



Technical Education & Social Welfare Society

(Run Under CG Society Act.Reg.No.122201950672)

SQL Hindi Note

Head Office: VIP Road Fundhar Raipur CG

Ques - SQL क्या है?

SQL को हम निम्नलिखित बिंदुओं के आधार पर आसानी से समझ सकते हैं:-

- 1-** SQL का पूरा नाम structured query language है।
- 2-** SQL का वास्तविक नाम SEQUEL है जिसे हम Structured English query Language कहते हैं।
- 3-** SQL के द्वारा हम डेटाबेस को access कर सकते हैं।
- 4-** SQL के द्वारा हम डेटाबेस में नए records को insert कर सकते हैं।
- 5-** SQL के द्वारा हम डेटाबेस से डेटा को delete कर सकते हैं।
- 6-** SQL के द्वारा हम डेटाबेस में स्थित data को update कर सकते हैं।
- 7-** यह एक ansi(American national standard Institute) language है।
- 8-** यह Oracle, MS-Access, DB2 तथा MS SQL Server के साथ प्रयोग में लाया जाता है।

SQL Data types क्या है?

SQL में data types निम्नलिखित प्रकार के होते हैं तो आइये हम इन्हें विस्तार से जानते हैं।

Exact numeric types

SQL में निम्नलिखित प्रकार के exact numeric data types होते हैं।

- 1. Int:-** int की सीमा(range) -2,147,483,648 से लेकर +2,147,483,648 तक होती है। इसको 4 bytes की storage की आवश्यकता होती है।
- 2. Bigint:-** Bigint की रेंज -9,223,372,036,854,775,808 से लेकर +9,223,372,036,854,775,808 होती है। इसको 8 bytes की storage की आवश्यकता होती है।
- 3. Smallint:-** इसकी रेंज -32,768 से लेकर +32,768 तक होती है। इसको 2 bytes की storage की आवश्यकता होती है।
- 4. Tinyint:-** इसकी रेंज 0 से लेकर 255 तक होती है। इसको 1 bytes के storage की आवश्यकता होती है।

5. Bit:- bit डेटा टाइप की रेंज 0 से लेकर 1 तक होती है। इसको 1 bytes की storage की आवश्यकता होती है।
6. Money:- इस डेटा टाइप की रेंज -922,337,203,685,477.5808 से लेकर +922,337,203,685,477.5808 तक होती है। इसको 8 bytes की स्टोरेज की आवश्यकता होती है।
7. Smallmoney:- इस data type की range -214,748.3648 से लेकर +214,748.3648 तक होती है। इसको 4 bytes की आवश्यकता होती है।
8. Decimal:- इसकी रेंज $-10^{38}+1$ से लेकर $+10^{38}+1$ तक होती है।
9. Numeric:- इसकी रेंज भी $-10^{38}+1$ से लेकर $+10^{38}+1$ तक होती है।

Datetime type

SQL में निम्नलिखित प्रकार के datetime data types होते हैं।

1. Date:- यह डेटा टाइप साल, महीने, तथा दिन की वैल्यू को स्टोर करता है। इसको 3 bytes की स्टोरेज की आवश्यकता होती है।
2. time:- यह डेटा टाइप घंटा, मिनट, तथा सेकण्ड की वैल्यू को स्टोर करता है।
3. Smalldatetime:- इस डेटा टाइप को 8 bytes की स्टोरेज की आवश्यकता होती है।

Character data type

SQL में निम्नलिखित प्रकार के character डेटा टाइप होते हैं।

1. Char
2. Varchar
3. Varchar(max)
4. Text

Unicode character type

Unicode character निम्नलिखित प्रकार के होते हैं।

1. nchar
2. nvarchar
3. nvarchar(max)
4. ntext

Binary data type

SQL में बाइनरी डेटा टाइप निम्नलिखित प्रकार के होते हैं।

1. Binary
2. Varbinary
3. Varbinary(max)
4. Image

Boolean data type

Boolean data type की सिर्फ दो वैल्यू होती है true या false.

Ques:- SQL में table को कैसे create करते हैं?

