**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**НАЦІОНАЛЬНОМУ УНІВЕРСИТЕТІ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”**

**Кафедра систем штучного інтелекту**

**Лабораторна робота №6**

з дисципліни

«Системний аналіз»

**Варіант №25**

**Виконав:**

студент групи КН-207

Шиманський П.С.

**Викладач:**

Бойко Н.І.

Львів – 2019

**Шаблон Стан**

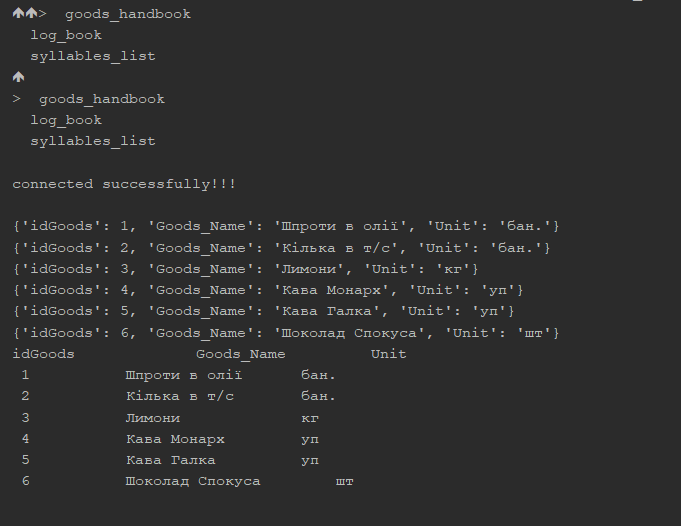
**Опис використання в програмі**

Створення шаблону Стан для виведення потрібних нам елементів з нашої бази даних у зрозумілому вигляді

**Код:**

import pymysql.cursors  
import os  
import keyboard  
  
  
print\_db\_row=True  
type\_test=False  
  
def connect():  
 connection = pymysql.connect(  
 host = "localhost",  
 user = "root",  
 password = "911102Pavlo320099",  
 db = "sa1",  
 charset = "utf8mb4",  
 cursorclass=pymysql.cursors.DictCursor  
 )  
 print("connected successfully!!!")  
 return connection  
  
#######################################additional module##################  
def cls():  
 os.system(['clear','cls'][os.name == 'nt'])  
cls()  
  
stop=False  
def stop():  
 global stop  
 stop=True  
keyboard.add\_hotkey('esc', stop)  
  
def create\_main\_menu(menu):  
 global show\_menu  
 for i in range(0, len(menu)):  
 show\_menu.append(" ")  
 menu[i]+="\n"  
 show\_menu[i+1]+=menu[i]  
 cls()  
 for e in show\_menu:  
 print(e, end="")  
   
def upchose():  
 global show\_menu  
 global current  
 global menu  
 if (current>0):  
 show\_menu[current]=show\_menu[current-1]  
 show\_menu[current-1]=">"  
 current-=1  
 cls()  
 print()  
 for e in show\_menu:  
 print(e, end="")  
   
def downchose():  
 global show\_menu  
 global current  
 global menu  
 if (current<len(menu)-1):  
 show\_menu[current]=show\_menu[current+1]  
 show\_menu[current+1]=">"  
 current+=1  
 cls()  
 print()  
 for e in show\_menu:  
 print(e, end="")  
   
def enter():  
 global show\_menu  
 global current  
 cls()  
 print()  
 for e in show\_menu:  
 print(e, end="")  
 chose(show\_menu[current+1])  
  
menu=["goods\_handbook", "log\_book", "syllables\_list"]  
show\_menu=[">"]  
current=0  
output=["\n", " "]  
  
create\_main\_menu(menu)  
  
def chose(point):  
 chose={" goods\_handbook\n" : operand1.get\_goods\_handbook,  
 " log\_book\n" : operand2.get\_log\_book,  
 " syllables\_list\n" : operand3.get\_syllables\_list  
 }  
 printer.show\_beautifully(main.adapter(chose[point]()))  
 chose[point]  
  
#######################################################################  
   
class printer():  
 def show\_beautifully(array):  
 for x in array:  
 for y in x:  
 if(type(y)==type("test") and len(y)<9):  
 print(y, "\t\t\t", end=" ")  
 elif(type(y)==type("test") and len(y)<17):  
 print(y, "\t\t", end=" ")  
 elif(type(y)==type("test") and len(y)>17):  
 print(y, "\t", end=" ")  
 elif(type(y)==type(type\_test)):  
 print(y, "\t\t", end=" ")  
 else:  
 print(y, "\t\t\t", end=" ")  
 print("\n", end=" ")  
  
class operand1():  
 def get\_goods\_handbook():  
 connection = connect()  
 try:  
 with connection.cursor() as cursor:  
 sql = "select \* from goods\_handbook"  
 cursor.execute(sql)  
 print()  
 ret=[]  
 for row in cursor:  
 if (print\_db\_row==True):  
 print(row)  
 ret.append(row)  
 finally:  
 connection.close()  
 return ret  
  
class operand2():  
 def get\_log\_book():  
 connection = connect()  
 try:  
 with connection.cursor() as cursor:  
 sql = "select \* from log\_book"  
 cursor.execute(sql)  
 print()  
 ret=[]  
 for row in cursor:  
 if (print\_db\_row==True):  
 print(row)  
 ret.append(row)  
 finally:  
 connection.close()  
 global type\_test  
 type\_test=ret[1]["date"]  
 return ret  
  
class operand3():  
 def get\_syllables\_list():  
 connection = connect()  
 try:  
 with connection.cursor() as cursor:  
 sql = "select \* from syllables\_list"  
 cursor.execute(sql)  
 print()  
 ret=[]  
 for row in cursor:  
 if (print\_db\_row==True):  
 print(row)  
 ret.append(row)  
 finally:  
 connection.close()  
 return ret  
  
class main():  
 def adapter(dbarray):  
 output=[]  
 a=[]  
 for b in dbarray[0]:  
 a.append(b)  
 output.append(a)  
 for row in dbarray:  
 elements=[]  
 for element in row:  
 elements.append(row[element])  
 output.append(elements)  
 return output  
  
keyboard.add\_hotkey('up', upchose)  
keyboard.add\_hotkey('down', downchose)  
keyboard.add\_hotkey('enter', enter)  
  
while True:  
 if (stop==True):  
 break  
  
  
#####

**Результат:**



**Висновок:**

На цій лабораторній роботі, я ознайомився з шаблоном стан, а також програмно його застосував.