Exam PCS2 – Trainee Registration Form June 2017

Time: 8:45 - 11:15 h (150 minutes).

ADMITTED RESOURCES:

- You are allowed to use everything on paper (books, notes, etc.) and on your laptop, but only what you bring in: you are not allowed to borrow something from someone else.
- During the exam it is not allowed to use the network. You should make the exam yourself: so no communication with MSDN or google for help and no communication with other students, like using Facebook, e-mail, Skype, Dropbox, gsm or whatever.

EXAM POINTS: see assignments

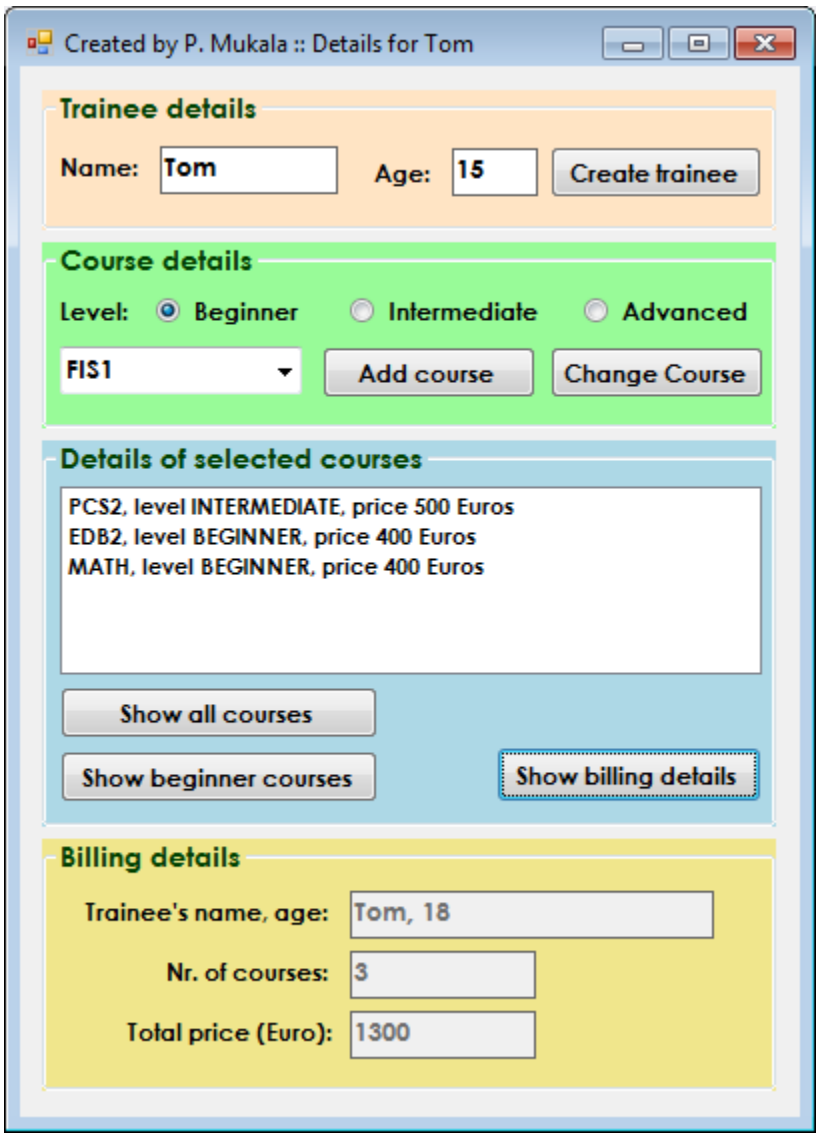
THE APPLICATION

You are asked to develop an application that allows a trainee to register for certain courses. A screenshot of the application is given on the next page.

The registration form contains four main sections:

- **Trainee details**
  This section allows to enter a trainee's name and age.
- **Course details**
  In this section, the trainee can see which courses are available and can thus choose the course he/she wishes to register for, as well as the level of proficiency (Beginner, Intermediate or Advanced).

  The trainee can register for a new course, or can make changes to a course that he/she is already registered for.
- **Details of selected courses**
  This section displays information about courses (all courses or only beginner courses). This section also allows the user to show billing details in the next section.
- **Billing details**
  This section shows some details pertaining to trainee and the corresponding registrations.



Now open the start-up project in Visual Studio. You can see that there is a form with controls on it (see the picture above). During this exam, you will implement two classes: class **Trainee** and class **Course**. You will make use of the members of these two classes to make Form1 operational (working).

Important note about naming - Whenever you see a **bold word**, you have to use this word as the name of a class, enum, method, field, property or whatever.

In the following assignments we ask you to implement certain things. If you feel the need to add more methods, fields or properties, then feel free to do so.

