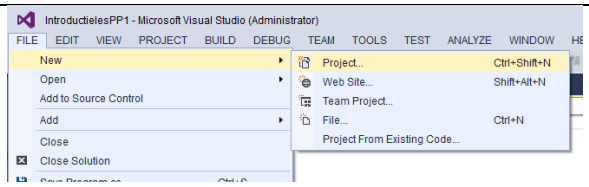
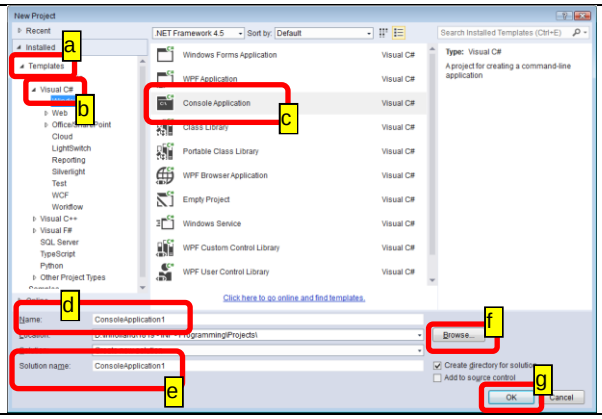
Exercise 0 - Visual studio and C#

1. Open Visual Studio

2. Click in the menubar on:
   a. *File*
   b. *New*
   c. *Project*



3. Choose the following
   a. *Templates*
   b. *Visual C#*
   c. *Console Application*
   d. At Name enter '**Exercise1**' and
   e. At Solution name '**IntroductionPP1**'
   f. Choose a suitable location
   g. Click on 'OK'



Exercise 1 - input/output

In this exercise we will create a simple program that has some input and output.

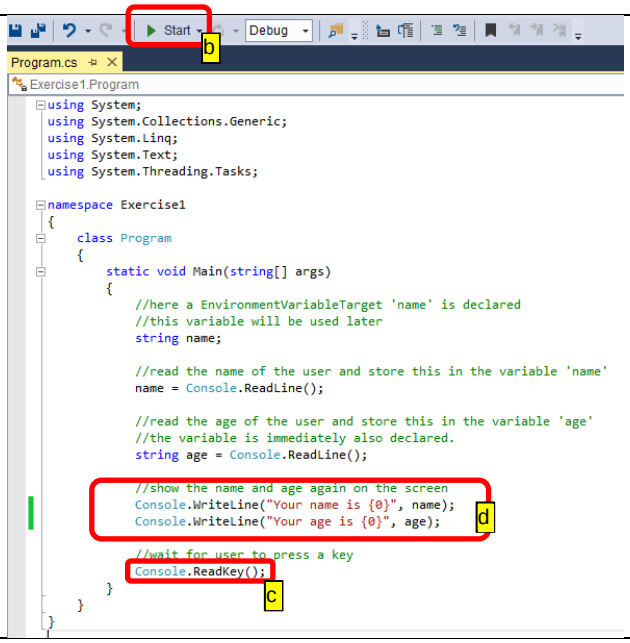
a) Enter the code as shown.

b) Click at the top on 

c) What happens if you change the last line to 'Console.ReadLine()'?

d) What happens if you change each of the 'Console.WriteLine(…)' in 'Console.Write(…)'?

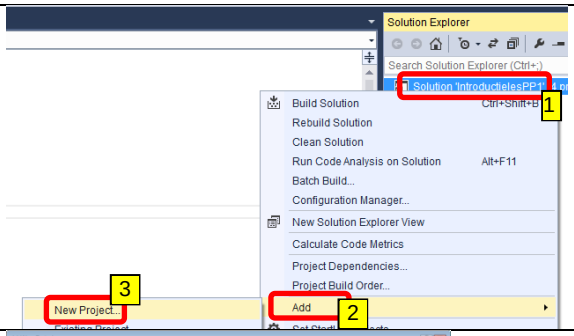
e) Adjust the program to make it more user friendly, by adding the text "Enter your name:" and "Enter your age:" prior to when the user has to enter them. Use 'Console.Write(…)' or 'Console.WriteLine(…)'.



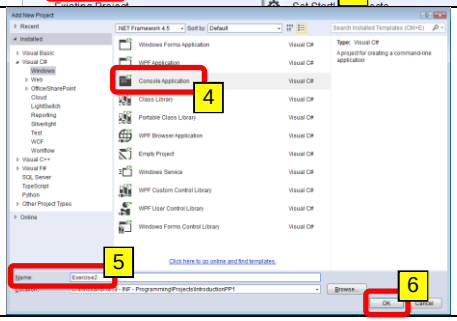
Exercise 2 - conversion

In this exercise we take a look at converting the data type of one variable into another data type, also known as a conversion.

