CSCI 1016 Assignment 4

Q1: Textbook Page 280, Chapter 4, Programming Challenges **5, Software Sales** [30 points]
Note: For this question, name your project `<my name> Software Sales`. For example, my project would be named: `Leong Lee Software Sales`.
The form's title bar should read `<my name> Software Sales`.
Important: If you do not put `<my name>` to the above mentioned fields (project name and form title bar), **you will get 0 point for the question.**

Estimated time: 3 hours

Q2: Textbook Page 281, Chapter 4, Programming Challenges **7, Shipping Charges** [40 points]
Note: For this question, name your project `<my name> Shipping Charges`.
The form's title bar should read `<my name> Shipping Charges`.
Take note of question requirement of input validation.
Important: If you do not put `<my name>` to the above mentioned fields (project name and form title bar), **you will get 0 point for the question.**

Estimated time: 4 hours

Q3: Textbook Page 283, Chapter 4, Programming Challenges **17, Time Calculator** [30 points]
Note: For this question, name your project `<my name> Time Calculator`.
The form's title bar should read `<my name> Time Calculator`.
Important: If you do not put `<my name>` to the above mentioned fields (project name and form title bar), **you will get 0 point for the question.**

Estimated time: 3 hours

Q Extra: Textbook Page 282, Chapter 4, Programming Challenges **11, ISP** [10 extra points]
Note: For this question, name your project `<my name> ISP`.
The form's title bar should read `<my name> ISP`.

Estimated time: 4 hours

Note: For all questions, no two students should produce exactly the same code (points would be deducted from both parties).

Submission instructions:
Zip your (entire) three solution/project folders into a single zip file **lastname_firstname_assignment04.zip.**
Submit your file **lastname_firstname_assignment04.zip** to the D2L Dropbox.

Grading guidelines (programming questions):
Your programs will be judged on several criteria, which are shown below.

- Correctness (50%): Does the program compile correctly? Does the program do what it's supposed to do?
- Design (20%): Are operations broken down in a reasonable way (e.g. classes and methods)?
- Style (10%): Is the program indented properly? Do variables have meaningful names?
- Robustness (10%): Does the program handle erroneous or unexpected input gracefully?
- Documentation (10%): Do all program files begin with a comment that identifies the author, the contents, and the compiler used for that particular file? Are all the classes, methods and data fields clearly documented? Are unclear parts of code documented? (Some items mentioned may not apply to some languages)

A program that does not compile will get at most **50% of the possible points**