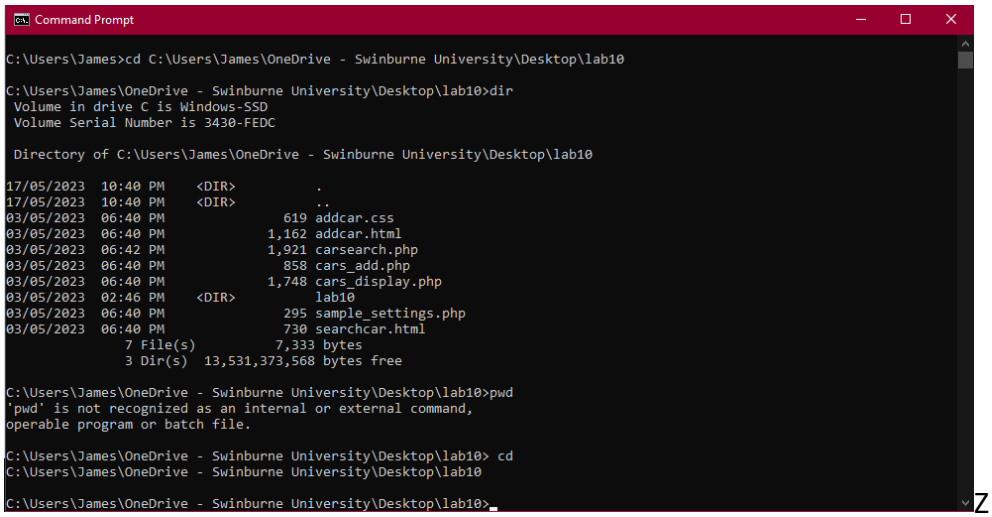
1.1P: Preparing for OOP – Answer Sheet

1. Explain the following terminal instructions:
    a. cd: Change directory.
    b. ls: list files (but it's a linx command for windows is dir)
    c. pwd: Print directory (for windows its just cd without passing any arguments)


Z

2. 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	Integer
A phone number	String or integer
A temperature in Celsius	Float
The average age of a group of people	Integer
Whether a person has eaten lunch	Boolean

3. Aside from the examples already given, come up with an example of information that could be stored as:

Data type	**Suggested Information**
String	A locations name
Integer	Postcode
Float	Height in cm
Boolean	Is ‘A’ the first letter in the alphabet

4. 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**
`5`		5	Integer
`True`		True	Boolean
`a`	`a = 2.5`	2.5	float
`1 + 2 * 3`		7	integer
`a and False`	`a = True`	True	Boolean
`a or False`	`a = True`	True	Boolean
`a + b`	`a = 1` `b = 2`	3	integer
`2 * a`	`a = 3`	6	integer
`a * 2 + b`	`a = 1.5` `b = 2`	5	integer
`a + 2 * b`	`a = 1.5` `b = 2`	7	integer

`(a + b) * c`	`a = 1`
	`b = 1`
	`c = 5`
		5.5	float

`“Fred” + “ Smith”`		`“Fred Smith”`	string
`a + “ Smith”`	`a = “Wilma”`	“Wilma Smith”	string

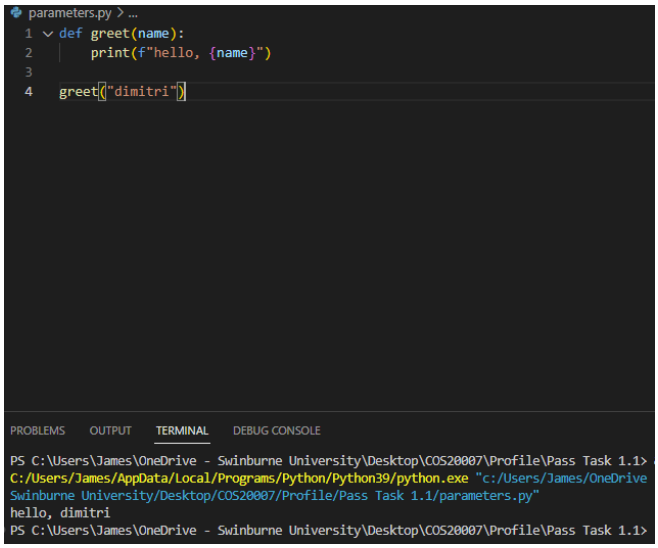
5. Explain the difference between **declaring** and **initialising** a variable.

The difference between the two is declaring is used to specify the type of data for the variable, while initialising is used to set initial value to the variable.

```
int x; // declaring a variable
int x = 10; // decalre and initialising a variable
```

6. Explain the term **parameter**. Write some code that demonstrates a simple of use of a parameter.

A parameter is variable inside a function and it becomes an argument after passing to the function



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

Scope is where variables can be visible and accessible.

Example:

```
def my_function():
    x = 10
    print(x)

my_function()
```

8. In any procedural language you like, write a function called Average, which accepts an array of integers and returns the average of those integers. 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 task.

9.

```python
def average():
    numbers = []
    while True:
        number = int(input("Enter a number: "))
        numbers.append(number)
        user_input = input("Do you want to enter another number? Yes/No: ")
        if user_input.lower() == 'no':
            break

average()
```

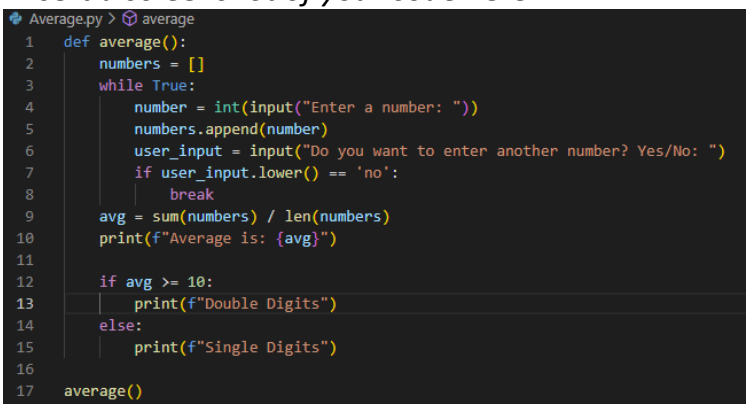
In the same language, write the code you would need to call that function and print out the result.

```python
def average():
    numbers = []
    while True:
        number = int(input("Enter a number: "))
        numbers.append(number)
        user_input = input("Do you want to enter another number? Yes/No: ")
        if user_input.lower() == 'no':
            break
    avg = sum(numbers) / len(numbers)
    print(f"Average is: {avg}")

average()
```

10. To the code from 9, add code to print the message “Double digits” if the average is above 10. Otherwise, print the message “Single digits”.

<insert a screenshot of your code here>



<insert a screenshot of your whole program running here>