SDU - Advanced Database Management Systems

May 5, 2021

JujutsuKaisen Probability Predicting

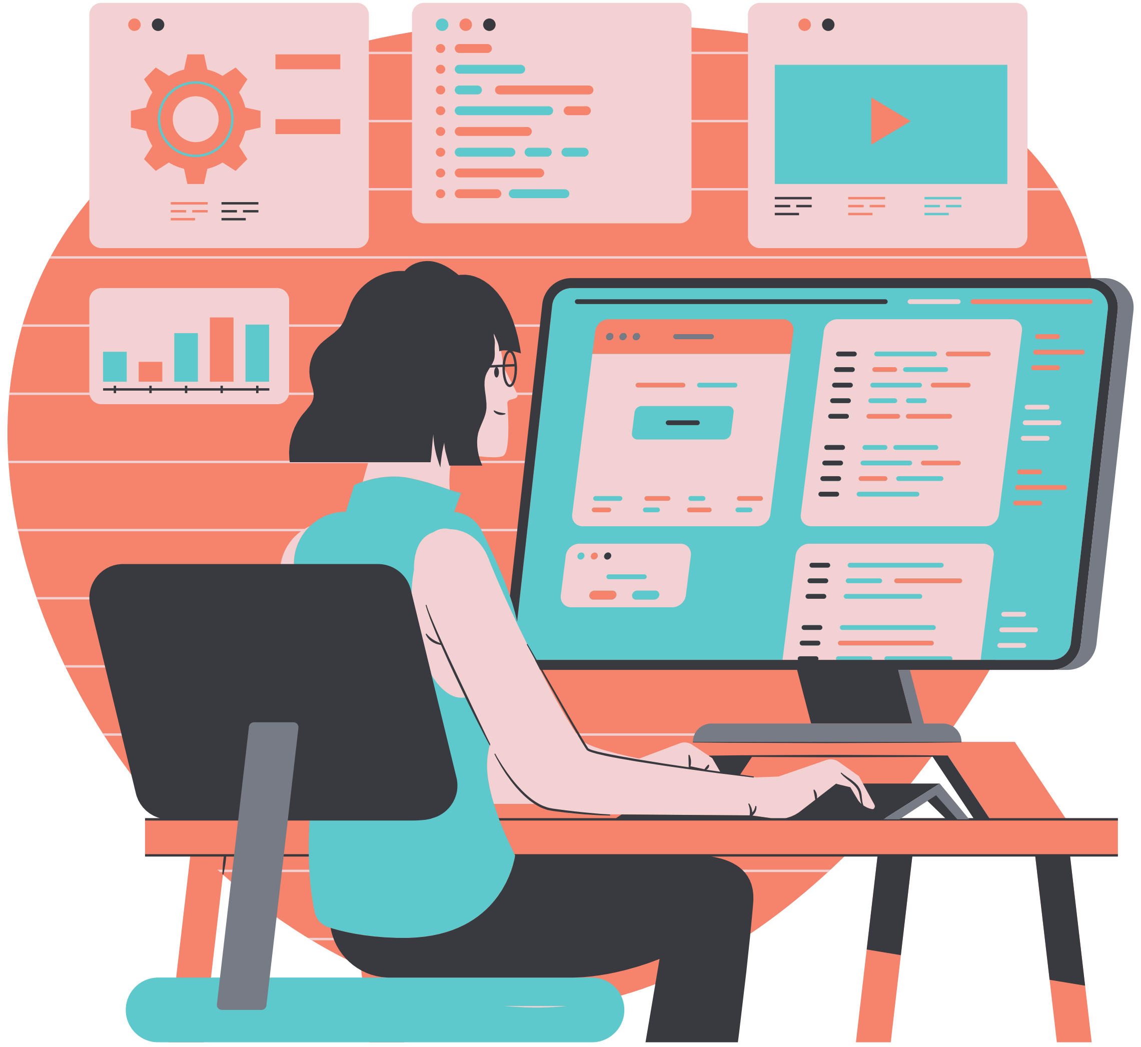
Ayazhan Allayarova
Dina Mugallimova
Akniyet Kadyrova

