i2i Systems
innovation to integration

i2i Academy

Training Document

Topic	SQL Language Fundamentals I
Document Name	PLSQL01-EX-08

Document Difficulty Level			
Beginner	**Junior**	**Senior**	**Expert**
□	■	□	□

Document History

Date	Author	Ver	Comments
20.09.2013	Onur Bayram	1.0	Initial Draft
03.10.2013	Erman İbrişim	1.1	Sample Outputs and Rules Added

SQL Language Fundamentals I

Exercise PLSQL01-EX-08:

Definiton : Write a PL/SQL program that prints out string in xml format.

Input String : “acbbcadefghkkhgfed”

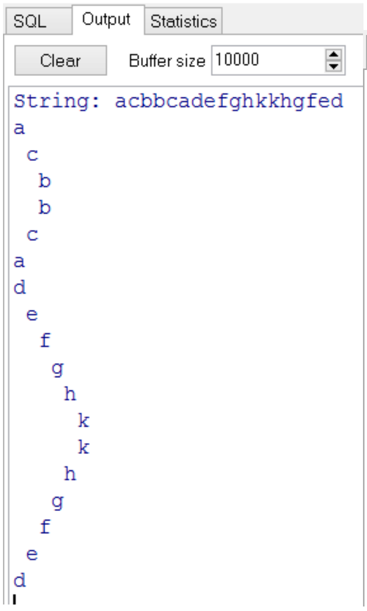
Input String: “abbcddfggfca”

Rules: -Maximum string length is 30.

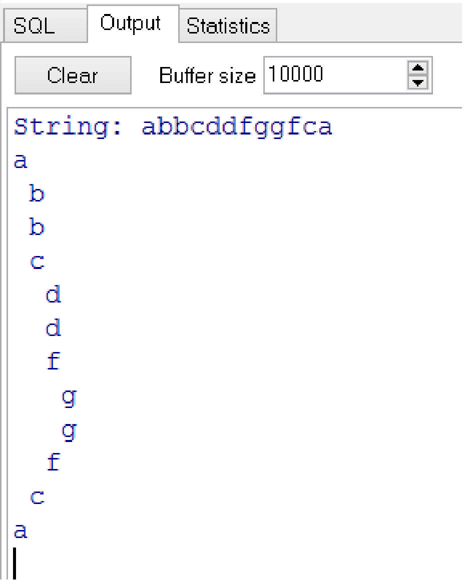
-No duplicate characters in string like ‘abbcddfaafca’ or ‘defgddgfed’.

-String can’t have space character or any numeric characters.

Sample Output :



```
String: acbbcadefghkkhgfed
a
 c
  b
  b
 c
a
d
 e
  f
   g
    h
     k
     k
    h
   g
  f
 e
d
```



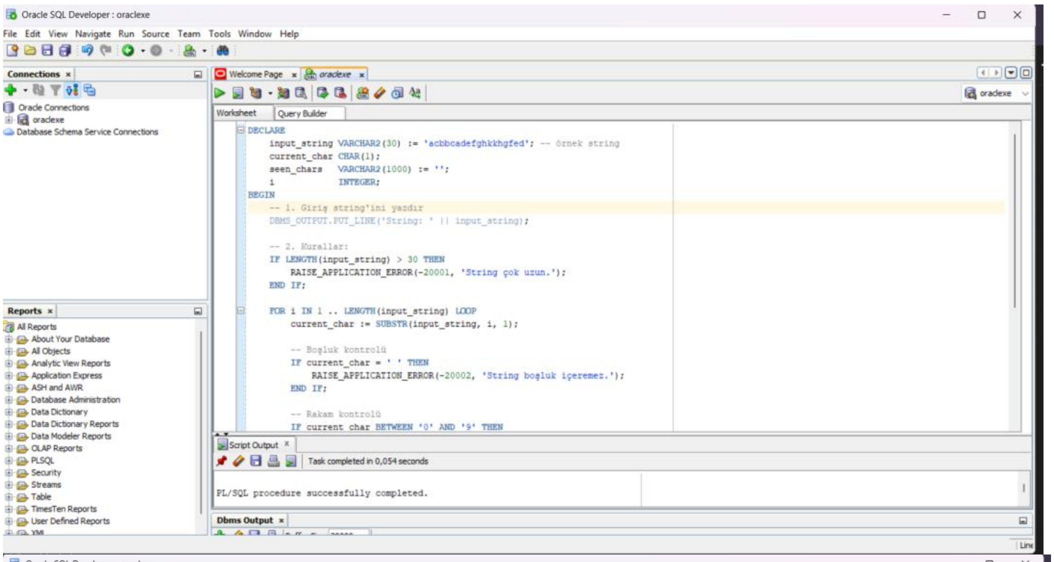
```
String: abbcddfggfca
a
 b
 b
 c
  d
  d
  f
   g
   g
  f
 c
a
```

