Witaj w Twojej pierwszej pracy domowej poświęconej nauce programowania.  
Na kolejnych stronach znajdziesz zadania do wykonania, które dla Ciebie przygotowałem.

PS. Nie zapomnij przywitać się na grupie i oznaczyć pracy domowej hasztagiem #MODUŁ3.

Jeśli masz problem z którymś z zadań i chcesz bym sprawdził Twoją pracę domową to:

* umieść pracę domową na dysku google(tak, bym nie musiał pobierać pliku)
* oznacz mnie w poście i napisz z czym masz konkretnie problem

Nie ma znaczenia czy Twój plik ma rozszerzenie .docx czy .sql. Zrób jak   
Ci wygodniej :)

# **ZADANIE 1: Stwórz użytkownika kurs\_plsql w Twojej bazie danych i zaimportuj do niego przygotowany przeze mnie plik.**

Uwaga. Tworzony użytkownik będzie nam potrzebny we wszystkich kolejnych modułach kursu.

Instrukcja:

1. Jako użytkownik system(administrator) wykonaj poniższe polecenia:
   1. ALTER SESSION SET CONTAINER = XEPDB1; - u Ciebie może być inna nazwa kontenera(jeśli podałeś podczas instalacji inną niż ja)

PS. Nie martw się, jeśli nie rozumiesz czym jest kontener. Nie jest to w naszym kursie do niczego potrzebne, ale jeśli należysz do ciekawskich to wspominałem o tym krótko w nagraniu o instalacji bazy danych: https://www.youtube.com/watch?v=oFKzUUOVX-I&list=PL8g9gILZp1OJaZTCpRmxnAAPqb7b2uHy0&index=603&t=645s

**Krok “a” jest jedynie po to by móc wykonać dalsze kroki instrukcji. Jeśli korzystasz z wersji bazy danych Oracle 11g i wcześniejszych to krok “a” pomiń, bo kontenery pojawiły się od wersji 12c.**

* 1. CREATE USER kurs\_plsql IDENTIFIED BY hasło;
  2. GRANT CREATE SESSION TO kurs\_plsql;
  3. GRANT UNLIMITED TABLESPACE TO KURS\_PLSQL;
  4. GRANT CREATE TABLE TO kurs\_plsql;
  5. GRANT CREATE SEQUENCE TO KURS\_PLSQL;
  6. GRANT CREATE VIEW TO KURS\_PLSQL;

1. Wykonajj skrypt z tego linku: https://drive.google.com/file/d/1MYIBFytRj35-07hWTl85pNuIpScj5YeX/view?usp=sharing

Wystarczy, że zawartość pliku przekopiujesz i wkleisz do Sql Developera a następnie naciśniesz przycisk “run script”(ten zielony trójkącik).

Gratulację! Od teraz masz w swojej bazie danych użytkownika z którym będziemy pracować przez pozostałe lekcje programu :)

# **ZADANIE 2 : ZADANIA DO SAMODZIELNEGO WYKONANIA.**

Uwaga. Wszystkie zadania należy wykonywać będąc zalogowanym jako użytkownik kurs\_plsql, chyba, że wskazano inaczej.

1. Napisz program, który wyliczy wskaźnik BMI osoby o następujących danych. Zmienne nazwij wedle uznania.
   * wzrost 180cm
   * waga: 80kg
   * wzór na BMI: waga/wzrost do kwadratu

Na konsoli wyświetl napis: Twój wskaźnik BMI to: TWÓJ\_WYNIK.

1. Stwórz program, który wyliczy cenę netto produktu. Zmienne i stałe nazwij wedle uznania.
   * cena brutto: 1600zł(w tym VAT 23%)
   * użyj stałej zamiast zmiennej dla stawki podatku
   * wyświetl w konsoli cenę netto używając do tego zmiennej
2. Dodaj w bloku PL/SQL nowy produkt do tabeli “produkty” o nazwie “monitor”.
   * Użyj instrukcji RETURNING INTO by zwrócić identyfikator(product\_id) nowego produktu
   * Odczytaną wartość przypisz do zmiennej, która korzysta z kotwiczenia(%TYPE)
3. Stwórz program z 3 zmiennymi:
   * v\_imie, typ varchar, 40 znaków
   * v\_wiek, typ numeryczny, rozmiar 2 cyfr
   * w sekcji wykonawczej przypisz Twoje dane do obu zmiennych
   * na konsoli wyświetl napis: TWOJE\_IMIE ma TWÓJ\_WIEK lat, np. Darek ma 29 lat
4. Oblicz obwód i pole koła o promieniu 10. Wykorzystaj stałą do deklaracji liczby pi.

SET SERVEROUTPUT ON;

--ZAD 1 MODUL 3

DECLARE

wzrost\_cm number:=180;

waga\_kg number :=80;

BMI number;

Begin

BMI := wzrost\_cm/waga\_kg;

DBMS\_OUTPUT.PUT\_LINE('Twoj wskaznik BMI to:'||' '||BMI);

END;

-- ZAD 2 MODUL 3

DECLARE

cena\_brutto number :=1600;

stawka constant number := 0.23;

cena\_netto number;

BEGIN

cena\_netto := round(cena\_brutto/(1+stawka),2);

dbms\_output.put\_line('cena netto'||' '||cena\_netto||' '|| 'zl');

END;

-- zad 3 modul 3

--DROP TABLE produkty;

CREATE TABLE produkty (id NUMBER GENERATED ALWAYS AS IDENTITY PRIMARY KEY, nazwa VARCHAR2(30));

DECLARE

product\_id NUMBER;

product\_id2 product\_id%TYPE;

BEGIN

INSERT INTO produkty (nazwa) VALUES('monitor')

RETURNING id INTO product\_id;

dbms\_output.put\_line(product\_id);

--DROP TABLE produkty;

END;

-- zad 4 modul 3

DECLARE

v\_imie VARCHAR(40);

v\_wiek NUMBER(2);

napis VARCHAR(50);

BEGIN

v\_imie :='Krzysztof';

v\_wiek := 40;

napis := v\_imie||' '||'ma'||' '||v\_wiek||' '||'lat';

DBMS\_OUTPUT.PUT\_LINE(napis);

END;

--- zad 5 modul 3

DECLARE

r\_promien number := 10;

pi CONSTANT number := 3.14;

p\_pole number;

o\_obwod number;

BEGIN

p\_pole := pi\*power(r\_promien, 2);

dbms\_output.put\_line('pole = ' ||p\_pole);

o\_obwod := 2\*pi\*r\_promien;

dbms\_output.put\_line('obwod = ' ||o\_obwod);

END;