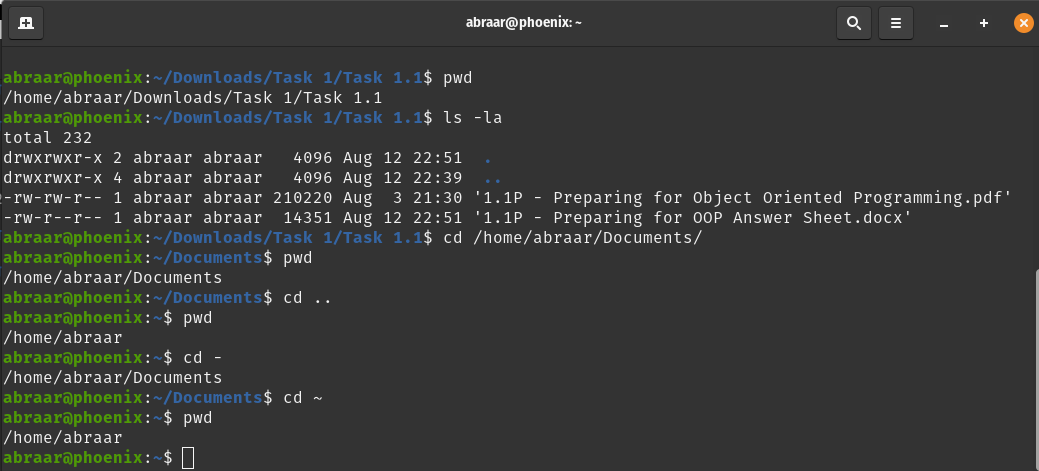
1.1P: Preparing for OOP – Answer Sheet

1. Explain the following terminal instructions:
   1. **cd:** This command is “change directory” / “cd”. This  command is used to change the current working directory. Example :

|  |  |
| --- | --- |
| Command | What it does |
| 1. cd “Absolute or Relative path” | Changes the path to specified path. |
| 2. cd .. | Navigates to parent directory. |
| 3. cd - | Navigate to previous directory. |
| cd ~ | Navigate to home directory. |

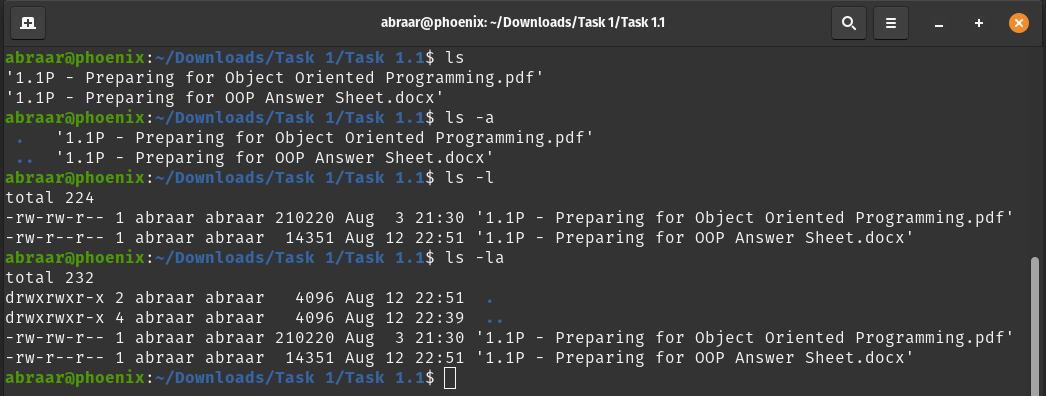
**Screenshot :**



* 1. **ls:** This command shows all the files and directories in the working directory. Example :

|  |  |
| --- | --- |
| Command | What it does |
| 1. ls | Lists files and dirs. |
| 2. ls -a | Doesn’t hide files starting with . |
| 3. ls -l | Uses long formatting. |

