Task 4 – [Web App] NAPFA Test System

The School engages you to design an NAPFA Test System using web application. In this task, you are required to implement a prototype using normalised database and flask web application to manage these records.

NAPFA Standard for Females

Age group	Grade	Points	No. of Sit-ups in 1 min	Standing Broad Jump	Sit & Reach Distance	No. of Inclined Pull-ups in 30 sec	4 X 10m Shuttle Run Time	2.4 km Run- Walk time (min : sec)
12	A	5	>29	>167cm	>39cm	>15	<11.5 sec	<14:41
	B	4	25-29	159-167	37-39	13-15	11.5-11.9	14:41-15:40
	C	3	21-24	150-158	34-36	10-12	12.0-12.3	15:41-16:40
	D	2	17-20	141-149	30-33	7-9	12.4-12.7	16:41-17:40
	E	1	13-16	132-140	25-29	3-6	12.8-13.2	17:41-18:40
…	…	…	…	…	…	…	…	…

The following information of each `Student` is stored:
`MatricNo` – unique string in the format of `"RVHS-YYYY-XXX"` where `YYYY` is the year of entry to school and `XXX` is a 3-digit string ranged from `"001"` to `"999"`.
`Name` – name of student
`Class` – class of student
`IndexNo` – index number of the student in the class
`Gender` – gender of student, to be stored as a single character, using either `"M"` or `"F"`
`BirthYear` – birth year of student

The following information of each `Standard` is stored:
`Age` – Age of the student
`Gender` – type of the apparel
`Item` – Item assessed, such as SitUp or Jump
`F,E,D,C,B` – The max/min number needed to get these grades, anything beyond B grade will be considered A grade. e.g, the information of the above table is stored in the following format:

```
Age,Gender,Item,F,E,D,C,B
12,F,SitUp,12,16,20,24,29,
12,F,Jump,131,140,149,158,167,
12,F,SitReach,24,29,33,36,39,
12,F,PullUp,2,6,9,12,15,
12,F,Shuttle,13.3,12.8,12.4,12.0,11.5,
12,F,Run24,18:41,17:41,16:41,15:41,14:41,
```

The following information of each `Result` is stored:
`MatricNo` – matric number of the student
`Year` – year which the NAPFA test is taken
`SitUp,Jump,SitReach,PullUp,Shuttle,Run24` – Result for the 6 items

The information is to be stored in three tables:
`Student`
`Standard`
`Result`

Task 4.1

Create an SQL file called `Task4_1.sql` to show the SQL code to create the database `napfa.db` with the three tables.

The table `Student` must use `MatricNo` as its primary key, and the table `Standard` must use both `Age, Gender, Item` as its primary key. The table `Result` should use `MatricNo` and `Year` as a composite key, while `MatricNo` must refer to `MatricNo` in `Student` as foreign keys.

Save your SQL code as
`Task4_1.sql`

Task 4.2

The files `students.csv`, `standards.csv` and `results.csv` contains information about the student, apparels and the past loan records. The first row of each file contains the header of the respective columns. Each row in the files is a comma-separated list of information.

Write a Python program to insert all information from the three files into the database `napfa.db`. Run the program.

Save your program code as
`Task4_2.py`

Task 4.3

Teacher would like to query all the students who did not pass (scored “F”) for 2.4km run in year 2021.

Write the SQL code required.

Save this code as
`Task4_3.sql`

Task 4.4

Create a web application using the flask micro-framework with the following requirements:

- Takes in student information, including gender, year of assessment, year of birth, and his/her NAPFA result
- Display the respective grades for this student.

Task 4.5

Create a web application using the flask micro-framework with the following requirements:

- Takes in the class and year of assessment
- Display the all NAPFA results of students from this class in a table format.

Task 4.6 [Optional]

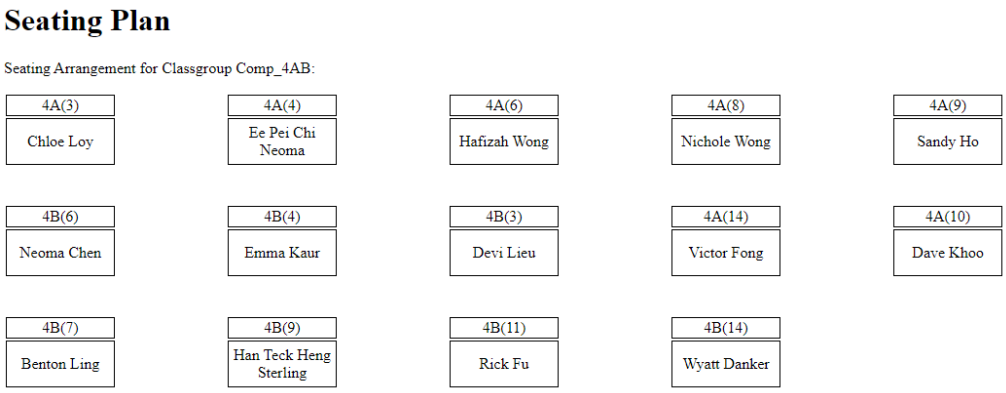
Based on the following table, further polish up the programme to include CRUD operations which would suit for real-life requirements and needs for both PE department and students.

AWARD REQUIREMENTS	
Awards	**Minimum Scores**
Gold	C grade in all 6 stations with a minimum of 21 points
Silver	D grade in all 6 stations with a minimum of 15 points
Bronze	E grade in all 6 stations with a minimum of 6 points

Task 2.5 [Bonus]

Lastly, if you would like to challenge yourself, copy paste your files in folder `Task_2_4` to a new folder named `Task_2_5`.
Attempt to modify your code to suit the following needs:

1. Search form can allow user to specify number of columns.
2. Snake shape with odd rows starting from left, even rows starting from right.
3. Use css style to display the data in the following format.



[2]