Today's Agenda

1. Introduction
2. Data for project
3. PyQt5, Python
4. SQL, PL/SQL

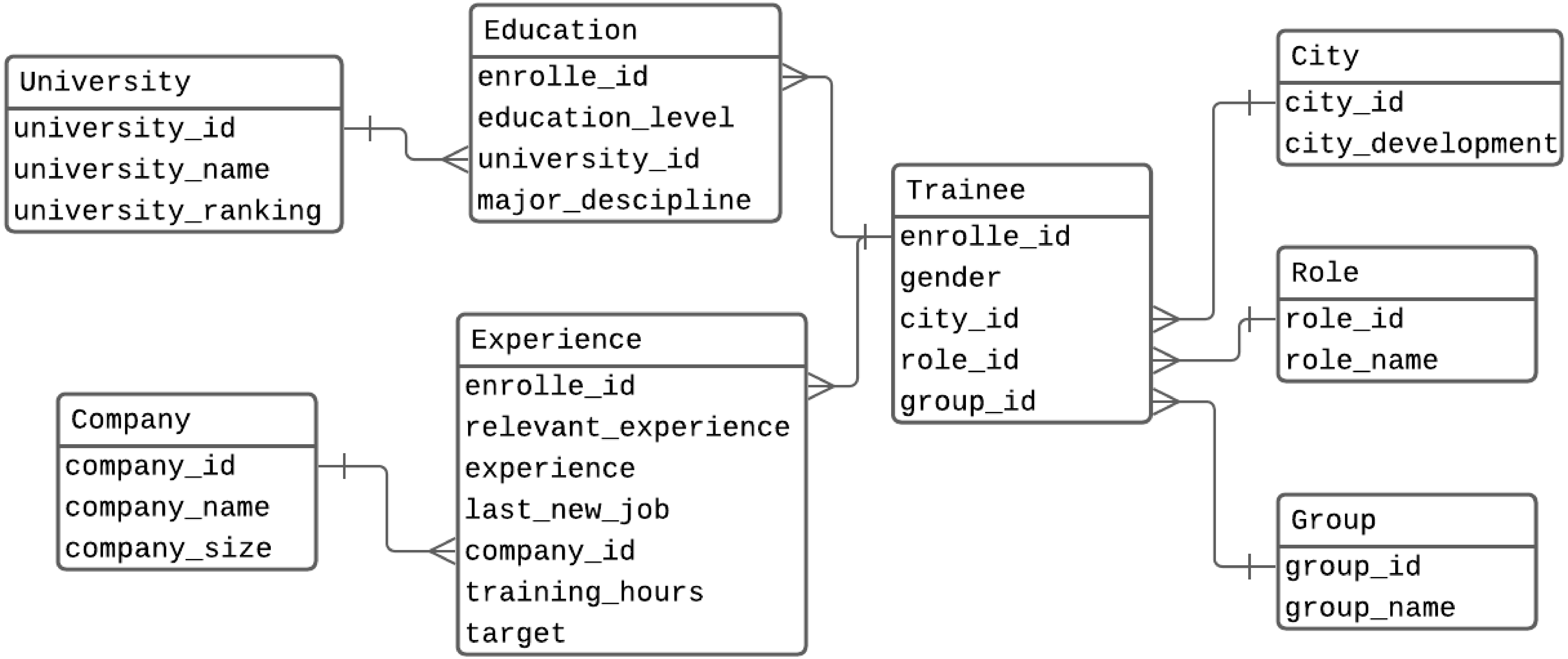
Purpose

The project allows you to identify the best trainees for the course, who stand out not only for experience but also for many hours of practice. Also shows the best universities and cities, companies where good interns come from.



INTRODUCTION

ER-Diagram



University
university_id
university_name
university_ranking

Education
enrolle_id
education_level
university_id
major_descipline

City
city_id
city_development

Trainee
enrolle_id
gender
city_id
role_id
group_id

Role
role_id
role_name

Experience
enrolle_id
relevant_experience
experience
last_new_job
company_id
training_hours
target

Company
company_id
company_name
company_size

Group
group_id
group_name

Data

Trainee

19159 rows

Basic information about the trainees of the company, such as unique ID, gender, city_id, role_id, group_id.

Experience

Information about the trainee's experience, showing last_job, relevant_experience, experience, company_id, as well as general training hours and goal.

19159 rows

University

This object contains the name and ranking of the university.

1001 rows

City

124 rows

Each city has an identifier and shows its development index

Data

Education

19159 rows

Links to trainees and universities, and shows what education the trainees graduated from or whether they graduated at all.

Role

1001 rows

This object contains job titles that identify the trainees' specialization, such as designer or accountant.

Group

105 rows

The group contains a unique identifier and name.

Company

Each company has its own name and ID, as well as the size of the companies.