SQL में हम TABLE को निम्नलिखित तरीके से create करते हैं।

SYNTAX:-

```
CREATE TABLE table_name  
(  
column_name1 data_type,  
column_name2 data_type,  
_____  
_____  
)
```

अब हम निम्न उदाहरण के द्वारा इसको आसानी से समझ सकते हैं:-

इस उदाहरण में हमने Student के नाम से table बनाई है जिसमें चार columns हैं। column के नाम Last Name, First Name, Address तथा Roll number हैं।

```
CREATE TABLE Student
```

```
(  
Last Name varchar,  
First Name varchar,  
Address varchar,  
Roll number int,
```

```
)
```

यदि हमने column के अधिकतम साइज़ को specify करना हो तो-

```
CREATE TABLE Student
```

```
(
```

Last Name varchar(30),
First Name varchar,
Address varchar,
Roll number int(5),
)

Ques: - SQL में table को drop और alter कैसे करते हैं?

DROP TABLE:- Table को ड्रॉप करने के लिए निम्नलिखित syntax का प्रयोग किया जाता है;

SYNTAX:-

[DROP TABLE table_name]

ALTER TABLE:- Alter table स्टेटमेंट का प्रयोग टेबल में column को add और drop करने के लिए किया जाता है।

SYNTAX:-

[ALTER TABLE table_name
ADD to column_name datatype
ALTER TABLE table_name
DROP COLUMN column_name]

Ques:- SQL में built in functions क्या हैं?

SQL को built-in-functions और भी ज्यादा powerful बनाते हैं। Built-in-functions से हम मुश्किल tasks को भी आसानी से हल कर सकते हैं। इसलिए हम यहां built-in-functions के बारे में चर्चा करेंगे। जो निम्न है:-

1. SUM:- SUM फंक्शन का प्रयोग न्यूमेरिक कॉलम के total sum को select करने के लिए किया जाता है।

इसका syntax निम्न है:-

【SELECT SUM (column) FROM table】

उदहारण के लिए:-

table name: - students

LAST NAME	FIRST NAME	AGE
Joshi	Yugal	21
Bhatt	Kamal	23
Sharma	Piya	20

यह टेबल students नाम से है। इस table की कॉलम के सभी ages को जोड़ दिया जाता है।

【SELECT SUM (age) FROM students】

इस उदहारण का answer= 64

2-AVG:- AVG फंक्शन, कॉलम की औसत वैल्यू को return करता है। इसका syntax निम्न है:-

【SELECT AVG (column) FROM table】

उदहारण:- ऊपर के चित्र की table में:-

【SELECT AVG (age) FROM students】

इस उदहारण का answer=21.33

3-MAX:- MAX फंक्शन कॉलम में सबसे उच्च(highest) वैल्यू को return करता है। इसका syntax

निम्न है:-

【SELECT MAX (column) FROM table】

उदहारण:- ऊपर की टेबल से

【SELECT MAX (age) FROM students】

इस उदहारण का answer= 23

4- MIN:- MIN फंक्शन का प्रयोग column की निम्न(lowest) वैल्यू को return करने के लिए किया जाता है।

इसका syntax निम्न है:-

【SELECT MIN (column) FROM table】

उदहारण:- ऊपर की टेबल से

【SELECT MIN (age) FROM students】

इस उदहारण का answer= 20

5- COUNT:- COUNT फंक्शन का प्रयोग एक डेटाबेस टेबल में rows की संख्या को गिनने(count) के लिये किया जाता है।

इसका syntax निम्न है:-

【SELECT COUNT (column) FROM table】

6-SQRT:- इस फंक्शन का प्रयोग दिए हुए नंबर के वर्ग मूल(square root) को ज्ञात करने के लिए किया जाता है।

7-RAND:- इस फंक्शन का प्रयोग random numbers को generate करने के लिए किया जाता है।

8-Numeric Function:- इन फंक्शनस का प्रयोग सामान्यतया गणितीय manipulation करने के लिए किया जाता है। कुछ numeric functions निम्न है:-
ASIN, ABS, FLOAT, INT, INTEGER, LN, COT, COS, DOUBLE, LOG10

9-String functions:- string functions का प्रयोग string manipulation के लिये किया जाता है। कुछ string functions निम्न है:-
CHAR, CONCAT, GRAPHIC, DIGITS, LCASE, LEFT, LENGTH, LOCATE, LOWER, DIFFERENCE

Ques – SQL में indexes क्या होते हैं?

