

## CTIS259 Database Management Systems and Applications

### Lab Guide 18

**Instructor** : Nimet Ceren SERİM  
**Assistant** : Engin Zafer KIRAÇBEDEL, Hatice Zehra YILMAZ

**Week:** 12  
**Date:** 01-02.12.2025

**Aim of this lab session:** 1. PL/SQL Practice

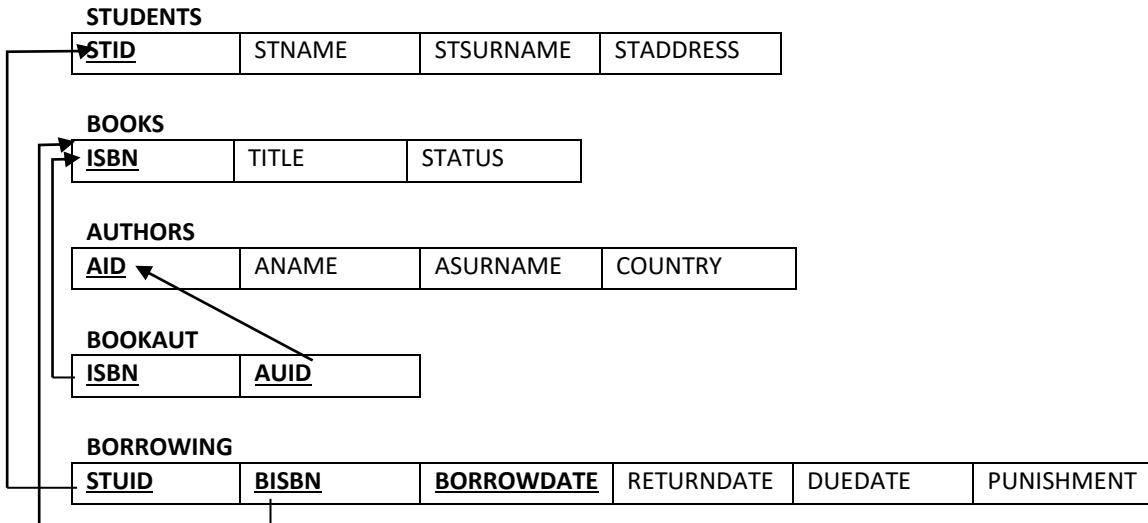
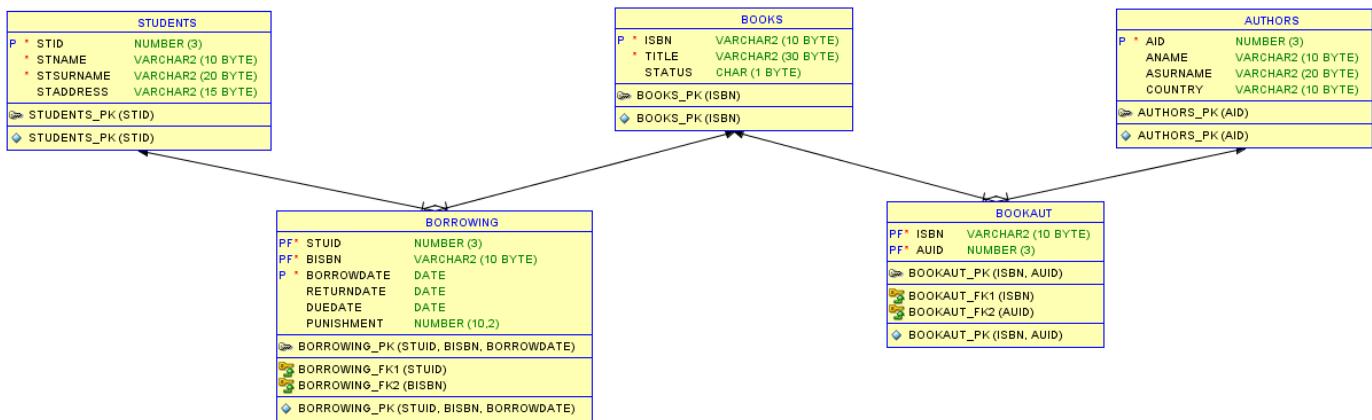
#### ORACLE Server Configurations:

**IP Address:** 139.179.33.231

**Port number:** 1522

**SID:** orclctis

### Please USE stu (fYourStudentID) accounts



# Creating tables and data insertions

Please download **library.zip** file from moodle: **Labguides/LG18**. The zip file contains **database creation and data insertion scripts for Library database**.