3378 rows

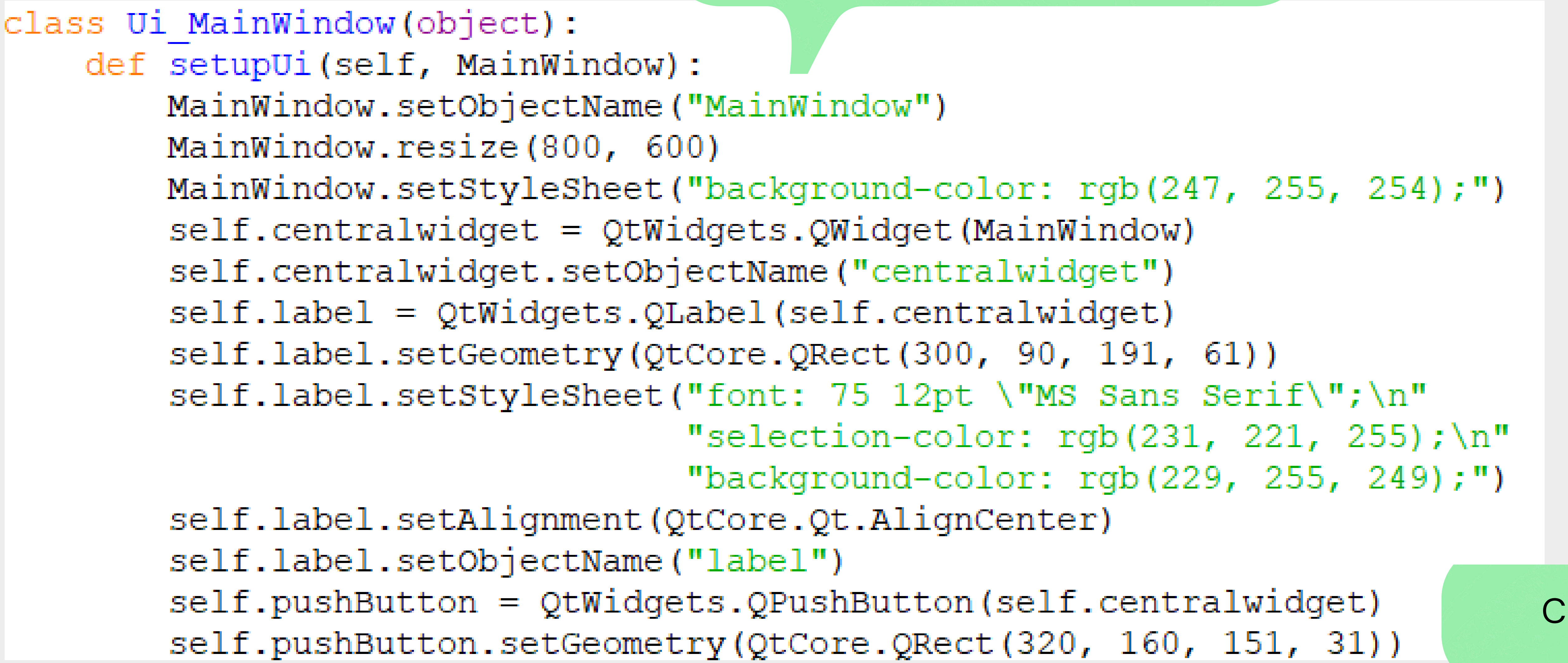
PyQt5

LABEL

PUSHBUTTON

centralwidget

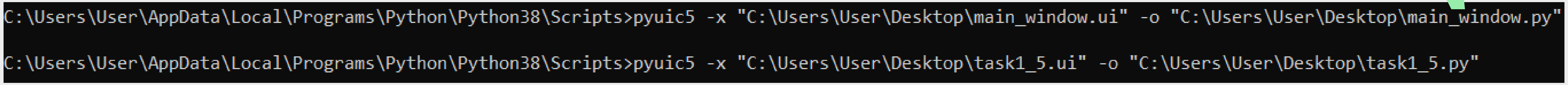
Main Window



```
class Ui_MainWindow(object):
    def setupUi(self, MainWindow):
        MainWindow.setObjectName("MainWindow")
        MainWindow.resize(800, 600)
        MainWindow.setStyleSheet("background-color: rgb(247, 255, 254);")
        self.centralwidget = QtWidgets.QWidget(MainWindow)
        self.centralwidget.setObjectName("centralwidget")
        self.label = QtWidgets.QLabel(self.centralwidget)
        self.label.setGeometry(QtCore.QRect(300, 90, 191, 61))
        self.label.setStyleSheet("font: 75 12pt \"MS Sans Serif\";\n"
                                 "selection-color: rgb(231, 221, 255);\n"
                                 "background-color: rgb(229, 255, 249);")
        self.label.setAlignment(QtCore.Qt.AlignCenter)
        self.label.setObjectName("label")
        self.pushButton = QtWidgets.QPushButton(self.centralwidget)
        self.pushButton.setGeometry(QtCore.QRect(320, 160, 151, 31))
```

font, size, color

Converting .ui into .py file



```
C:\Users\User\AppData\Local\Programs\Python\Python38\Scripts>pyuic5 -x "C:\Users\User\Desktop\main_window.ui" -o "C:\Users\User\Desktop\main_window.py"

C:\Users\User\AppData\Local\Programs\Python\Python38\Scripts>pyuic5 -x "C:\Users\User\Desktop\task1_5.ui" -o "C:\Users\User\Desktop\task1_5.py"
```