डेटा को तेजी तथा सुविधापूर्ण तरीके से ढूँढने (find) के लिए index को टेबल में create करते है। INDEX के द्वारा हम data को बिना पूरी table पढ़ें तेजी से ढूँढ सकते है।

डेटाबेस की index उस तरह समान होती है जिस तरह किसी किताब के पीछे index होती है। किताब में हम किसी topic को पढ़ने से पहले index को देखते है कि वह topic किस पेज पर है फिर उस पेज में जाते है।

Indexes को existing table में create करते है जिससे कि हम rows को तेजी से तथा सुविधापूर्ण तरीके से locate कर सकते है। जो users होते है वो indexes को देख नहीं सकते है वे सिर्फ speed up queries का प्रयोग करते है।

Indexes को किसी table की एक या उससे ज्यादा columns में create कर सकते है तथा प्रत्येक index को एक नाम दे सकते है।

Unique index:- यूनिक इंडेक्स उस इंडेक्स को कहते है जिसमें दो rows के पास समान index वैल्यू नहीं होती है।

Unique index को create करने के लिए निम्नलिखित syntax का प्रयोग किया जाता है:-

```
CREATE UNIQUE INDEX index_name  
ON table_name (column_name)
```

column_name उस column को specify करती है जिसे हमने indexed करना हो।

Simple index:- यदि UNIQUE keyword का प्रयोग नहीं किया जाता है तो duplicate वैल्यू को permit किया जा सकता है।

इसका syntax निम्न है:-

```
CREATE INDEX index_name  
ON table_name (column_name)
```

composite index:- दो या उससे अधिक table में जब index को create करते हैं तो वह composite index कहलाता है। इसका syntax निम्न है:-

```
CREATE INDEX index_name
```

```
ON table_name (column1,column2);
```

implicit index:- implicit index वह indexes होती है जो डेटाबेस server द्वारा अपने आप create हो जाती है जब ऑब्जेक्ट(object) create होता है।

DROP INDEX:- Index को drop करने के लिए निम्नलिखित syntax का प्रयोग किया जाता है:-

```
Drop index_name index_name
```

Ques - SQL में view क्या है?

VIEW एक virtual टेबल होती है लेकिन यह किसी वास्तविक टेबल की तरह rows और columns को contain करती है।

जब एक टेबल create होती है और उसमें डेटा को insert किया जाता है। प्रत्येक user को इस डेटा को access करने के लिए permission की आवश्यकता होती है परन्तु हर किसी को इसकी permission नहीं दी जाती है जिससे डेटा की security बनी रहती है तथा data को दुबारा भी नहीं लिखा जाता है जिससे डेटा redundant भी नहीं होता है।

View कोई भी record नहीं रखता यह केवल टेबल की definition रखता है तथा टेबल से डेटा fetch करके show करता है।

CREATE VIEW:- View को create करने के लिए निम्न syntax का प्रयोग किया जाता है:-

```
CREATE VIEW view_name AS
```

```
SELECT columns
```

```
FROM tables
```

```
WHERE conditions;
```

यहां पर हम view को कुछ भी नाम दे सकते हैं।

DROP VIEW:- view को drop करने के लिए निम्न syntax का प्रयोग किया जाता है:-

```
DROP VIEW view_name;
```

UPDATE VIEW:- view को अपडेट करने के लिए निम्न syntax का प्रयोग किया जाता है:-

```
CREATE OR REPLACE VIEW view_name AS
```

```
SELECT columns
```


FROM table

WHERE conditions;

view का प्रयोग data manipulation के लिए भी किया जाता है परन्तु view निम्न condition फॉलो करता हो:-

- 1:-** view सिंगल टेबल से बनाया गया हो।
- 2:-** यदि यूजर record insert करना चाहता हो तो view में primary key तथा सभी not null column भी include किये गए हो।
- 3:-** view को update तथा delete करने के लिए view में से primary key तथा not null column को exclude(निकालना) किया गया हो।

Ques - PL/SQL में procedure क्या होता है?