Create the below tables in your account and insert data into the tables using [the given scripts](#) file.

- Create a table as **STUDENTS** shown below.

FIELD NAME	FIELD TYPE	SIZE	CONSTRAINTS
<b>STID</b>	NUMBER	PRECISION: 3	PRIMARY KEY
STNAME	VARCHAR2	10	NOT NULL
STSURNAME	VARCHAR2	20	NOT NULL
STADDRESS	VARCHAR2	15	

- Create a table as **BOOKS** shown below.

FIELD NAME	FIELD TYPE	SIZE	CONSTRAINTS
<b>ISBN</b>	VARCHAR2	10	PRIMARY KEY
TITLE	VARCHAR2	30	NOT NULL
STATUS	CHAR	1	A or N

- Create a table as **AUTHORS** shown below.

FIELD NAME	FIELD TYPE	SIZE	CONSTRAINTS
<b>AID</b>	NUMBER	PRECISION: 3	PRIMARY KEY
ANAME	VARCHAR2	10	
ASURNAME	VARCHAR2	20	
COUNTRY	VARCHAR2	10	

- Create a table as **BOOKAUT** shown below. This table has **composite primary key**.

FIELD NAME	FIELD TYPE	SIZE	CONSTRAINTS
<b>ISBN</b>	VARCHAR2	10	PRIMARY KEY FOREIGN KEY
<b>AUID</b>	NUMBER	PRECISION: 3	PRIMARY KEY FOREIGN KEY

- Create a table as **BORROWING** shown below. This table has **composite primary key**.

FIELD NAME	FIELD TYPE	SIZE	CONSTRAINTS
<b>STUID</b>	NUMBER	PRECISION: 3	PRIMARY KEY FOREIGN KEY
<b>BISBN</b>	VARCHAR2	10	PRIMARY KEY FOREIGN KEY
<b>BORROWDATE</b>	DATE		PRIMARY KEY DEFAULT SYSTEM DATE
RETURNDATE	DATE		
DUEDATE	DATE		
PUNISHMENT	NUMBER	PRECISION: 10,SCALE: 2	DEFAULT 0

## A. Data Insertion

- The records as given below for the table of **STUDENTS**.

STID	STNAME	STSURNAME	STADDRESS
111	ALI	OKUR	ANKARA
222	SEVINC	ER	IZMIR
333	MESUT	ERSOY	ANKARA
444	VELI	SURER	ISTANBUL
555	AYSE	GEZER	ISTANBUL

- The records as given below for the table of **BOOKS**.

ISBN	TITLE	STATUS
0-201-20-8	Problem Solving in C	A
0-501-70-6	Programming in Java	N
4-258-60-7	DBMS	A
0-201-00-6	Visual Basic	A
5-401-80-5	ASP	A
6-201-50-9	Algorithm	A
4-580-20-5	Object Oriented Programming	N

- The records as given below for the table of **AUTHORS**.

AID	ANAME	ASURNAME	COUNTRY
121	JERI R.	HANLY	USA
122	ELLIOOT B.	KOFFMAN	USA
211	RICK	DECKER	GERMANY
212	STUART	HIRSHFIELD	GERMANY
311	GARY J.	BRONSON	USA
411	ANDREW	DUTHIE	USA
512	ROBERT	SEDGEWICK	ENGLAND

- The records as given below for the table of **BOOKAUT**.

ISBN	AUID
0-201-20-8	121
0-201-20-8	122
0-501-70-6	211
0-501-70-6	212
4-258-60-7	512
0-201-00-6	512
5-401-80-5	411
6-201-50-9	311
4-580-20-5	311
4-580-20-5	121

- The records as given below for the table of **BORROWING**.

