_adi)
    values(10, null);

    exception

        when e_null_insert then

            dbms_output.put_line('Null olmayan bir değer girmelisiniz!');

end;


--------------------------------------------
--Hata Yakalama - Fonksiyonlar
--------------------------------------------

create table hata_bilgileri
(
    tarih date,
    hata_kodu number,
    hata_bilgi varchar2(1000)
)

declare
    v_name varchar2(25);
    v_ecode number;
    v_emesg varchar2(1000);

begin
    select ad into v_name from personel
    where unvan = 'UZMAN';
--    where personel_id = 5010;

    dbms_output.put_line('Bölme İşlemi: '|| 1/0);

    exception
        when OTHERS then
            --rollback;
            v_ecode := SQLCODE;
            v_emesg := SQLERRM;
            insert into hata_bilgileri
            values (sysdate, v_ecode, v_emesg);
end;
```

```
----------------------------------------------
--Hata Yakalama - Kullanıcı Tanımlı
----------------------------------------------

declare
    v_maas number := &maas_degeri;
    e_maas_hatasi exception;

begin

    if v_maas > 20000 then
        raise e_maas_hatasi;
    end if;

    insert into personel (personel_id, ad, soyad, maas)
    values(1000, 'Esra', 'Yılmaz', v_maas);

    exception
        when e_maas_hatasi then
            dbms_output.put_line('Maaş değeri 20000 den büyük olamaz');
end;


----------------------------------------------

set serveroutput on;
declare
    v_dep pls_integer := 105;
    v_per_sayisi pls_integer;
    e_departman exception;

begin

    select count(*) into v_per_sayisi from personel
    where dept_id = v_dep;

    if v_per_sayisi < 5 then
        raise e_departman;
    end if;

    exception
        when e_departman then
            dbms_output.put_line('Dep. No: '|| v_dep ||', Personel sayısı: ' || v_per_sayisi);
end;
```

```
-----------------------------------------------
--Alt Bloklarla Hata Yönetme
-----------------------------------------------

declare
    e_departman exception;
    e_adet exception;

    cursor c_depart is
        select dept_id, count(*) adet
        from personel group by dept_id;
begin

    for row_d in c_depart loop
        begin

            if row_d.dept_id = 104 then
                raise e_departman;
            end if;

            dbms_output.put_line('[---]Dep. No: '|| row_d.dept_id ||',Adet: '||row_d.adet);

            if row_d.adet < 5 then
                raise e_adet;
            end if;

            exception
                when e_adet then
                dbms_output.put_line('[XXX]Dep. No: '|| row_d.dept_id ||',Adet: '||row_d.adet);
        end;
    end loop;

    exception
        when e_departman then
            dbms_output.put_line('!!! GÜVENLİK İHLALİ !!!');
end;

-----------------------------------------------
--Raise_Application_Error
-----------------------------------------------

declare
    v_deger number := &deger;
begin
    if v_deger >= 100 then
        raise_application_error(-20001, '100 den küçük bir değer girmelisiniz...');
    else
        dbms_output.put_line('Teşekkürler...');

    end if;
end;

-----------------------------------------------

declare
    v_prim number;
begin

    select prim into v_prim from personel
    where personel_id = 5009;

    if v_prim is null then
        raise_application_error(-20005, 'Personel primi hak ediyor :)');
    else
        dbms_output.put_line('Prim Değeri:' || v_prim);

    end if;
end;
```

```
---------------------------------------------

declare
    e1 exception;
    pragma exception_init (e1, -20001);
    e2 exception;
    pragma exception_init (e2, -20002);
    e3 exception;
    pragma exception_init (e3, -20003);
    v_deger number := 1;

begin
    begin
        if v_deger = 1 then
            raise_application_error(-20001,'Hata Oluştu: No==> 1');
        elsif v_deger = 2 then
            raise_application_error(-20002,'Hata Oluştu: No==> 2');
        else
            raise_application_error(-20003,'Hata Oluştu: No==> 1 ve 2 değil');
        end if;
        -- iç blokta hata yakalanıyor
        exception
            when e1 then
                dbms_output.put_line('Hata Yakalandı: No==> 1');
    end;