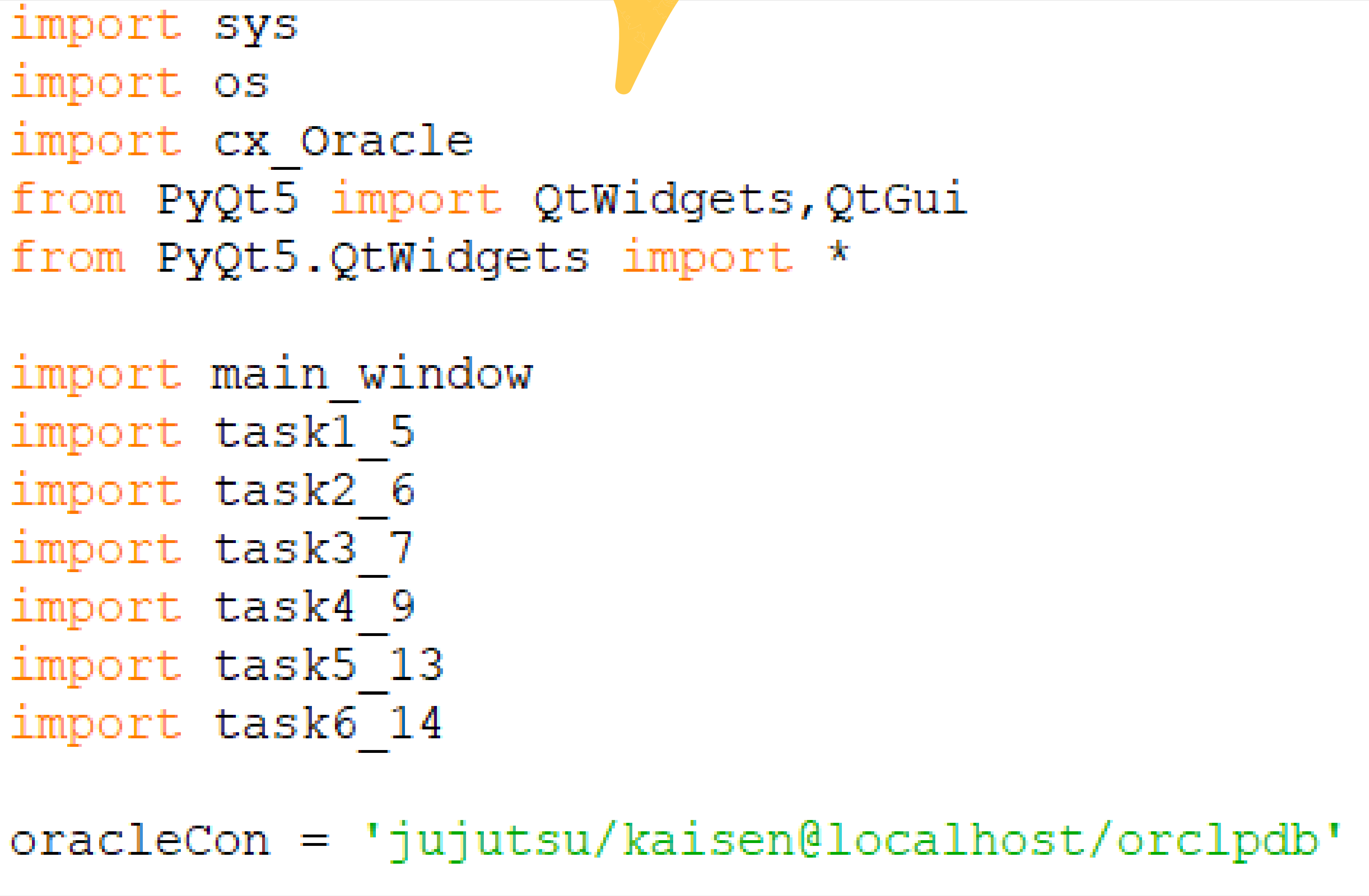
Python

Connection with Oracle

import PyQt5

classes

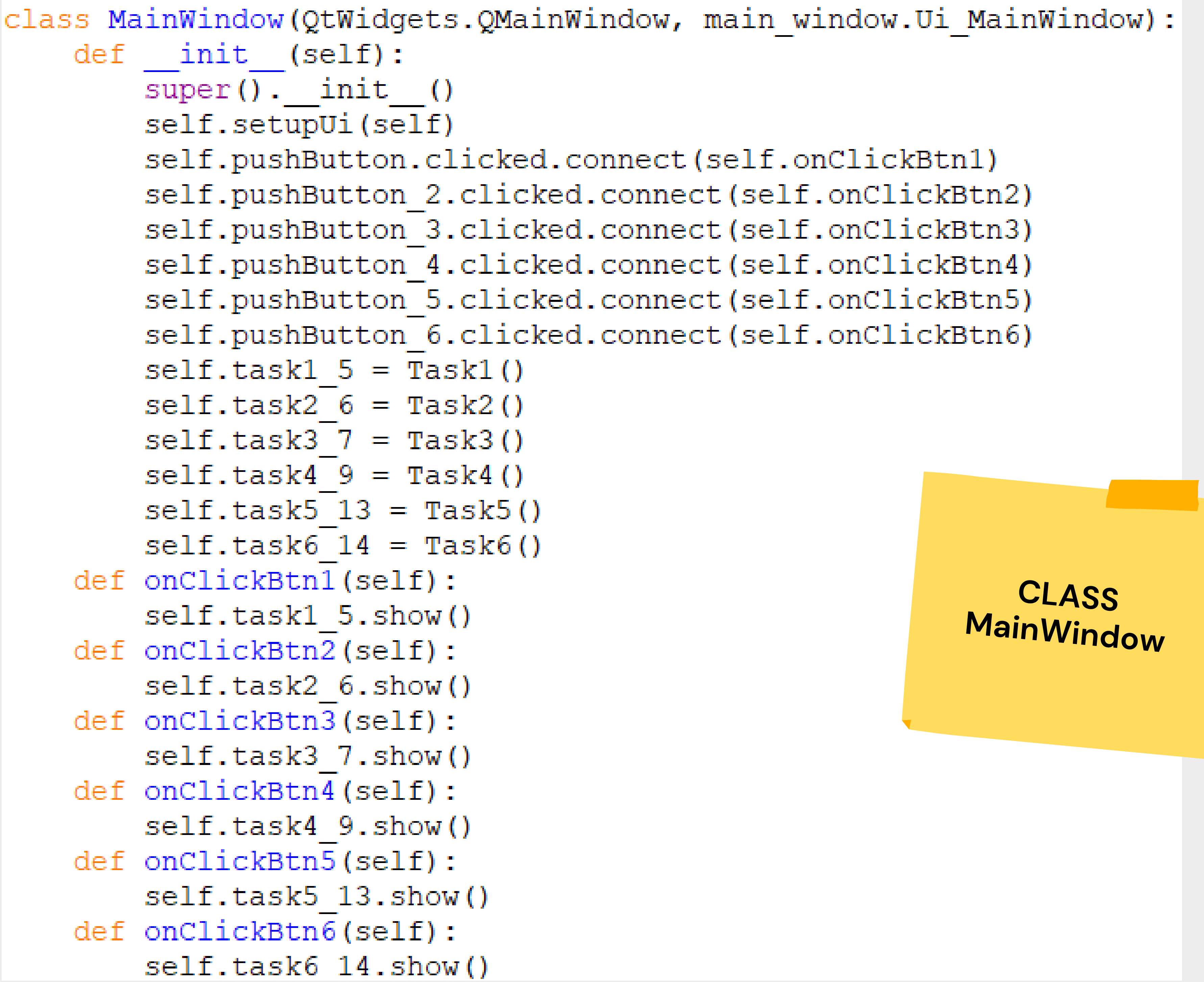
cx_Oracle -> Connection with Oracle database



```python
import sys
import os
import cx_Oracle
from PyQt5 import QtWidgets,QtGui
from PyQt5.QtWidgets import *

import main_window
import task1_5
import task2_6
import task3_7
import task4_9
import task5_13
import task6_14

oracleCon = 'jujutsu/kaisen@localhost/orclpdb'
```



```python
class MainWindow(QtWidgets.QMainWindow, main_window.Ui_MainWindow):
    def __init__(self):
        super().__init__()
        self.setupUi(self)
        self.pushButton.clicked.connect(self.onClickBtn1)
        self.pushButton_2.clicked.connect(self.onClickBtn2)
        self.pushButton_3.clicked.connect(self.onClickBtn3)
        self.pushButton_4.clicked.connect(self.onClickBtn4)
        self.pushButton_5.clicked.connect(self.onClickBtn5)
        self.pushButton_6.clicked.connect(self.onClickBtn6)
        self.task1_5 = Task1()
        self.task2_6 = Task2()
        self.task3_7 = Task3()
        self.task4_9 = Task4()
        self.task5_13 = Task5()
        self.task6_14 = Task6()
    def onClickBtn1(self):
        self.task1_5.show()
    def onClickBtn2(self):
        self.task2_6.show()
    def onClickBtn3(self):
        self.task3_7.show()
    def onClickBtn4(self):
        self.task4_9.show()
    def onClickBtn5(self):
        self.task5_13.show()
    def onClickBtn6(self):
        self.task6_14.show()
```