Assignment 1: Class Course (6 + 5 + 5 + 9 pts)

The project has a class **Course**, but it is not implemented yet.

a) Add fields to the class **Course**. It is not allowed to access any of these fields from outside this class. You may add associated properties if you like.
- A field **code** of type string that holds the course code (for instance "PCS2");
- A field **level** that holds the level of this course; the type of the field level should be an enumeration **CourseLevel** which can hold three values: BEGINNER, INTERMEDIATE, and ADVANCED. You also need to add such an enumeration to your project.

b) Add a constructor to this class with 2 parameters: a parameter for its code and a parameter for its level. The values of these parameters should be assigned to the associated fields.

c) Add a method **GetPrice** that returns the price of this course. The price for a course on level BEGINNER is 400 Euros; a course on level INTERMEDIATE is 500 Euros and on ADVANCED level 600 Euros.

d) Add a method **AsString** that returns a string containing all information about the current course. When executed, a sample value from the method would be: "**PCS2, level INTERMEDIATE, price 500 Euros**".

Assignment 2: Class Trainee (5 + 5 + 3 + 12 + 10 + 10 pts)

Add a class **Trainee** to your project. A trainee has a name and an age. Furthermore, a trainee has a collection of courses the trainee is interested in.

a) Add the following fields and properties to this class:
- A property **Name** of type string that holds the trainee's name;
- A property **Age** of type integer that holds the trainee's age. It should not be possible to assign a value less than 18 to the age of a trainee (in case the value is lower than 18, assign 18 to it);
- A private list **myCourses** that holds the list of courses a trainee has registered for.

b) Add a constructor with a parameter for the trainee's name (of type string) and a parameter for his/her age (of type integer). The constructor should store the values for the parameters on the corresponding fields and create an empty list of courses. Be aware, in case the value for the age parameter is less than 18, it should be set to 18.

c) Add a method **GetAllCourses**, without parameters, which returns the list of courses that the trainee registered for.

d) Add a method **RegisterForCourse** with two parameters: a course code of type string and a level of type **CourseLevel**. This method returns a Boolean. It is not allowed to have more than one course in the list **myCourses** with the same code (every course has a unique code). This method needs to check if there is already a course in the list **myCourses** with the same code. If not, a new **Course** object is created, added to the list **myCourses** and the return value is true. If there is already a course with that code, nothing is added and the return value is false.

e) Add a method **ChangeCourse** with two parameters: a course code and a level. The method returns a Boolean. The method will check if a course with the supplied course code already exists, and in that case, it will change the level of the course according to the value of the supplied parameter and it will return true. If such a course is not present in the course list, then the return value is false.

f) Add a method **GetAllBeginnerCourses,** without parameters, which will return a list of beginner courses (courses with level BEGINNER) that the trainee registered for.

Assignment 3: Main Form (2 + 4 + 12 + 12 pts)

a) In the screenshot some pages ago you see in the title bar of this application the text "Created by P. Mukala : : Details for Tom". After startup of the application, the text in the title must be "Created by XXX" with XXX replaced by your name.

This application is about 1 trainee. Add a variable to the main form of type Trainee. After startup of the application this variable does not hold a trainee yet.

b) In the "Trainee details" section the user can fill in a name and age. Clicking the "Create trainee" button should result in:

- Creating a Trainee-object and storing it in the variable you declared in the assignment 3a.
- Change the text in the title-bar in "Created by XXX : : Details for YYY", where XXX is your name and YYY is the trainee's name.

Be aware that the trainee's age should be at least 18 (in the screenshot you see 15 as its age, so it should become 18 in the trainee's-object).

Give the right code for the event-handler of the button with text "Create trainee".

c) In the "Trainee details" section the user can choose a course-name and a level. As you can see, all course-codes are unique. There are two buttons in this section.

- Clicking the button with text "Add course" results in checking if the trainee has already a course in its list with the selected code.
  If so, an appropriate message must be shown. If not, a course with that selected code and level must be created and added to the trainee's list.
- Clicking the button with text "Change course" results in checking if the trainee has already a course in its list with the selected code. If so, its level must be changed in the selected level. If not, an appropriate message must be shown.

Give an implementation of the event-handlers for both buttons.

d) In the “Details of selected courses” section there are three buttons.

- Clicking the button with text “Show all courses” results in showing information of all courses as selected by the trainee, in the listbox (for example: see the screenshot above: Tom has selected the courses PCS2, EDB2 and MATH).
- Clicking the button with text “Show beginner courses” results in showing only information of all beginner courses as selected by the trainee, in the listbox (for example: see the screenshot above: the courses EDB2 and MATH should be shown in the listbox).
- Clicking the button with text “Show billing details” results in showing details in yellow-colored groupbox at the bottom of the form with text “Billing details”. You need to show the trainee’s name and age, the number of selected courses and the total price for all selected courses together.

Give an implementation of the event-handlers for these buttons.

Veel Succes!!!

Do not forget to save and close your project before uploading!!!