1. **RIGHT**-click op 'Solution '**IntroductionPP1**'
2. Click on *Add*
3. Click on *New Project*

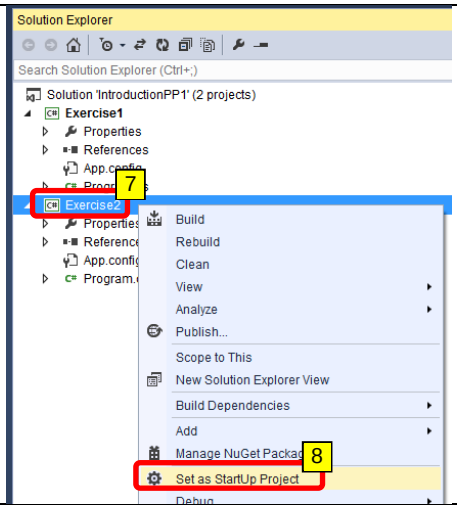


4. Choose again for 'Console Application'.
5. Enter as *Name* : **Exercise2**
6. Click on **OK**



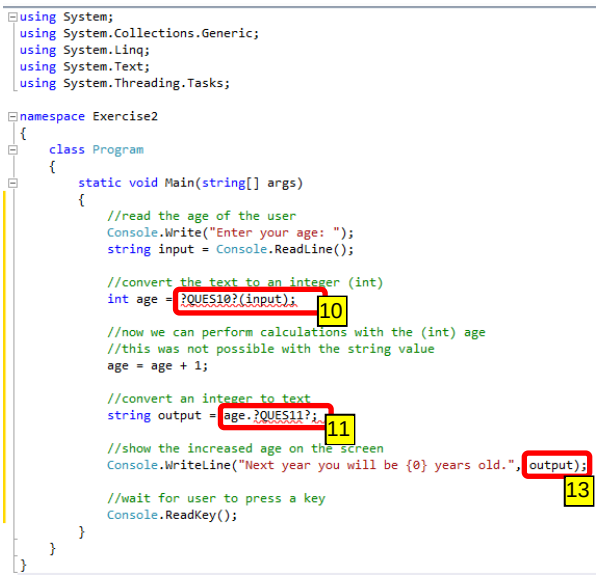
The new project will not do anything yet, since Exercise1 is still the 'start project'. We will have to change this:

7. **RIGHT**-click on **Exercise2**
8. Click on *Set as StartUp Project*



To convert a string to an int, we use **int.Parse(***variable***)**.
To convert an int to a string, we use *variable***.ToString()**.

9. Enter the code shown to the right
10. Convert the string value (*input*) to an int-value (*age*) where it has ?QUES10?
11. Convert the int value (*age*) to a string value (*output*) where it has ?QUES11?
12. Run the program and check if it works.
13. Replace the variable *output* with *age* at the Console.WriteLine(…). Does this still work?
14. To increase the age with 1 you can also use +=1, or age++. Test both.
15. Try out changing some other operators to change the age (e.g. -, *, /).



Exercise 3 - debugging

In this exercise we take a look at debugging a program. A debugger helps to track down errors in the program. With a debugger we can run the program step-by-step and see how values of variables change.

1. Add a new project tot he solution IntroductionPP1, again a Console Application, and use as a project name '**Exercise3**'
2. Enter the code shown below:

```
static void Main(string[] args)
{
    int score = 200;
    int level = 3;
    int nItems = 4;

    int score_new, level_new;

    // increase score and level
    score_new = score++;
    level_new = ++level;

    Console.WriteLine("Score (old): {0}, score (new): {1}", score, score_new);
    Console.WriteLine("level (old): {0}, level (new): {1}", level, level_new);

    nItems += 5;
    Console.WriteLine("number of items: {0}", nItems);

    // wait for user to press a key
    Console.ReadKey();
}
```

3. Run the program (F5) and look at the result. What do you notice?

4. Add a breakpoint (F9) on the line score_new = score++; and run the program again(F5).

5. The program now stops at the selected line. With F10 you can run a single line/statement. If you hold the mouse pointer on a variable the value of it will appear.

6. What is the difference between var++ and ++var?

7. Adjust the code so the new score and the new level will be increased, but the old score and old level do not change.

Exercise 4 - boolean

In the previous exercises we used variables of the type int and string. Int-variables can be integeres, string-variables can contain text. Another data type in C# is a bool. Bool variables can either be true or false false.

1. Add a new project tot he solution IntroductionPP1, again a Console Application, and use as a project name '**Exercise4**'

2. Enter the code shown below:

```
static void Main(string[] args)
{
    bool answer;
    answer = (5 != 3);
    Console.WriteLine("boolean answer: {0}", answer);

    // wait for user to press a key
    Console.ReadKey();

}
```

3. What will the output of this program show (without running the program)?

4. Run the program (F5)
   Note that the result of (5!=3) can be stored in a Boolean variable.

5. Adjust the code so the user can enter his/her age, and then store this in a integer variable *age*.

6. Create a new bool-variable *tooOld* .

7. Assign the appropriate value tot his new variable, depending on the age entered by the user. An age of 65 or higher is too old. Show the result.

Exercise 5 – Mathematics

On the exam of Mathematics one of the first questions is often to add four positive integers between 100 and 10000. In C# you can generate a random number using:

```
Random rnd = new Random();
```

And then declare a random variable using for example

```
int number1 = rnd.Next(101, 10000);
```

- Write a program that will show four numbers between 101 and 10000, then asks to add these up, and checks the result.