CLASS MainWindow

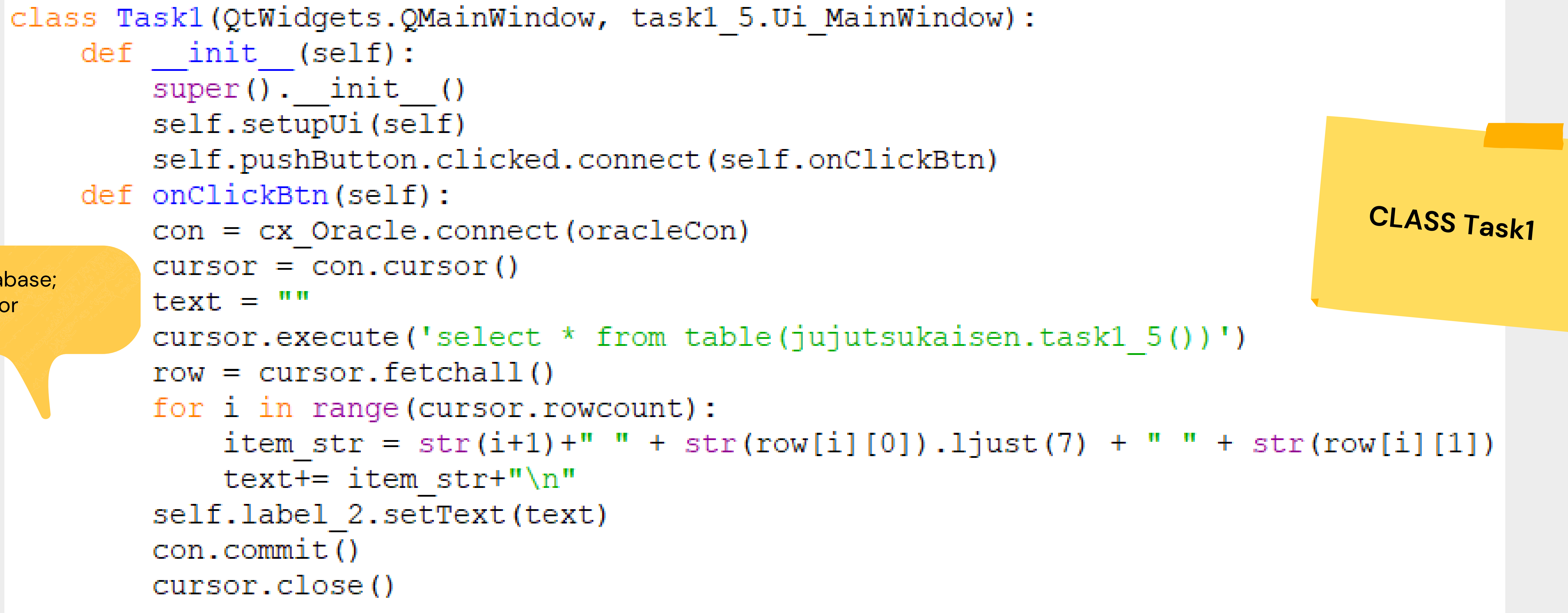
Python

Connection with Oracle

import PyQt5

classes

Connection with Oracle database; fetch records using cursor



```
class Task1(QtWidgets.QMainWindow, task1_5.Ui_MainWindow):
    def __init__(self):
        super().__init__()
        self.setupUi(self)
        self.pushButton.clicked.connect(self.onClickBtn)
    def onClickBtn(self):
        con = cx_Oracle.connect(oracleCon)
        cursor = con.cursor()
        text = ""
        cursor.execute('select * from table(jujutsukaisen.task1_5())')
        row = cursor.fetchall()
        for i in range(cursor.rowcount):
            item_str = str(i+1)+" " + str(row[i][0]).ljust(7) + " " + str(row[i][1])
            text+= item_str+"\n"
        self.label_2.setText(text)
        con.commit()
        cursor.close()
```