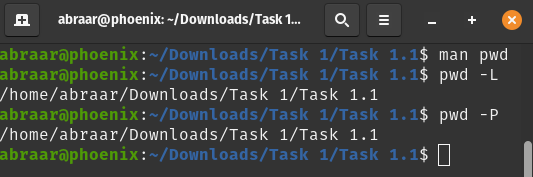
**Screenshot :**

****

* 1. **pwd:** Prints the name of current/working directory. Example :

|  |  |
| --- | --- |
| Command | What it does |
| 1. pwd | prints the current working directory path from / |
| 2. pwd -L | L : Logical. Outputs the $PWD [environment variable](https://phoenixnap.com/kb/linux-set-environment-variable) contents, including symbolic links. |
| 3. pwd -P | P : Physical. All the components are directory names, and symbolic links are resolved. |

**Screenshot :**

****

1. Consider the following kinds of information, and suggest the most appropriate data type to store or represent each:

|  |  |
| --- | --- |
| **Information** | **Suggested Data Type** |
| A person’s name | string |
| A person’s age in years | int |
| A phone number | string |
| A temperature in Celsius | float/double/decimal |
| The average age of a group of people | float |
| Whether a person has eaten lunch | bool |

1. Aside from the examples already provided in question 2, come up with an example of information that could be stored as:

|  |  |
| --- | --- |
| **Data type** | **Suggested Information** |
| String | Any text, ex : Name of course I have to take in this semester. |
| Integer | Number of classes I have to take in this semester. |
| Float | CGPA of last semester. |
| Boolean | Whether I have taken a class or not. |

1. Fill out the following table, evaluating the value of each expression and identifying the data type the value is most likely to be:

|  |  |  |  |
| --- | --- | --- | --- |
| **Expression** | **Given** | **Value** | **Data Type** |
| 6 |  | 6 | int |
| True |  | True | bool |
| a | a = 2.5 | 2.5 | double |
| 1 + 2 \* 3 |  | 7 | int |
| a and False | a = True | False | bool |
| a or False | a = True | True | bool |
| a + b | a = 1  b = 2 | 3 | int |
| 2 \* a | a = 3 | 6 | int |
| a \* 2 + b | a = 2.5 b = 2 | 7 | double |
| a + 2 \* b | a = 2.5  b = 2 | 6.5 | double |
| (a + b) \* c | a = 1  b = 1  c = 5 | 10 | int |
| “Fred” + “ Smith” |  | Fred Smith | String |
| a + “ Smith” | a = “Wilma” | Wilma Smith | String |

1. Using an example, explain the difference between **declaring** and **initialising** a variable.

***Difference between initialisation and declaration :***

* Initialisation: Giving an initial value to a variable.
* Declaration: of a variable is informing the compiler with the variable name, the type of value the variable holds, and the initial value

Example :