Procedure एक stored प्रोग्राम होता है जिसमें से हम parameters pass कर सकते हैं। procedures का प्रयोग एक specific task को perform करने के लिए किया जाता है। यह दूसरे प्रोग्रामिंग लैंग्वेज के procedure की तरह ही similar होता है।

PROCEDURE फंक्शन(function) की तरह ही कार्य करता है परन्तु function में return value पहले से निर्धारित होती है जबकि procedure में ऐसा नहीं किया जाता है। function को call करना तथा procedure को call करना अलग-अलग विधियां हैं। Function call करते समय उसकी वैल्यू या तो किसी variable में ली जाती है या output में print करायी जाती है। जबकि PROCEDURE को direct call किया जाता है तथा आवश्यक parameters pass कर दिए जाते हैं। PROCEDURE भी FUNCTION की तरह oracle database में store रहते हैं जिसे आवश्यकता के अनुसार user call कर सकता है।

TYPES OF PARAMETERS IN PL/SQL PROCEDURE:-

- 1.IN :-** IN पैरामीटर procedure को एक वैल्यू pass करने देता है। यह एक read-only पैरामीटर है। Procedure के अंदर, IN पैरामीटर Constant की तरह कार्य करता है।
- 2.OUT:-** एक OUT पैरामीटर calling program को वैल्यू return करता है। Procedure के अंदर, OUT पैरामीटर variable की तरह कार्य करता है।
- 3.IN OUT:-** IN OUT पैरामीटर procedure को एक initial वैल्यू pass करता है और caller को एक updated वैल्यू return करता है।

SQL object क्या होता है?

SQL OBJECT यूजर के द्वारा define किये जाते हैं। SQL द्वारा कई प्रकार के object बनाये जाते हैं जैसे:-table, view, synonym तथा series इत्यादि। Objects को बनाने के लिए कुछ rules होते हैं जो निम्न हैं:-

1:-Object का नाम अधिकतम 30 characters का हो सकता है।

2:-Object का नाम case sensitive नहीं होता है। इसे upper तथा lower case में लिखा जा सकता है। object का नाम alphabet से शुरू होता है।

3:-object के नाम में " " या ' ' का use नहीं किया जाता है।

4:-object के नाम में digit alphabets, \$, # इत्यादि का प्रयोग किया जा सकता है। Database का नाम 8 अक्षरों से अधिक नहीं होना चाहिए।

SQL में cursor क्या है?

SQL स्टेटमेंट को प्रोसेस करने के लिए oracle एक मेमोरी एरिया create करता है, जिसे context एरिया भी कहा जाता है।

Cursor इस मेमोरी एरिया के लिए पॉइंटर होता है और cursor के द्वारा इस मेमोरी(context) एरिया को नियंत्रित किया जाता है।

cursor का size उसके डेटा के अनुसार flexible होता है। Oracle cursor को open करने के लिए main memory में कुछ space predefined रखता है इसलिए cursor की size limited होती है। जब भी कोई query run होती है तो cursor कार्य करता है।

Cursor दो प्रकार की होता है:-

1:-Implicit cursor

2:-Explicit cursor

1:-Implicit cursor:- जब भी SQL स्टेटमेंट execute होता है, Oracle के द्वारा implicit cursor को automatically create कर दिया जाता है(जब स्टेटमेंट के लिए कोई explicit cursor नहीं होता है।)

“प्रोग्रामर्स implicit cursor तथा उसमें उपस्थित सूचना को नियंत्रित नहीं कर सकते हैं।”

Oracle द्वारा internal processing के लिए जो cursor open किया जाता है उसे implicit cursor कहते हैं।

2:- Explicit cursor:- Explicit cursor यूजर-डिफाइंड होता है। Explicit cursor के द्वारा हम मेमोरी(context) एरिया में ज्यादा नियंत्रण कर सकते हैं।

जब किसी टेबल से कुछ रिकॉर्ड को PL/SQL code block में प्रयोग किया जाता है तब cursor का उपयोग किया जाता है। इस cursor को declare करके के लिए SQL queries का प्रयोग किया जाता है।

इन दोनों प्रकार के cursor में चार common attributes होते हैं, जो निम्नलिखित हैं:-

1:-%isopen:- यदि cursor open है तो true अथवा false वैल्यू return करता है।

2:-%found:- यदि record सफलतापूर्वक fetch किये गए हैं तो true अथवा false वैल्यू return करता है।

3:-%notfound:- यदि record successfully fetch नहीं किये गए हो तो true अथवा false वैल्यू return करता है।

4:-%Rowcount:- यह process किये गए record की संख्या return करता है।

Explicit cursor management:- Explicit cursor management करने के लिए निम्न steps होते हैं:-