CLASS Task1

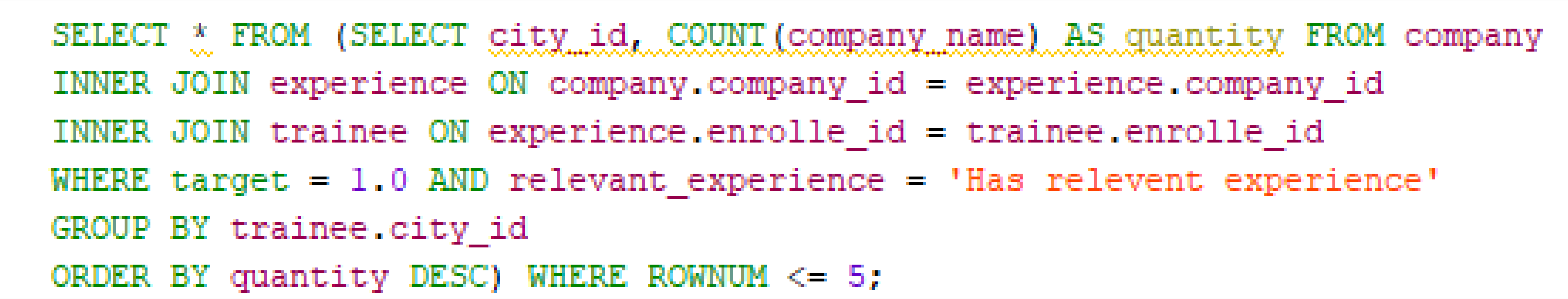
SQL

JOIN

GROUP BY

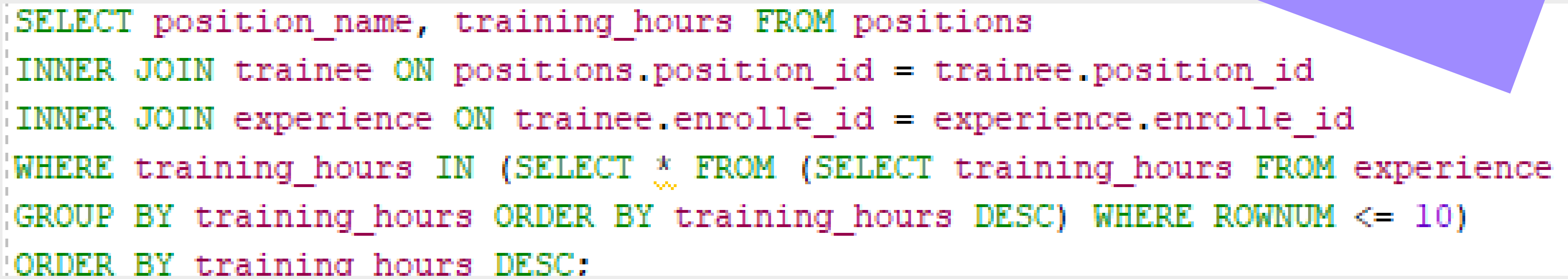
ORDER BY

Subquery, join, group by, order by, rownum



```
SELECT * FROM (SELECT city_id, COUNT(company_name) AS quantity FROM company
INNER JOIN experience ON company.company_id = experience.company_id
INNER JOIN trainee ON experience.enrolle_id = trainee.enrolle_id
WHERE target = 1.0 AND relevant_experience = 'Has relevent experience'
GROUP BY trainee.city_id
ORDER BY quantity DESC) WHERE ROWNUM <= 5;
```

SUBQUERIES



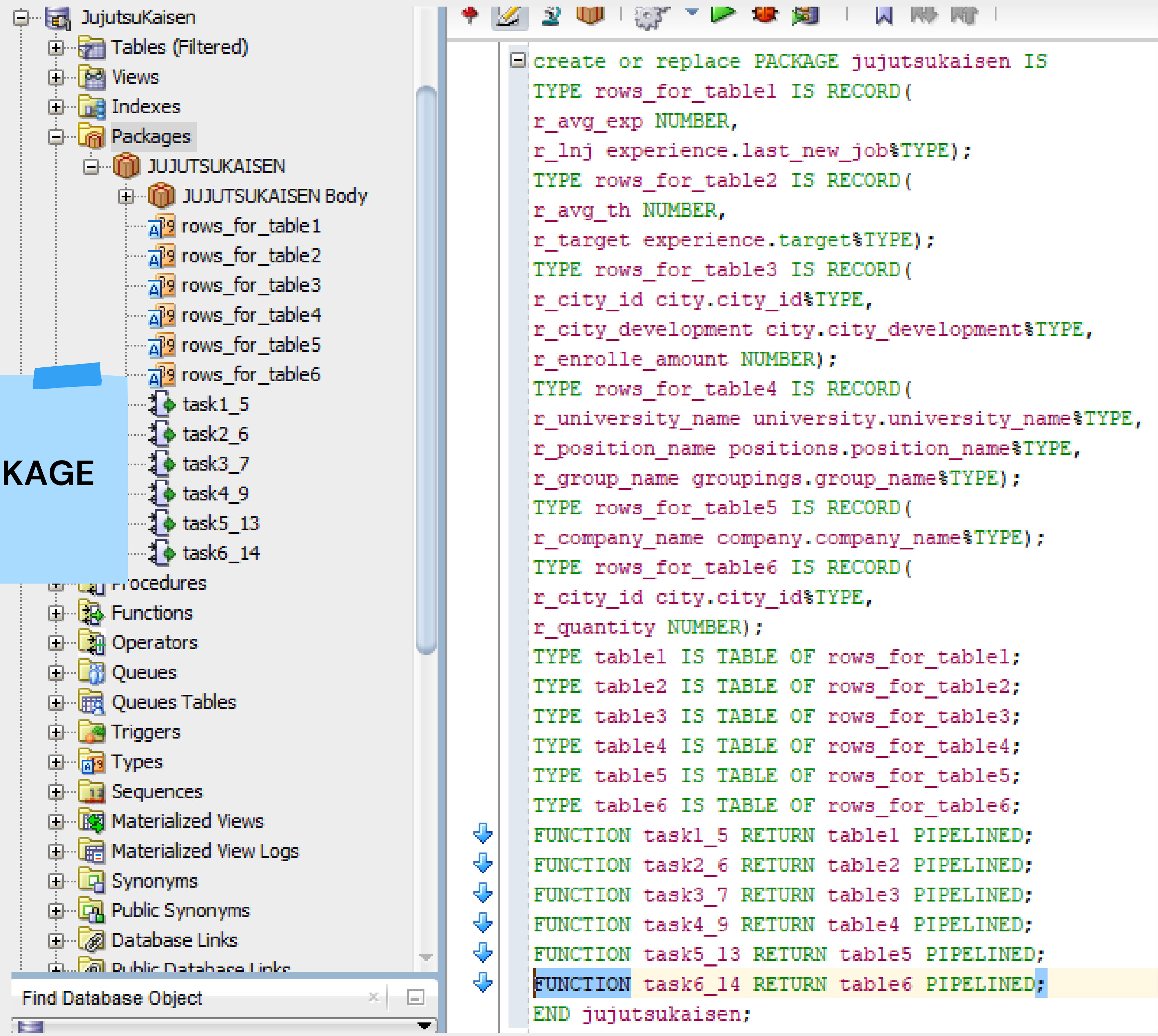
```
SELECT position_name, training_hours FROM positions
INNER JOIN trainee ON positions.position_id = trainee.position_id
INNER JOIN experience ON trainee.enrolle_id = experience.enrolle_id
WHERE training_hours IN (SELECT * FROM (SELECT training_hours FROM experience
GROUP BY training_hours ORDER BY training_hours DESC) WHERE ROWNUM <= 10)
ORDER BY training_hours DESC;
```

