List 3
Classes

1. Write a Python program to create a person class. Include attributes like name, country and date of birth. Implement a method to calculate the person's age.
2. Write a Python class BankAccount with attributes like account_number, balance, date_of_opening and customer_name, and methods like deposit, withdraw, and check_balance.
3. Write a Python class Employee with attributes like emp_id, emp_name, emp_salary, and emp_department and methods like calculate_emp_salary, emp_assign_department, and print_employee_details.

   Sample Employee Data:

"ADAMS",	"E7876",	50000,	"ACCOUNTING"
"JONES",	"E7499",	45000,	"RESEARCH"
"MARTIN",	"E7900",	50000,	"SALES"
"SMITH",	"E7698",	55000,	"OPERATIONS"

   - Use 'assign_department' method to change the department of an employee.
   - Use 'print_employee_details' method to print the details of an employee.
   - Use 'calculate_emp_salary' method takes two arguments: salary and hours_worked, which is the number of hours worked by the employee. If the number of hours worked is more than 50, the method computes overtime and adds it to the salary. Overtime is calculated as following formula:

     overtime = hours_worked - 50
     overtime amount = (overtime * (salary / 50))

4. Write a Python program to create a class that represents a shape. Include methods to calculate its area and perimeter. Implement subclasses for different shapes like circle, triangle, and square.
5. Write a program that generates a set of samples of a sine wave. The parameters are as follows: Frequency = last 2 digits of your index number [Hz] (if it is ‘00’ then use ‘01’), Sampling Frequency = 48 [kHz], Acquisition time = 2 [s], Amplitude = 2. Hint – use numpy and matplotlib libraries. It should be realized as a class which stores the sine wave, has a method to generate sine wave samples with chosen parameters, plotting the sine wave as well as returning samples of downsampled wave. Plot both sets on one figure for comparison.
6. Write a program with the same functionality as in List 1 task 6. It should be realized as a ‘Player’ class, where each of the player is a separate object. Inside it stores the score of the player and has method for calculating the score of a given word. Make a simple text UI to enhance the program's accessibility for players.
7. *For next classes get familiar with Digilent Analog Discovery 2 board and its python API - pydwf.