int a ; // a is *declared but not initialised.*

int b = 5; // *declared and initialised.*

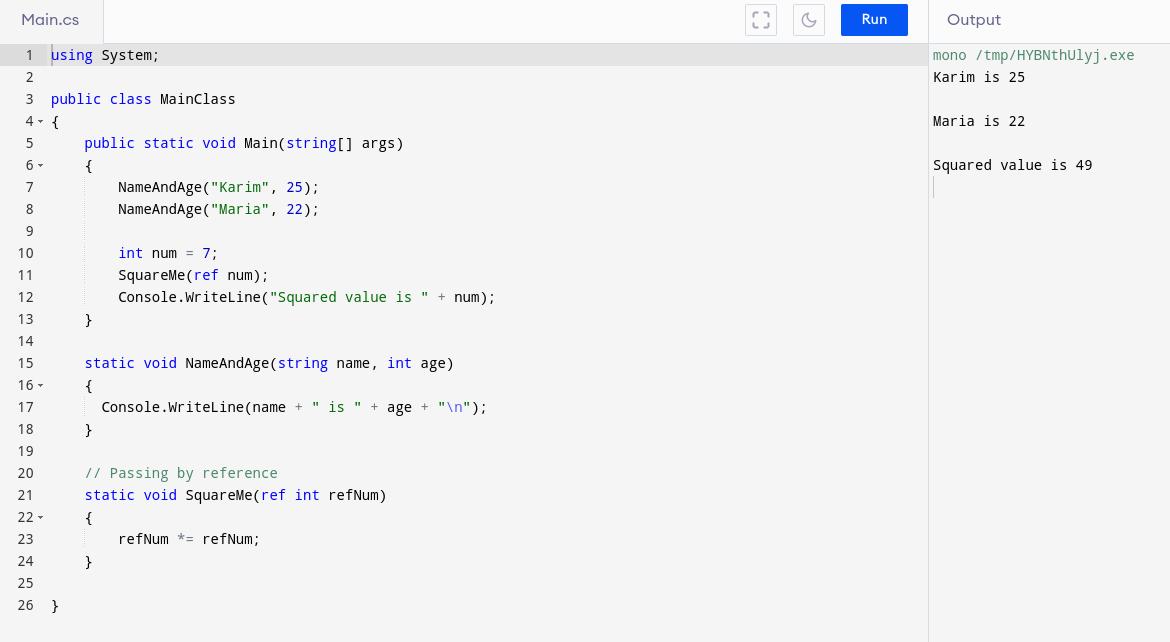
We can declare and initialise on two lines.

int x; // declares integer variable

x = 10; // initialises integer variable that's already been declared.

1. Explain the term **parameter**. Write some code that demonstrates a simple of use of a parameter. You should show a procedure or function that uses a parameter, and how you would call that procedure or function.

A **parameter** is an argument passed to a method in C#. Parameters can be passed by value or by reference. Here’s an example that shows the use of parameters:

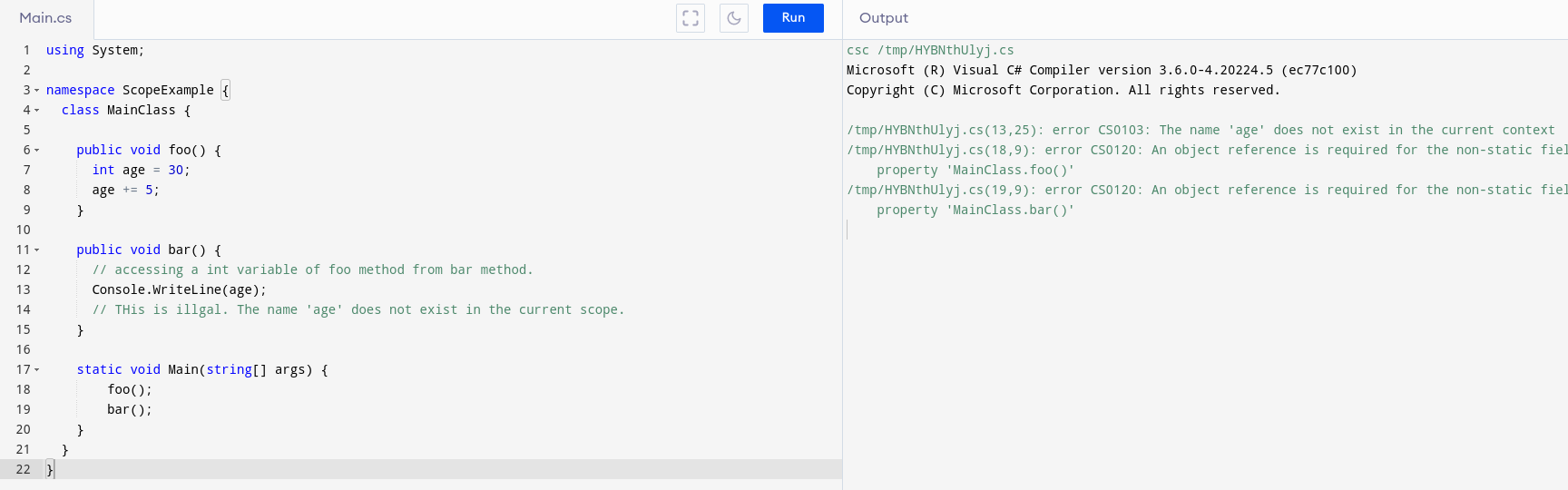


1. Using an example, describe the term **scope**.

**Scope** of a variable is the part of the program where a particular variable is accessible. C# has 3 layers of scope :

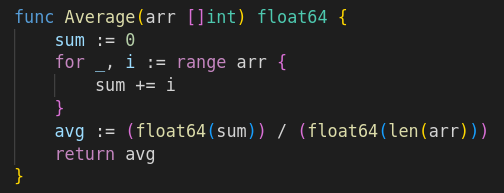
* Class Level;
* Method Level;
* Block Level;

Here’s an example of method level scope of an integer:

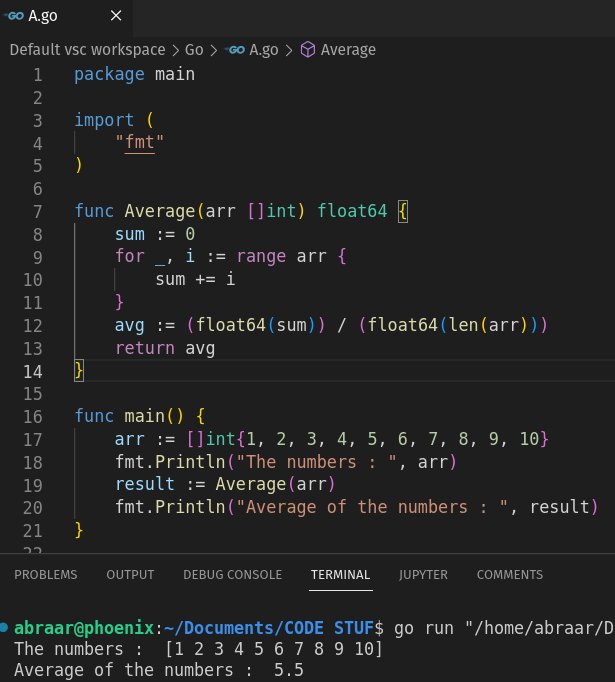


1. In any procedural language you like, write a function called Average, which accepts an array of integers and returns the average of those integers. Do not use any libraries for calculating the average. You must demonstrate appropriate use of parameters, returning and assigning values, and use of a loop. Note — just write the function at this point, we’ll *use* it in the next task. You shouldn’t have a complete program or even code that outputs anything yet at the end of this question.

Here’s a go function that returns the average of an array of integers:

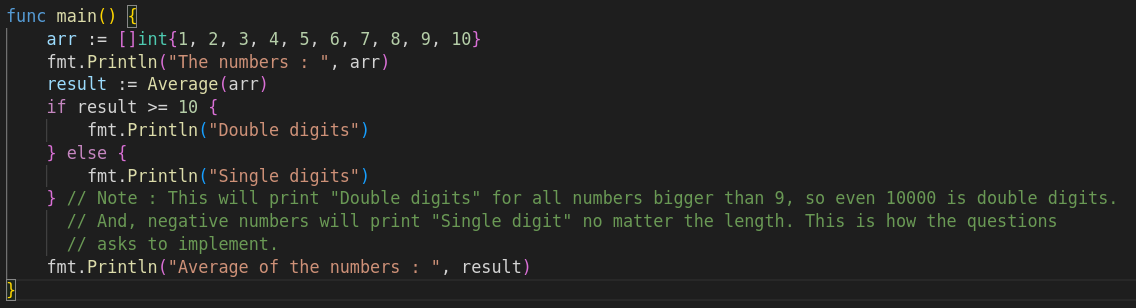


1. In the same language, write the code you would need to call that function and print out the result.



1. To the code from 9, add code to print the message “Double digits” if the average is above or equal to 10. Otherwise, print the message “Single digits”. Provide a screenshot of your program running.

*<insert a screenshot of your code here>*

 *<insert a screenshot of your whole program running here>*