PL/SQL

PACKAGE

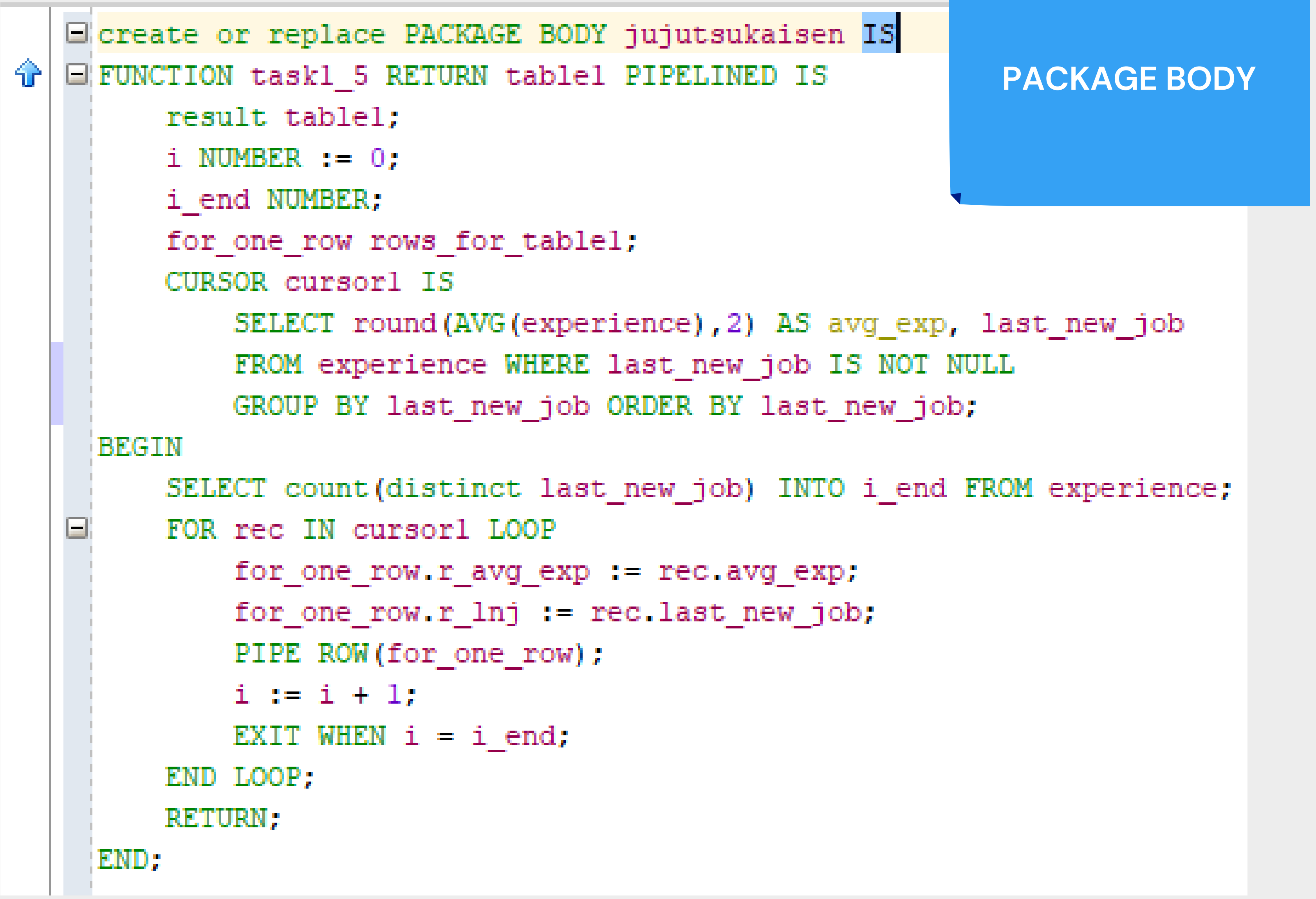
FUNCTION

COLLECTIONS



```
create or replace PACKAGE jujutsukaisen IS
TYPE rows_for_table1 IS RECORD(
r_avg_exp NUMBER,
r_lnj experience.last_new_job%TYPE);
TYPE rows_for_table2 IS RECORD(
r_avg_th NUMBER,
r_target experience.target%TYPE);
TYPE rows_for_table3 IS RECORD(
r_city_id city.city_id%TYPE,
r_city_development city.city_development%TYPE,
r_enrolle_amount NUMBER);
TYPE rows_for_table4 IS RECORD(
r_university_name university.university_name%TYPE,
r_position_name positions.position_name%TYPE,
r_group_name groupings.group_name%TYPE);
TYPE rows_for_table5 IS RECORD(
r_company_name company.company_name%TYPE);
TYPE rows_for_table6 IS RECORD(
r_city_id city.city_id%TYPE,
r_quantity NUMBER);
TYPE table1 IS TABLE OF rows_for_table1;
TYPE table2 IS TABLE OF rows_for_table2;
TYPE table3 IS TABLE OF rows_for_table3;
TYPE table4 IS TABLE OF rows_for_table4;
TYPE table5 IS TABLE OF rows_for_table5;
TYPE table6 IS TABLE OF rows_for_table6;
FUNCTION task1_5 RETURN table1 PIPELINED;
FUNCTION task2_6 RETURN table2 PIPELINED;
FUNCTION task3_7 RETURN table3 PIPELINED;
FUNCTION task4_9 RETURN table4 PIPELINED;
FUNCTION task5_13 RETURN table5 PIPELINED;
FUNCTION task6_14 RETURN table6 PIPELINED;
END jujutsukaisen;
```

