Witaj w kolejnej pracy domowej.  
Na kolejnych stronach znajdziesz zadania do wykonania, które pomogą Ci lepiej zrozumieć i utrwalić wiedzę dotyczącą instrukcji sterujących i pętli.

PS. Twoją pracę domową oznacz hasztagiem #MODUŁ4.

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 :)

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

1. Rozszerz funkcjonalność programu z poprzedniego modułu, który liczył wskaźnik BMI.

Wyświetl odpowiedni komunikat na konsoli w zależności od wyniku:

* <16 “ale z Ciebie niejadek”
* >=16 i <25 “dobra forma, tak trzymaj”
* >=25 “zrób kilka pompek”

\*\*\*Dla chętnych\*\*\* Postaraj się wykonać zadanie w ten sposób, by w całym programie instrukcję wypisującą coś na konsoli(dbms\_output) użyć tylko raz.

**DECLARE**

**wzrost\_cm number:= to\_number('1.8','9.99');**

**waga\_kg number :=-80;**

**BMI number(4,2);**

**napis Varchar(150) ;**

**Begin**

**BMI := (waga\_kg/power(wzrost\_cm,2));**

**if BMI>=0 OR BMI<0 then**

**IF BMI < 16 AND BMI >=0 THEN**

**napis := 'ale z Ciebie niejadek';**

**ELSIF BMI>=16 AND BMI<25 THEN**

**napis := 'dobra forma, tak trzymaj';**

**ELSIF BMI>=25 THEN**

**napis :='zrób kilka pompek';**

**ELSIF BMI<0 then napis :='bledne dane';**

**— dodana instrukcja obslugujaca wartosc ujemną BMI**

**end if;**

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

**END IF;**

**END;**

1. Napisz program zwracający liczby od 20 do 41 co 3, tzn. 20, 23, 26..41. Na konsoli wyświetl napis: “aktualna wartość: x , poprzednia wartość: y”, gdzie x i y to odpowiednie wartości zmiennej.

Uwaga. Możesz wykorzystać dowolny rodzaj pętli.

**DECLARE**

**Y NUMBER :=17;**

**X NUMBER ;**

**BEGIN**

**LOOP**

**y := y+3;**

**x := y-3;**

**DBMS\_OUTPUT.PUT\_LINE('aktualna wartosc'||' '||Y||','||'poprzednia wartosc'||' '||x);**

**EXIT WHEN Y=41;**

**END LOOP;**

**END;**

1. Napisz trzy programy, wyświetlające wyłącznie liczby parzyste z zakresu od 1 do 10. Każdy program powinien użyć innego rodzaju pętli(simple, while loop, for loop).

**-- petla simple**

**DECLARE**

**X NUMBER :=0;**

**BEGIN**

**LOOP**

**X := X+2;**

**DBMS\_OUTPUT.PUT\_LINE(X);**

**EXIT WHEN X=10;**

**END LOOP;**

**END;**

**--PETLA WHILE LOOP**

**DECLARE**

**X NUMBER :=0;**

**BEGIN**

**—WHILE x between 0 and 8**

**WHILE X <10**

**LOOP**

**X := X+2;**

**DBMS\_OUTPUT.PUT\_LINE(X);**

**END LOOP;**

**END;**

**-- pętla for**

**DECLARE**

**x NUMBER :=0;**

**BEGIN**

**FOR i IN 0..10**

**LOOP**

**x:=i+2;**

**if mod(x,2)=0 and x<=10 then**

**DBMS\_OUTPUT.PUT\_LINE(x);**

**END IF;**

**END LOOP;**

**END;**

1. Stwórz program z użyciem pętli zagnieżdżonej, który będzie iterował 3 razy po pętli zewnętrznej i 3 raz po pętli wewnętrznej za każdym razem wyświetlając numer obiegu każdej pętli, tzn. 1.1, 1.2, 1.3, 2.1 …, 3.2, 3.3
   1. pętla zewnętrzna: for, wewnętrzna: simple
   2. pętla zewnętrzna: simple, wewnętrzna: for
   3. pętla zewnętrzna: while, wewnętrzna: for

–a: pętla zewnętrzna: for, wewnętrzna: simple

**DECLARE**

**x number :=0;**

**BEGIN**

**FOR i IN 1..3**

**LOOP**

**x:=0;**

**LOOP**

**x:=x+1;**

**DBMS\_OUTPUT.PUT\_LINE(i||'.'||x);**

**EXIT WHEN X=3;**

**END LOOP;**

**END LOOP;**

**END;**

b: pętla zewnętrzna: simple, wewnętrzna: for

**DECLARE**

**x number :=0;**

**BEGIN**

**LOOP**

**x:=x+1;**

**FOR i in 1..3**

**LOOP**

**DBMS\_OUTPUT.PUT\_LINE(i||'.'||x);**

**END LOOP;**

**EXIT WHEN X=3;**

**END LOOP;**

**END;**

d: pętla zewnętrzna: while, wewnętrzna: for

**DECLARE**

**x number :=0;**

**BEGIN**

**WHILE x<3**

**LOOP**

**x:=x+1;**

**FOR i in 1..3**

**LOOP**

**DBMS\_OUTPUT.PUT\_LINE(i||'.'||x);**

**END LOOP;**

**END LOOP;**

**END;**

1. Wyjaśnij czym różni się EXIT od CONTINUE.

**EXIT skutkuje opuszczeniem pętli. CONTINUE powoduje kontynuację iteracji w pętli.**

\*\*\*Zadanie dla chętnych\*\*\*

Rozwiń program z lekcji 13 tego modułu. Gdy uda Ci się znaleźć schronienie przed deszczem to zamiast czekać aż przestanie padać sprawdź czy wziąłeś ze sobą parasolkę.

* proces szukania parasolki trwa 2 sekundy
* informację o tym czy zabrałeś ze sobą parasolkę przypisz do zmiennej typu BOOLEAN
* jeśli masz ze sobą parasolkę to kontynuuj spacer wychodząc ze schronienia
* jeśli nie masz parasolki program działa po staremu, tzn. czekasz w schronieniu aż przestanie padać