STUID	BISBN	BORROWDATE	RETURNDATE	DUEDATE	PUNISHMENT
111	0-501-70-6	07-OCT-25	(null)	22-OCT-25	0
111	5-401-80-5	04-OCT-25	08-OCT-25	19-OCT-25	0
111	5-401-80-5	08-NOV-25	28-NOV-25	23-NOV-25	0
222	5-401-80-5	29-OCT-25	28-NOV-25	13-NOV-25	0
333	6-201-50-9	27-SEP-25	14-OCT-25	12-OCT-25	0
444	4-580-20-5	14-SEP-25	05-OCT-25	29-SEP-25	0
444	6-201-50-9	05-NOV-25	28-NOV-25	20-NOV-25	0
333	4-580-20-5	09-NOV-25	(null)	24-NOV-25	0

- Do NOT forget to COMMIT changes!!!

## Creating VIEW

Create a view which displays the borrowing information (id, stname, stsurname and the total number of borrowings) for each student.

STUID	STNAME	STSURNAME	number of borrowings
111	ALI	OKUR	3
222	SEVINC	ER	1
333	MESUT	ERSOY	2
444	VELI	SURER	2

## Creating Procedures & Functions

**Q1.** Write a procedure which calculates and updates the punishment payments for all students. If the student returned the book back later than the due date there will be a punishment payment. Daily punishment payment is 2.25 TL.

The procedure displays the **number of records** that have been updated. Call the procedure using a program block.

Script Output X  
Task completed in 1.171 seconds

```
Procedure CALCULATEPUNISHMENT compiled
5 records have been updated!
punishment calculation done :)

PL/SQL procedure successfully completed.
```

## Updated version of the table

(You may check the content of the borrowing table from sqldeveloper)

STUID	BISBN	BORROWDATE	RETURNDATE	DUEDATE	PUNISHMENT
111	0-501-70-6	07-OCT-25	(null)	22-OCT-25	0
111	5-401-80-5	04-OCT-25	08-OCT-25	19-OCT-25	0
111	5-401-80-5	08-NOV-25	28-NOV-25	23-NOV-25	11.25
222	5-401-80-5	29-OCT-25	28-NOV-25	13-NOV-25	33.75
333	6-201-50-9	27-SEP-25	14-OCT-25	12-OCT-25	4.5
444	4-580-20-5	14-SEP-25	05-OCT-25	29-SEP-25	13.5
444	6-201-50-9	05-NOV-25	28-NOV-25	20-NOV-25	18
333	4-580-20-5	09-NOV-25	(null)	24-NOV-25	0

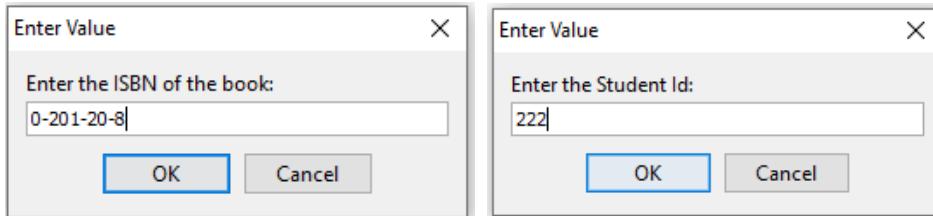
**Q2.** Write a procedure that displays the available book list.

```
Procedure DISPLAYBOOKS compiled

List of Available Books
0-201-20-8 Problem Solving in C A
4-258-60-7 DBMS A
0-201-00-6 Visual Basic A
5-401-80-5 ASP A
6-201-50-9 Algorithm A
```

**Q3.** Write a procedure that makes a reservation for a book. The procedure takes the book isbn and the student id as input parameters and makes the reservation if the book is available, otherwise displays a warning message like “SORRY! The book is NOT available”.

