

Date: 05/02/2023  
Course Code: 5CS322  
Course Name: ADS  
PRN: 21520012 Batch: T1

## Assignment No. 2

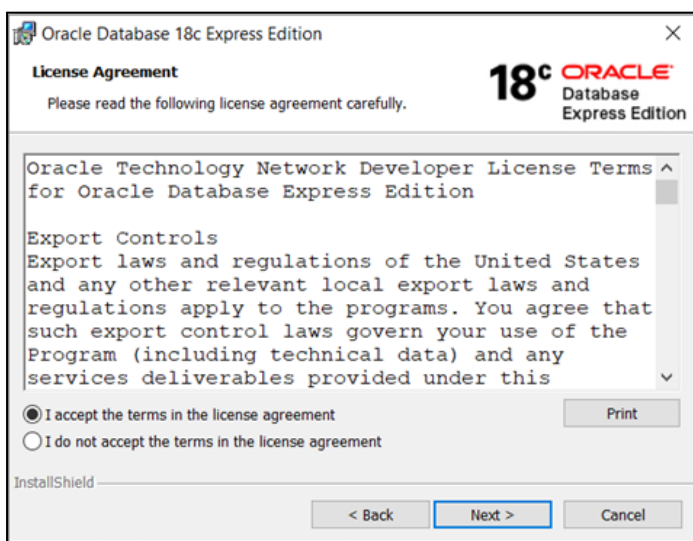
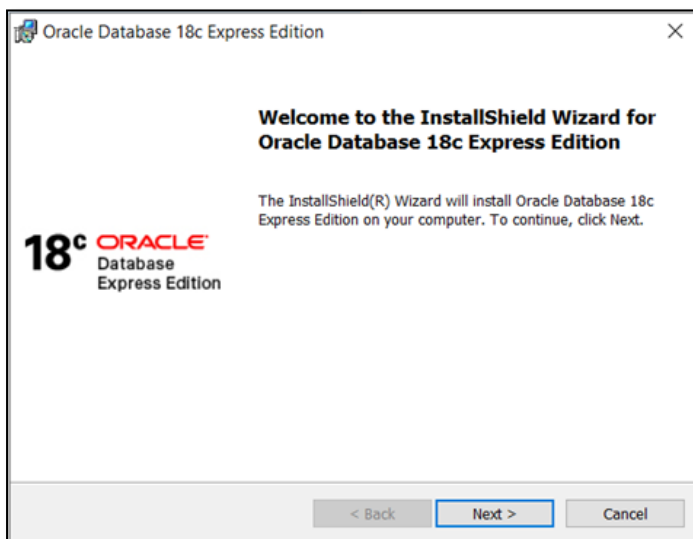
### Title:

Installation, configuration & testing of Oracle 12c EE / 18c XE

### Objective / Aim:

To Study CRUD operations from remote machine configuration

### Installation and configuration steps:



Oracle Database 18c Express Edition

**Oracle Database Information**  
Specify the database password.

18<sup>c</sup> ORACLE  
Database  
Express Edition

This password will be used for SYS, SYSTEM and PDBADMIN accounts.

Enter Database Password

Confirm Database Password

InstallShield

< Back Next > Cancel

Oracle Database 18c Express Edition

**Oracle Database Installed Successfully.**

The InstallShield Wizard has successfully installed Oracle Database 18c Express Edition. Click Finish to exit the wizard.

Oracle Database Express Edition Connection Information:

Multitenant container database: localhost:1521

Pluggable database: localhost:1521/XEPDB1

EM Express URL: <https://localhost:5500/em>

18<sup>c</sup> ORACLE  
Database  
Express Edition

< Back Finish Cancel

## Specific settings:

Once you download Oracle 18C XE you have to execute some queries before processing further. Below are some screenshots of some specific queries you need to make connections.

Wireless LAN adapter Wi-Fi:

```
Connection-specific DNS Suffix  . :  
Link-local IPv6 Address . . . . . : fe80::e2e7:7de8:d868:9965%21  
IPv4 Address. . . . . : 10.40.8.215  
Subnet Mask . . . . . : 255.255.224.0  
Default Gateway . . . . . : 10.40.0.2
```

```
D:\B.Tech\WCE\6th Sem\ADS\Assignments\Parshwa\assignment02>sqlplus system/123@10.40.8.215:1521/XEPDB1
```

```
SQL*Plus: Release 18.0.0.0.0 - Production on Sun Feb 5 14:57:46 2023  
Version 18.4.0.0.0
```

```
Copyright (c) 1982, 2018, Oracle. All rights reserved.
```

```
Last Successful login time: Sun Feb 05 2023 14:16:14 +05:30
```

```
Connected to:  
Oracle Database 18c Express Edition Release 18.0.0.0.0 - Production  
Version 18.4.0.0.0
```

```
SQL> show user;  
USER is "SYSTEM"  
SQL> alter user hr account unlock;
```

```
User altered.
```

```
SQL> alter user hr identified by hr;
```

```
User altered.
```

```
SQL> exit  
Disconnected from Oracle Database 18c Express Edition Release 18.0.0.0.0 - Production  
Version 18.4.0.0.0
```

```
D:\B.Tech\WCE\6th Sem\ADS\Assignments\Parshwa\assignment02>sqlplus hr/hr@10.40.8.215:1521/XEPDB1
```

```
SQL*Plus: Release 18.0.0.0.0 - Production on Sun Feb 5 14:59:42 2023  
Version 18.4.0.0.0
```

```
Copyright (c) 1982, 2018, Oracle. All rights reserved.
```

```
Connected to:  
Oracle Database 18c Express Edition Release 18.0.0.0.0 - Production  
Version 18.4.0.0.0
```

```
SQL> show user;  
USER is "HR"
```

```
SQL> select tname from tab;
```

```
TNAME
```

```
-----  
REGIONS  
COUNTRIES  
LOCATIONS  
DEPARTMENTS  
JOBS  
EMPLOYEES  
JOB_HISTORY  
EMP_DETAILS_VIEW
```

```
8 rows selected.
```