1:-SQL की select query द्वारा cursor एरिया select करते हैं।

2:-Cursor को open करते हैं।

3:-Cursor variable में Record को fetch करते हैं।

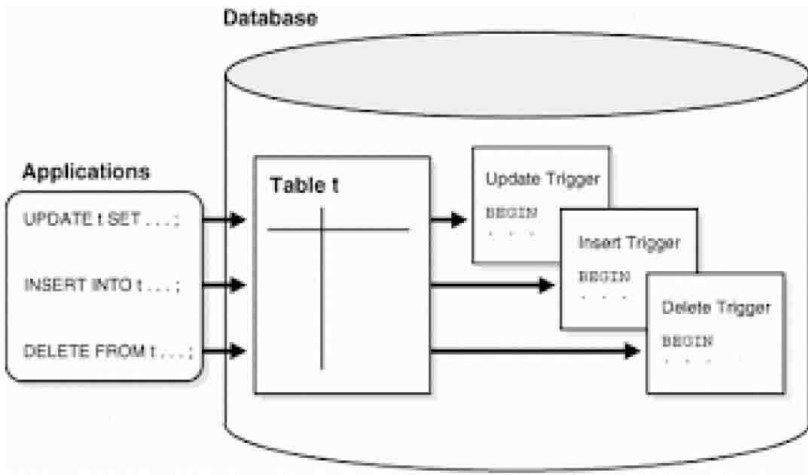
Explicit cursor को create करने के लिए निम्नलिखित syntax है:-

[CURSOR cursor_name IS select_statement]

PL/SQL Trigger क्या होता है?

Trigger संग्रहित(stored) प्रोग्राम्स होते हैं जब कभी table में कोई action जैसे:-insertion, deletion, update किया जाता है तो triggers अपने आप स्वयं execute हो जाते हैं।

Triggers को call या invoke नहीं किया जा सकता क्योंकि जब भी किसी DML स्टेटमेंट को perform किया जाता है तो trigger अपने आप execute हो जाता है।



syntax of creating trigger:-

```
CREATE [OR REPLACE ] TRIGGER trigger_name
{BEFORE | AFTER | INSTEAD OF }
{INSERT [OR] | UPDATE [OR] | DELETE}
[OF col_name]
ON table_name
[REFERENCING OLD AS o NEW AS n]
[FOR EACH ROW]
WHEN (condition)
BEGIN
--- sql statements
END;
```

Types of triggers:- Trigger दो प्रकार के होते हैं:-

1:-Row level trigger:- जब प्रत्येक row में update, deletion, तथा insertion होता है तो तब Row level trigger घटित होता है।

2:-Statement level trigger:- जब प्रत्येक SQL स्टेटमेंट execute होता है तब statement level trigger घटित होता है।

SQL में if और while statement क्या है?

while statement:-

यदि कंडीशन true है तो स्टेटमेंट execute होता है, परन्तु कंडीशन को दुबारा evaluate किया जाता है यह देखने के लिए कि कंडीशन true है या false. और यह प्रक्रिया तब तक चलती रहती है जब तक कि कंडीशन false नहीं हो जाती है।

syntax:-

```
WHILE bool_exp  
{sql_statement}  
[BREAK]  
{sql_statement}  
[CONTINUE]
```

If statement:-

यदि कंडीशन true है तो if स्टेटमेंट run होगा।
नहीं तो if स्टेटमेंट का कुछ काम नहीं
है।

syntax:-

```
IF condition THEN  
statements  
END IF;
```

PL/SQL में Exception handling क्या है?

PL/SQL में, Exception एक error होता है जो program के सामान्य flow को रोक देता है। यह program के execution के दौरान घटित होने वाला एक error होता है।

दूसरे शब्दों में कहें तो, "exception एक error होता है जो कि run-time में घटित होता है जिसके कारण program का execution नहीं हो पाता।"

इन errors को हम compile-time में पकड़ नहीं पाते और इसलिए हमें इन्हें run-time में handle करने की जरूरत होती है।

PL/SQL में exception block के द्वारा हम program में error को find और resolve कर सकते हैं।

Exception दो प्रकार के होते हैं:-

- User-Defined Exception
- System-Defined Exception

Exception Handling

System-defined Exception

User-defined Exception

Exception Handling का syntax – इसका syntax नीचे दिया गया है. आप जितने चाहो उतने exceptions को handle कर सकते हो. Default exception को WHEN और THEN के द्वारा handle किया जाता है.

```
DECLARE
<declarations section>
BEGIN
<executable command(s)>
EXCEPTION
<exception handling goes here >
WHEN exception1 THEN
exception1-handling-statements
WHEN exception2 THEN
exception2-handling-statements
WHEN exception3 THEN
exception3-handling-statements
.....
WHEN others THEN
exception3-handling-statements
END;
```

यहाँ ध्यान देने वाली बात यह है कि WHEN others THEN का प्रयोग केवल exception handling block के अंत में किया जाता है. इसके बाद किसी भी exception को handle नहीं किया जाएगा.

