**CS 132 - C# Beginner**

**HOS04 – Methods and Scope**

School of Technology & Computing (STC) @ City University of Seattle (CityU)

10/17/2024 Reviewed by Shahid Khan

01/23/2025 Reviewed by Megha Narendra Simha

03/19/2025 Reviewed by Megha Narendra Simha

A logo of a cat

Description automatically generated

**Before You Start**

1. **Screenshots may be different from your environment.**
2. The directory path shown in screenshots may be different from yours.
3. Version numbers may not match the most current version at the time of writing. If given the option to choose between the stable release (long-term support) or the most recent, please select the **stable release** rather than the beta-testing version.
4. The steps might have subtle discrepancies. Please use your best judgment while following this cookbook-style tutorial to complete each step.
5. All the steps and concepts in this tutorial are from references, so if you encounter problems, please **try to read and compare the references to solve the problem**. If you still can't solve the problem, don't hesitate to contact your course Student Assistant.
6. **Avoid copying code from the book or the GitHub repository. Instead, type out the code yourself. Re-resort to copying** only when you are stuck and find things not working as expected.
7. Some steps may not be explained in detail. If you are not sure what to do:
8. Consult the resources from the course.

If you cannot solve the problem after a few tries (usually 15 -30 minutes), ask a Student Assistant for help.

**Learning Outcomes**

Students will be able to:

* Understand, define, and call methods
* Understand, explain, and call parameters and return types
* Understand and differentiate scopes of variables.

**Resources**

* C# Tutorials | W3Schools.com- <https://www.w3schools.com/cs/default.asp>
* C# Tutorials | tutorials.com- [https://www.tutorialspoint.com/csharp/](https://www.tutorialspoint.com/csharp/csharp_strings.htm)
* Scope of Variables in C# | geeksforgeeks.org - <https://www.geeksforgeeks.org/scope-of-variables-in-c-sharp/>
* Understanding Classes and Objects the C# Way| informit.com - <http://www.informit.com/articles/article.aspx?p=1609145&seqNum=4>

**Methods:**

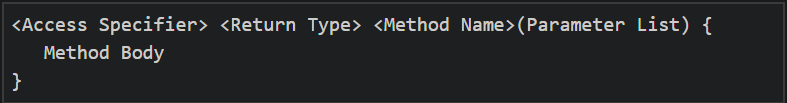
A method is a block of code or a group of statements that together perform a task when it is called. Every C# program has at least one class with a method named Main.

Methods are used to perform specific actions, also known as functions.

You can pass data, known as parameters, into a method.

A method is defined with the technique's name, followed by parentheses (). C# provides some pre-defined methods you are already familiar with, such as Main(), but you can also create your own methods to perform specific actions.

When you define a method, you declare the elements of its structure. The syntax for explaining a process in C# is as follows:



The following are the various elements of a method:

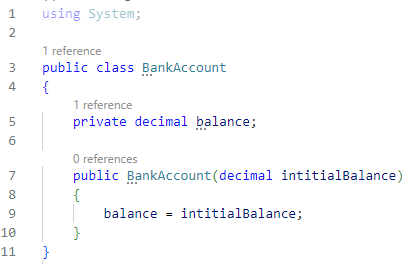
* **Access Specifier:** This determines the visibility of a variable or a method from another class.
* **Return type:** A method may return a value. The return type is the data type of the value it returns. If the method does not return any values, then the return type is void.
* **Method name:** The method name is a unique, case-sensitive identifier that cannot be the same as any other identifier declared in the class.
* **Parameter list:** Enclosed between parentheses, the parameters are used to pass and receive data from a method. The parameter list refers to the type, order, and number of the parameters of a process. Parameters are optional; that is, a method may contain no parameters.
* **Method body:** This contains instructions for completing the required activity.

**Scope**

The part of the program where a particular variable is accessible is termed the Scope of that variable. A variable can be defined in a class, method, loop, etc. In C#, a scope is determined by the curly braces "{}." If a variable is declared inside a method, the variable has method-level Scope and is inaccessible outside the process. Those variables are called "local variables."

**Create a project**

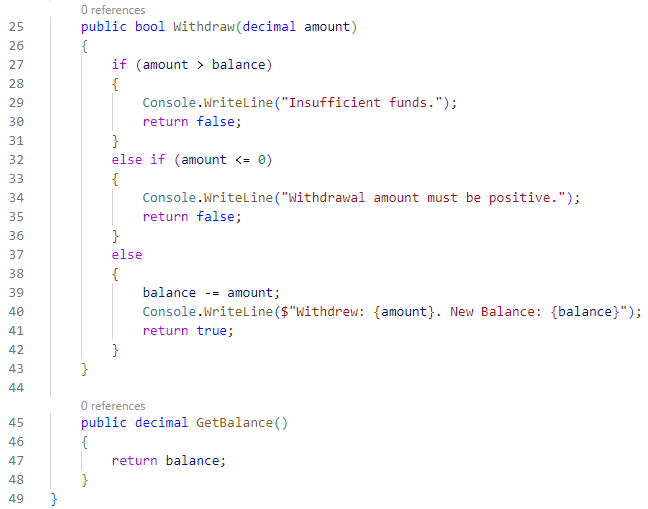
1. Follow instructions from HOS1 to use Codespaces or any code editors of your choice (Visual Studio, Visual Studio Code, Sublime, Vim, etc.…)
2. We will create a program for Bank Accounts in C # to illustrate the concept of methods and scopes.
3. Create a BankAccount class and initialize a private variable balance with the double to prevent access outside the class. Create a constructor to initialize a new BankAccount object with a starting balance.
4. Create a new Class BankAccount.cs and add the code



1. Create a function to deposit money into a bank account, first checking if the deposit is positive.



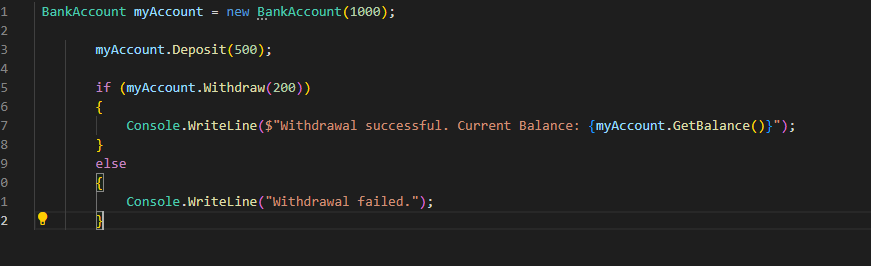
1. Create a function for withdrawing money and checking the current balance. Make sure the withdrawal money is positive and has a sufficient balance.



1. Create a primary function to test all the methods: initial balance with 1000, deposit 500, and withdraw 200.

In the Program.cs enter the code below.

It may look like this



Or like this based on .Net version.

A screenshot of a computer code

Description automatically generated

1. After executing the code, expect the output:

**A close up of a text

Description automatically generated**

**Submit your Work to GitHub**

Please upload all your files in a zip file for this hands-on skill to the HOS assignment.

Right-click and download the “Program. cs” file. And then submit it to GitHub.

**>> git add .**

**>> git commit -m “HOS04 – your name”**

**>> git push**

Follow the instructions on the CityU STC TA Center Github.io [Submit your work page.](https://cityuseattle.github.io/docs/hoporhos/submit/)