## SQL statements for creating schema, tables:

```
D:\B.Tech\WCE\6th Sem\ADS\Assignments\Parshwa\assignment02>sqlplus system/123@10.40.8.215:1521/XEPDB1

SQL*Plus: Release 18.0.0.0.0 - Production on Sun Feb 5 15:16:30 2023
Version 18.4.0.0.0

Copyright (c) 1982, 2018, Oracle. All rights reserved.

Last Successful login time: Sun Feb 05 2023 15:08:12 +05:30

Connected to:
Oracle Database 18c Express Edition Release 18.0.0.0.0 - Production
Version 18.4.0.0.0
```

```
SQL> create user P_21520008 identified by 21520008;

User created.
```

```
SQL> create user p_21520008 identified by 21520008;

User created.

SQL> grant all privileges to p_21520008;

Grant succeeded.

SQL> exit
Disconnected from Oracle Database 18c Express Edition Release 18.0.0.0.0 - Production
Version 18.4.0.0.0

D:\B.Tech\WCE\6th Sem\ADS\Assignments\Parshwa\assignment02>sqlplus p_21520008/21520008@10.40.8.215:1521/xepdb1

SQL*Plus: Release 18.0.0.0.0 - Production on Sun Feb 5 15:54:26 2023
Version 18.4.0.0.0

Copyright (c) 1982, 2018, Oracle. All rights reserved.

Connected to:
Oracle Database 18c Express Edition Release 18.0.0.0.0 - Production
Version 18.4.0.0.0

SQL> create table product(id int,name varchar(20),price int);

Table created.

SQL> insert into product values(101,'Bag',1000);

1 row created.

SQL> insert into product values(102,'Bottle',500);

1 row created.

SQL> commit;

Commit complete.

SQL> insert into product values(103,'Books',720);

1 row created.

SQL> commit;

Commit complete.
```

## Python (.py) source files:

```
from tkinter import *
import tkinter as tk
from tkinter import messagebox
import os
import sys
import cx_Oracle

base = tk.Tk()
base.geometry("550x450")
base.title('CRUD Operations')
# base['background']='#826ff4'

# Connect to the Oracle database
conn = cx_Oracle.connect("p_21520008/21520008@192.168.16.159:1521/XEPDB1")
cursor = conn.cursor()

# Function to create a record
def create_record():
    id=enter_1.get()
    name = enter_2.get()
    price = enter_3.get()
    cursor.execute("INSERT INTO product (id, name, price) VALUES (:id, :name, :price)", (id, name, price))
    conn.commit()
    messagebox.showinfo("Information", "Record created successfully.")

# Function to read a record
def read_record():
    id=enter_1.get()
    cursor.execute("SELECT * FROM product WHERE id=:id", (id,))
    result = cursor.fetchone()
    if result:
        id=result[0]
        nm=result[1]
        pz=result[2]
        clear()
        enter_4.insert(0,id)
        enter_5.insert(0,nm)
        enter_6.insert(0,pz)
        messagebox.showinfo("Information", "Id:{0}, Name:{1}, Price:{2}".format(id,nm,pz))
    else:
        messagebox.showinfo("Information", "Record not found.")

# Function to update a record
def update_record():
    id=enter_1.get()
```

```

        name = enter_2.get()
        price = enter_3.get()
        cursor.execute("UPDATE product SET name=:name, price=:price WHERE id=:id",
(name, price, id))
        conn.commit()
        messagebox.showinfo("Information","Record updated successfully.")

# Function to delete a record
def delete_record():
    id=enter_1.get()
    cursor.execute("DELETE FROM product WHERE id=:id", (id,))
    conn.commit()
    messagebox.showinfo("Information","Record deleted successfully.")

#clear the string
def clear():
    enter_4.delete(0,END)
    enter_5.delete(0,END)
    enter_6.delete(0,END)

# Create the labels and entry fields
lbl_0 = Label(base, text="Product Management", width=20,font=("bold",20))
lbl_0.place(x=90,y=60)

lbl_1 =Label(base, text= "Product Id", width=20,font=("bold",10))
lbl_1.place(x=80,y=130)
enter_1 = Entry(base)
enter_1.place(x=240,y=130)

lbl_2 = Label(base, text="Product Name", width=20,font=("bold",10))
lbl_2.place(x=80,y=180)
enter_2 = Entry(base)
enter_2.place(x=240,y=180)

lbl_3 =Label(base, text= "Product Price", width=20,font=("bold",10))
lbl_3.place(x=80,y=230)
enter_3 = Entry(base)
enter_3.place(x=240,y=230)

enter_4 = Entry(base)
enter_4.place(x=70,y=280)

enter_5 = Entry(base)
enter_5.place(x=190,y=280)

enter_6 = Entry(base)
enter_6.place(x=310,y=280)