PACKAGE



```
create or replace PACKAGE BODY jujutsukaisen IS
FUNCTION task1_5 RETURN table1 PIPELINED IS
    result table1;
    i NUMBER := 0;
    i_end NUMBER;
    for_one_row rows_for_table1;
    CURSOR cursor1 IS
        SELECT round(AVG(experience),2) AS avg_exp, last_new_job
        FROM experience WHERE last_new_job IS NOT NULL
        GROUP BY last_new_job ORDER BY last_new_job;
BEGIN
    SELECT count(distinct last_new_job) INTO i_end FROM experience;
    FOR rec IN cursor1 LOOP
        for_one_row.r_avg_exp := rec.avg_exp;
        for_one_row.r_lnj := rec.last_new_job;
        PIPE ROW(for_one_row);
        i := i + 1;
        EXIT WHEN i = i_end;
    END LOOP;
    RETURN;
END;
```

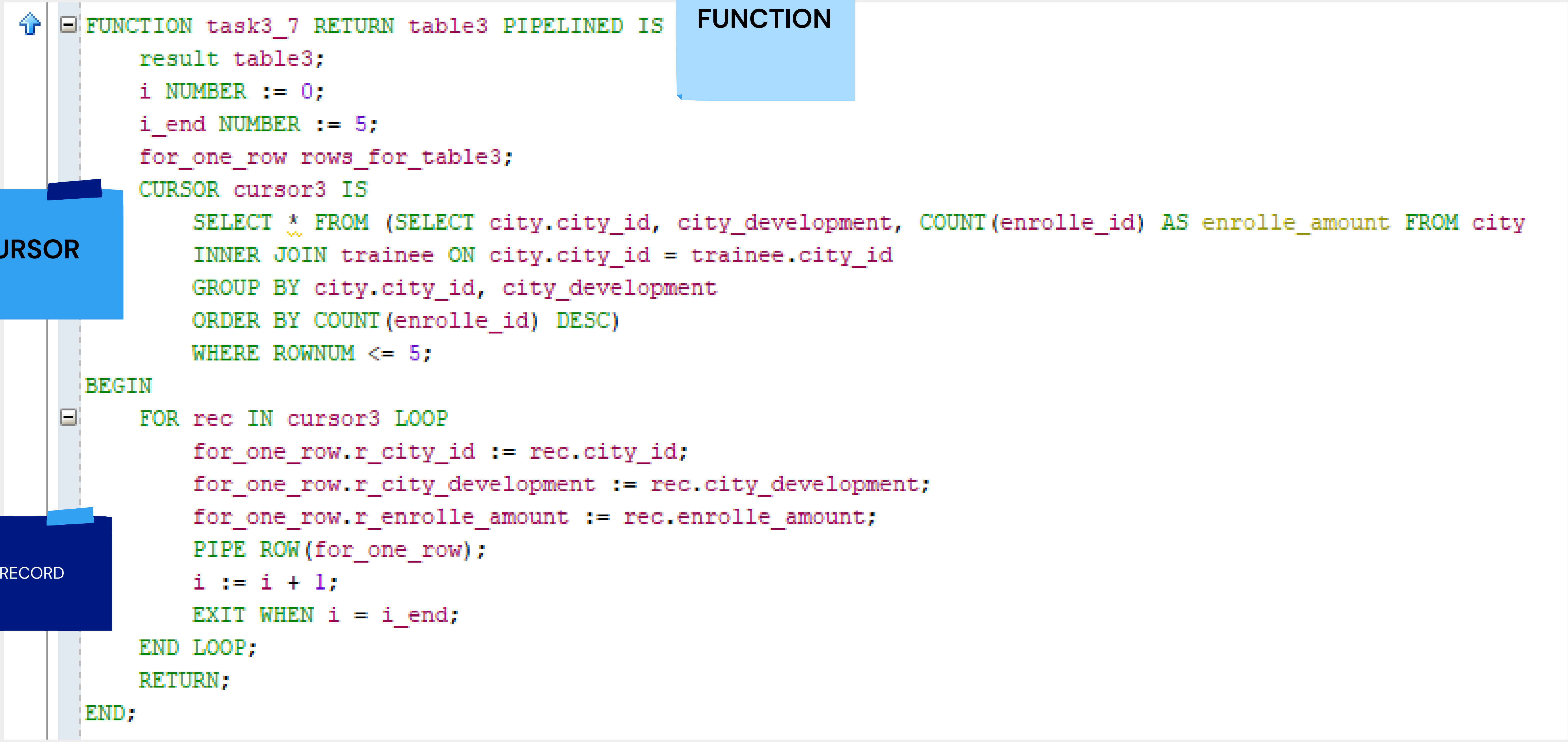
PACKAGE BODY

PL/SQL

PACKAGE

FUNCTION

COLLECTIONS



```
FUNCTION task3_7 RETURN table3 PIPELINED IS
    result table3;
    i NUMBER := 0;
    i_end NUMBER := 5;
    for_one_row rows_for_table3;
    CURSOR cursor3 IS
        SELECT * FROM (SELECT city.city_id, city_development, COUNT(enrolle_id) AS enrolle_amount FROM city
        INNER JOIN trainee ON city.city_id = trainee.city_id
        GROUP BY city.city_id, city_development
        ORDER BY COUNT(enrolle_id) DESC)
        WHERE ROWNUM <= 5;
BEGIN
    FOR rec IN cursor3 LOOP
        for_one_row.r_city_id := rec.city_id;
        for_one_row.r_city_development := rec.city_development;
        for_one_row.r_enrolle_amount := rec.enrolle_amount;
        PIPE ROW(for_one_row);
        i := i + 1;
        EXIT WHEN i = i_end;
    END LOOP;
    RETURN;
END;
```

FUNCTION

CURSOR

RECORD

Have a great day ahead.

Thank you for your attention!