- Each borrowed book should be returned in 15 days. So, add the 15 days to system date and use it as due date.
- Insert a new record into the borrowing table.
- The borrowed book’s status should also be updated to ‘N’ in the Books table.
- Display the appropriate messages for the user inputs using “accept command”.



```
The Book 0-201-20-8 borrowed
```

### Updated version of the tables

(Borrowdate: 01-DEC-25, You may check the content of the books and borrowing tables from sqldeveloper)

STUID	BISBN	BORROWDATE	RETURNDATE	DUEDATE	PUNISHMENT
111	0-501-70-6	07-OCT-25	(null)	22-OCT-25	0
111	5-401-80-5	04-OCT-25	08-OCT-25	19-OCT-25	0
111	5-401-80-5	08-NOV-25	28-NOV-25	23-NOV-25	11.25
222	5-401-80-5	29-OCT-25	28-NOV-25	13-NOV-25	33.75
333	6-201-50-9	27-SEP-25	14-OCT-25	12-OCT-25	4.5
444	4-580-20-5	14-SEP-25	05-OCT-25	29-SEP-25	13.5
444	6-201-50-9	05-NOV-25	28-NOV-25	20-NOV-25	18
333	4-580-20-5	09-NOV-25	(null)	24-NOV-25	0
222	0-201-20-8	01-DEC-25	(null)	16-DEC-25	0

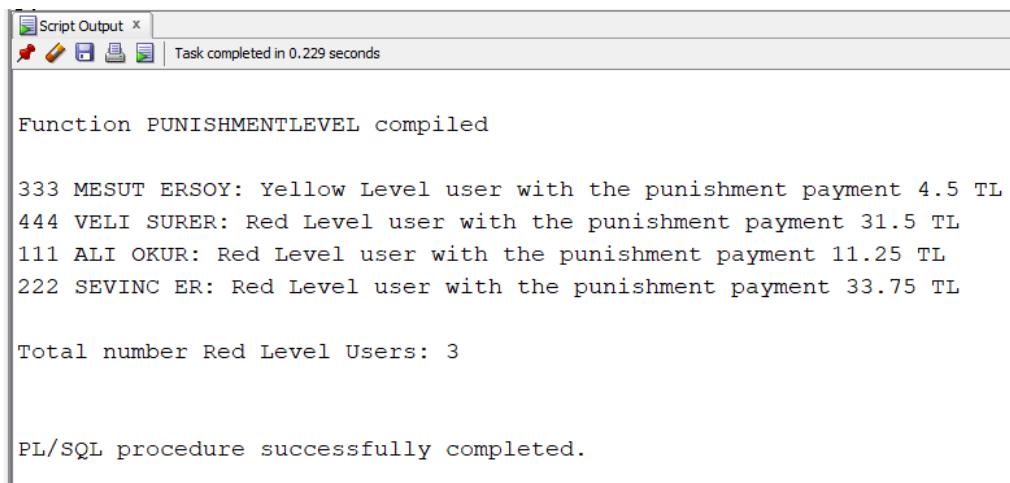
ISBN	TITLE	STATUS
0-201-20-8	Problem Solving in C	N
0-501-70-6	Programming in Java	N
4-258-60-7	DBMS	A
0-201-00-6	Visual Basic	A
5-401-80-5	ASP	A
6-201-50-9	Algorithm	A
4-580-20-5	Object Oriented Programming	N

**Q4.** Write a function which finds the punishment level of each student according to their total punishment payments. The function displays the id, name, surname, punishment level and the total punishment payment for all students.

The punishment level criteria is as follows;

Punishment Level	Total punishment payment (TL)
Red	>10
Yellow	1 - 10
Green	0

The function counts and returns the number of red level users.



Script Output X | Task completed in 0.229 seconds

```
Function PUNISHMENTLEVEL compiled

333 MESUT ERSOY: Yellow Level user with the punishment payment 4.5 TL
444 VELI SURER: Red Level user with the punishment payment 31.5 TL
111 ALI OKUR: Red Level user with the punishment payment 11.25 TL
222 SEVINC ER: Red Level user with the punishment payment 33.75 TL

Total number Red Level Users: 3

PL/SQL procedure successfully completed.
```