    -- dış blokta hata yakalanıyor
    exception
        when e2 then
            dbms_output.put_line('Hata Yakalandı: No==> 2');
end;
```

```
----------------------------------------------------------
--ALIŞTIRMALARIN CEVAPLARI
----------------------------------------------------------


---------------------------------------------
--Hata Yakalama Predefined
---------------------------------------------

declare
    v_car_name varchar2(50) := 'Ferrari';
    v_mesaj varchar2(100);
begin


    select brand || ' is found in the table' into v_mesaj
    from cars where brand = v_car_name;

    dbms_output.put_line(v_mesaj);

    exception
        when NO_DATA_FOUND then
            dbms_output.put_line(v_car_name || ' is not found');

end;
/

------------------------------------------
--Hata Yakalama – Kullanıcı Tanımlı
------------------------------------------

declare
    v_budget number := 15000;
    v_car_brand varchar2(50) := 'Peugeot';
    v_car_price number;
    e_price_exceed exception;

begin

    select price into v_car_price from cars
    where brand = v_car_brand;

    if v_car_price > v_budget then
        raise e_price_exceed;
    else
        dbms_output.put_line('Tebirkler... Bu arabayı alabilirsiniz :)');
    end if;

    --Kredi işlemleri...

    exception
        when e_price_exceed then
            dbms_output.put_line('Üzgünüz! Bu arabanın fiyatı bütçenizi aşıyor :(');
end;
```

```
----------------------------------------------
--Alt Bloklarla Hata Yönetme – Alıştırma
----------------------------------------------

declare

    type v_colors_type is table of varchar2(50);
    v_colors v_colors_type;

    v_color varchar2(50);
begin

    v_colors := v_colors_type('Blue', 'Dark Green', 'Yellow');

    for i in v_colors.first..v_colors.last
    loop

        begin
            select color || ' is found in the table' into v_color
            from colors where color = v_colors(i);

            dbms_output.put_line(v_color);

            exception
                when NO_DATA_FOUND then
                    dbms_output.put_line(v_colors(i) || ' is not found');

        end;
    end loop;

end;
/
```

```
------------------------------------------
--Raise_Application_Error
------------------------------------------

declare
    e_stock_price_exceed_level1 exception;
    pragma exception_init (e_stock_price_exceed_level1, -20001);

    e_stock_price_exceed_level2 exception;
    pragma exception_init (e_stock_price_exceed_level2, -20002);

    e_stock_price_exceed_level3 exception;
    pragma exception_init (e_stock_price_exceed_level3, -20003);

    cursor c_products is
        select product_name,
            unit_price*units_in_stock as stock_price
        from products;
    v_prod_info varchar2(150);

begin

    for v_product_row in c_products loop

        v_prod_info := v_product_row.product_name||': '||v_product_row.stock_price;

        begin
            if v_product_row.stock_price between 2000 and 3000 then
                --raise e_stock_price_exceed_level1;
                raise_application_error(-20001,'Alarm!!! Stock Price Level1: ' || v_prod_info);
            elsif v_product_row.stock_price between 3000 and 4000 then
                --raise e_stock_price_exceed_level2;
                raise_application_error(-20002,'Alarm!!! Stock Price Level2: ' || v_prod_info);
            elsif v_product_row.stock_price > 4000 then
                raise_application_error(-20003,'Critical Alarm!!! Stock Price Level3: ' ||
v_prod_info);
            end if;

            exception
                when e_stock_price_exceed_level1 then
                    dbms_output.put_line('!!!Alarm!!! Stock Price Level1: ' || v_prod_info);
                when e_stock_price_exceed_level2 then
                    dbms_output.put_line('!!!Alarm!!! Stock Price Level2: ' || v_prod_info);
                when e_stock_price_exceed_level3 then
                    dbms_output.put_line(SQLERRM);
        end;
    end loop;

end;
```