Objectives : To learn SQL string functions and conditional claueses(IF THEN ELSE), loops(FOR, WHILE)

Solution of PLSQL01-EX-08:

Kullanılan kod:

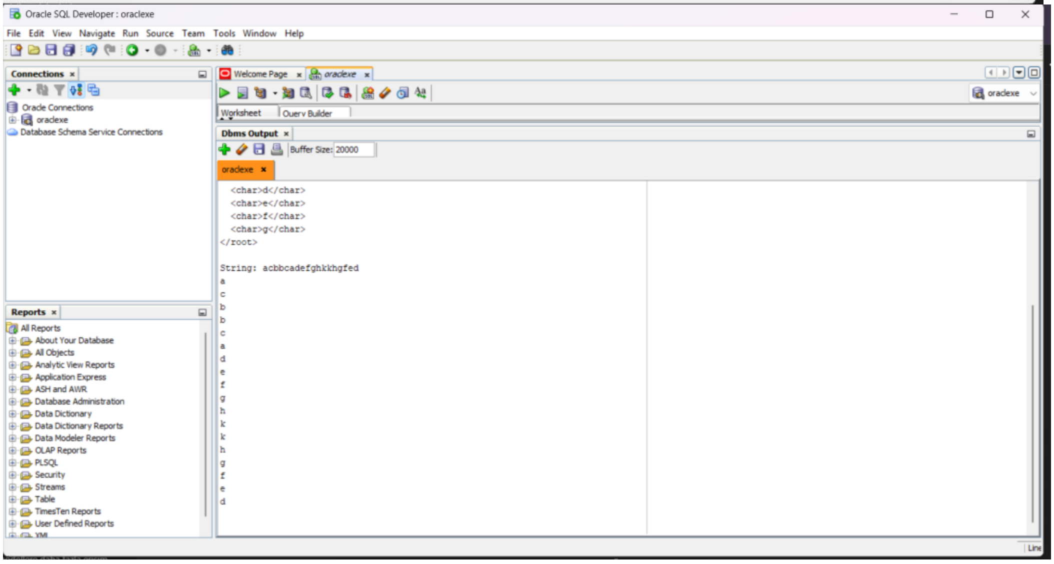
```
DECLARE input_string VARCHAR2(30) := 'acbbcadefghkkhgfed'; -- örnek string current_char CHAR(1);
seen_chars VARCHAR2(1000) := ''; i INTEGER; BEGIN -- 1. Giriş string'ini yazdır
DBMS_OUTPUT.PUT_LINE('String: ' || input_string);

-- 2. Kurallar:
IF LENGTH(input_string) > 30 THEN
    RAISE_APPLICATION_ERROR(-20001, 'String çok uzun.');
END IF;

FOR i IN 1 .. LENGTH(input_string) LOOP
    current_char := SUBSTR(input_string, i, 1);

    -- Boşluk kontrolü
    IF current_char = ' ' THEN
        RAISE_APPLICATION_ERROR(-20002, 'String boşluk içeremez.');
    END IF;

    -- Rakam kontrolü
    IF current_char BETWEEN '0' AND '9' THEN
        RAISE_APPLICATION_ERROR(-20003, 'String rakam içeremez.');
    END IF;

    -- Tekrar eden karakter kontrolü
    IF INSTR(seen_chars, current_char) > 0 THEN
        -- Tekrar var, sadece yazdır, ama hata üretme
        NULL;
    ELSE
        seen_chars := seen_chars || current_char;
    END IF;

    -- Karakteri yazdır
    DBMS_OUTPUT.PUT_LINE(current_char);
END LOOP;


END;
```



```
DECLARE
    input_string VARCHAR2(30) := 'acbbcadefghkkhgfed'; -- örnek string
    current_char CHAR(1);
    seen_chars   VARCHAR2(1000) := '';
    i            INTEGER;
BEGIN
    -- 1. Giriş string'ini yazdır
    DBMS_OUTPUT.PUT_LINE('String: ' || input_string);

    -- 2. Kurallar:
    IF LENGTH(input_string) > 30 THEN
        RAISE_APPLICATION_ERROR(-20001, 'String çok uzun.');
    END IF;

    FOR i IN 1 .. LENGTH(input_string) LOOP
        current_char := SUBSTR(input_string, i, 1);

        -- Boşluk kontrolü
        IF current_char = ' ' THEN
            RAISE_APPLICATION_ERROR(-20002, 'String boşluk içeremez.');
        END IF;

        -- Rakam kontrolü
        IF current_char BETWEEN '0' AND '9' THEN
```

Task completed in 0,054 seconds

```
PL/SQL procedure successfully completed.
```



Buffer Size: 20000

```
  <char>d</char>
  <char>e</char>
  <char>f</char>
  <char>g</char>
</root>

String: acbbcadefghkkhgfed
a
c
b
b
c
a
d
e
f
g
h
k
k
h
g
f
e
d
```