```

```

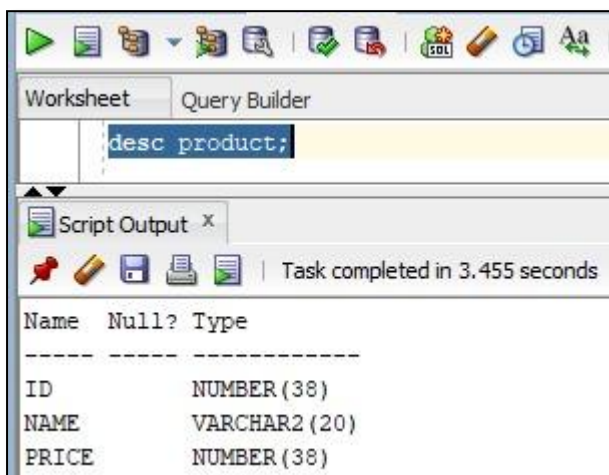
Button(base, text='Read' , width=10,
bg="black",fg='white',command=read_record).place(x=50,y=330)
Button(base, text='Insert' , width=10,
bg="black",fg='white',command=create_record).place(x=150,y=330)
Button(base, text='Update' , width=10,
bg="black",fg='white',command=update_record).place(x=250,y=330)
Button(base, text='Delete' , width=10,
bg="black",fg='white',command=delete_record).place(x=350,y=330)
Button(base, text='Exit' , width=10, bg="black",fg='white',
command=base.destroy).place(x=450,y=330)

base.mainloop()

```

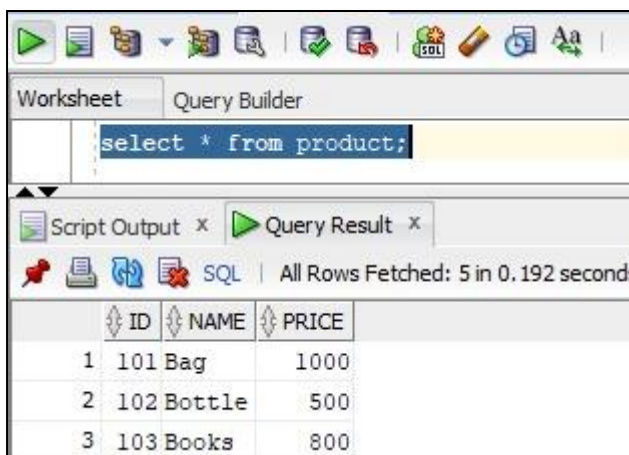
**Screen shot of program execution / output / results:**

**Desc schema:**



Name	Null?	Type
ID		NUMBER (38)
NAME		VARCHAR2 (20)
PRICE		NUMBER (38)

**Read Operation:**



	ID	NAME	PRICE
1	101	Bag	1000
2	102	Bottle	500
3	103	Books	800

## Insert Operation:

CRUD Operations

### Product Management

Product Id:

Product Name:

Product Price:

Information

Record created successfully.

OK

Worksheet | Query Builder

select \* from product;

Script Output x | Query Result x

SQL | All Rows Fetched: 5 in 0.192 seconds

	ID	NAME	PRICE
1	101	Bag	1000
2	102	Bottle	500
3	103	Books	800
4	104	Gun	20000



## Update Operation:

CRUD Operations

### Product Management

Product Id: 104

Product Name: Ball

Product Price: 1500

Read Insert Update Delete Exit

Information

Record updated successfully.

OK

CRUD Operations

### Product Management

Product Id: 104

Product Name: Ball

Product Price: 1500

104 Ball 1500

Read Insert Update Delete Exit

Information

Id:104, Name:Ball, Price:1500

OK

## Delete Operation:

CRUD Operations

### Product Management

Product Id

Product Name

Product Price

Information

Record deleted successfully.

OK

Worksheet Query Builder

```
select * from product;
```

Script Output x Query Result x

SQL | All Rows Fetched: 5 in 0.192 second

	ID	NAME	PRICE
1	101	Bag	1000
2	102	Bottle	500
3	103	Books	800