System Defined Exceptions in Hindi

ये एक्सेप्शन PL/SQL में पहले से ही define होते हैं और इन्हें तब execute किया जाता है जब database के किसी विशेष rule को violate (उल्लंघन) किया जाता है. इन्हें pre-defined exceptions भी कहा जाता है.

system defined exceptions के भी दो प्रकार होते हैं:-

1. Named system exceptions.
2. Unnamed system exceptions

Named system exceptions – सिस्टम के द्वारा इनका नाम पहले से ही define होता है जैसे कि – ACCESS_INTO_NULL, DUP_VAL_ON_INDEX, LOGIN_DENIED आदि.

Unnamed system exceptions – वे exceptions जिन्हें system के द्वारा कोई भी नाम नहीं दिया जाता उन्हें unnamed system exceptions कहते हैं. ये exceptions सामान्यतया घटित नहीं होते हैं. इनके दो parts होते हैं पहला code और दूसरा message. इस प्रकार के एक्सेप्शन को handle करने के लिए इनको Pragma EXCEPTION_INIT का प्रयोग करके नाम assign किया जाता है.

इसका **syntax** –

```
PRAGMA EXCEPTION_INIT(exception_name, -error_number);
```

User-defined Exceptions in Hindi

इस प्रकार के exceptions को user के द्वारा खुद create किया जाता है. यूजर अपनी आवश्यकता के अनुसार exceptions को create कर सकते हैं और उन्हें handle कर सकते हैं. User-define exception को सबसे पहले declare करना आवश्यक होता है और उसके बाद उन्हें raise किया जाता है. इन्हें RAISE स्टेटमेंट और DBMS_STANDARD.RAISE_APPLICATION_ERROR की प्रक्रिया के द्वारा raise किया जाता है.

इसको declare करने का syntax निम्नलिखित हैं:-

```
DECLARE  
my-exception EXCEPTION;
```

- [SQL क्या है?](#)
- [SQL Data Type in Hindi](#)

Raising Exceptios – एक्सेप्शन को raise करना

जब भी कभी आंतरिक database error होता है तो database server के द्वारा exceptions को automatically (अपने-आप) raise कर दिया जाता है परन्तु exceptions को RAISE command के द्वारा भी raise किया जा सकता है.

एक exception को raise करने का syntax निम्नलिखित हैं:-

```
DECLARE  
exception_name EXCEPTION;  
BEGIN  
IF condition THEN
```

```

RAISE exception_name;
END IF;
EXCEPTION
WHEN exception_name THEN
statement;
END;

```

Exception Handling का example

नीचे हमने STUDENT [table को create](#) किया है.

```
SELECT* FROM STUDENT;
```

ID	NAME	AGE	ADDRESS	ROLL No
1	Yugal	26	Delhi	111111
2	Kamal	27	Dehradun	222222
3	Pankaj	22	Nainital	333333
4	Vishal	23	Kanpur	444444
5	Deepak	24	Mumbai	555555
6	Vinay	25	Haridwar	666666

इस Table का program –

```

DECLARE
s_id student.id%type := 9;
s_name student.name%type;
s_addr student.address%type;
BEGIN
SELECT name, address INTO s_name, s_addr
FROM student
WHERE id = s_id;
DBMS_OUTPUT.PUT_LINE ('Name: ' || s_name);
DBMS_OUTPUT.PUT_LINE ('Address: ' || s_addr);
EXCEPTION
WHEN no_data_found THEN
dbms_output.put_line('No such student!');
WHEN others THEN
dbms_output.put_line('Error!');
END;

```


इसका आउटपुट –

No such student

PL/SQL procedure successfully completed.

ऊपर दिए गये program में student का नाम और address आउटपुट में आना चाहिए. परन्तु हमने यहाँ id 9 लिखा है और table में id 9 नहीं है इसलिए प्रोग्राम run-time exception – NO_DATA_FOUND raise करेगा.