# INDEX

Contents

[INDEX 0](#_Toc33633626)

[List of Figures 1](#_Toc33633627)

[List of Tables 1](#_Toc33633628)

[Practical 1: Report on 2](#_Toc33633629)

[LIFESKILL WORKSHOP CUM BOOTCAMP. 2](#_Toc33633630)

[Day 1 2](#_Toc33633631)

[Session 1 2](#_Toc33633632)

[Session 2 2](#_Toc33633633)

[Day 2 2](#_Toc33633634)

[Session 1 2](#_Toc33633635)

[Session 2 3](#_Toc33633636)

[Day 3 3](#_Toc33633637)

[Session 1 3](#_Toc33633638)

[Session 2 3](#_Toc33633639)

[Practical 2: Green Computing 4](#_Toc33633640)

[Green Sustainability: 4](#_Toc33633641)

[Green Technology Types: 4](#_Toc33633642)

[ Green use: 4](#_Toc33633643)

[ Green disposal: 5](#_Toc33633644)

[ Green design: 5](#_Toc33633645)

[ Green manufacturing: 5](#_Toc33633646)

[Practical 3: Free software and Open Source Software 7](#_Toc33633647)

[What is Free Software? 7](#_Toc33633648)

[What is open source software? 8](#_Toc33633649)

[Control. 8](#_Toc33633650)

[Training 8](#_Toc33633651)

[Security. 8](#_Toc33633652)

[Stability 9](#_Toc33633653)

[Free redistribution of the software. 9](#_Toc33633654)

[Freeware 9](#_Toc33633655)

[Difference between Free Software and Open Source Software 9](#_Toc33633656)

[Practical 4: PEP-8 11](#_Toc33633657)

[Standard way of writing a Program 11](#_Toc33633658)

[Rules 11](#_Toc33633659)

[Indentation: 11](#_Toc33633660)

[Maximum Line Length: 11](#_Toc33633661)

[Line Break Before and After Operators: 11](#_Toc33633662)

[Blank Lines 11](#_Toc33633663)

[Imports 11](#_Toc33633664)

[String Quotes 11](#_Toc33633665)

[Whitespace in Expressions and Statements 11](#_Toc33633666)

[Comments: 12](#_Toc33633667)

[Class Names 12](#_Toc33633668)

[Exception Names 12](#_Toc33633669)

[Function and Variable Names 12](#_Toc33633670)

[Names to Avoid 12](#_Toc33633671)

[Book program 12](#_Toc33633672)

[GUI Program 14](#_Toc33633673)

[Practical 5: GitHub 15](#_Toc33633674)

[Steps to create a GitHub repository: 15](#_Toc33633675)

[Practical: 6 17](#_Toc33633676)

[BLOG: FOOD 17](#_Toc33633677)

# List of Figures

[Figure 1 : Green Computing 4](#_Toc33633678)

[Figure 2: Green Logo 5](file:///C:\Users\JOHANUS%20MENDES\Downloads\final%20it%20tools.docx#_Toc33633679)

[Figure 3 : Open Source Software 8](#_Toc33633680)

[Figure 4 Repository account 16](#_Toc33633681)

[Figure 5 blog food 17](#_Toc33633682)

# List of Tables

[Table 1: Difference between Freeware and Open Source Software 11](#_Toc32661748)

# Practical 1: Report on

# LIFESKILL WORKSHOP CUM BOOTCAMP.

Date: 28th November 2019

Venue: Nagindas Khandwala College Auditorium, Bhavishya Bharat campus, Malad (w)

The workshop was arranged for three days starting from 25th November 2019 to 27th November 2019. It was a life skill based workshop for students of BSc. IT and BSc. CS students of NK College. The workshop was free of cost.

Each day was divided into two sessions. Students are supposed to do various activities for life skill enhancement.

## Day 1

### Session 1

The students were given a warm welcome, fully energetic and excited. The session started with a short speech delivered by Principle Dr. ANCY JOSE and gave importance of life skills. Our felicitator Dr. SUJATA SINGHI then carried the session. The students enjoyed dancing on the song FOLLOW THE LEADER.

### Session 2

The session two of the first day was all about developing marketing skills. All the students were divided into groups of 9 and given a task of trying to sell a product which can be anything to judges. Dr. Sujata taught about gratitude and importance of expressing feelings.

## Day 2

### Session 1

Session one on day 2 was about sharing the experience of expressing love to parents by the students. The session was quit emotional. Each students were taught the importance of their feelings. The students enjoyed dancing on the song FOLLOW THE LEADER. . Dr. Sujata gave task of making a spaceship and working with team.

### Session 2

The ROCKSTAR concert was organized to energized the students and make them comfortable in that environment. Stage fear of many students was eliminated and they were confident.

## Day 3

### Session 1

The students were charged by dancing on the song Follow the Leader at the start of session. Meaning of values and its importance of implement of values in life was explained.

***Vision with Mission*** was the proverb which was mainly focused. Students made their vision boards.

### Session 2

Students were taught the power of sound and meditation. The used to play music and make their own rhythms.

Session two for the last day which was the last session ended giving a heartfelt thanks to all the members, participants and people behind the huge success of the workshop.

# Practical 2: Green Computing

Green computing aims to attain economic viability and improve the way computing devices are used. Green IT practices include the development of environmentally sustainable production practices, energy-efficient computers and improved disposal and recycling procedures.

The environment response to human ignorance; setting off uncontrollable often-immediate inconsequential processes that eventually changes the course of our existence.

:

Figure 1 : Green Computing

## Green Sustainability:

Meeting the needs of society the ways that can continue in definitely into the future without damaging or depletion natural resources. In short, meeting present needs without compromising the ability of future generation to meet their own needs.

## Green Technology Types:

* Green use:

Minimizing the electricity consumption of computers and their peripheral devices and using them in an eco-friendly manner

* Green disposal:

 Repurposing existing equipment or appropriately disposing of, or recycling, unwanted electronic equipment

* Green design:

Designing energy-efficient computers, servers, printers, projectors and other digital devices

* Green manufacturing:

Minimizing waste during the manufacturing of computers and other subsystems to reduce the environmental impact of these activities

Government regulatory authorities also actively work to promote green computing concepts by introducing several voluntary programs and regulations for their enforcement.

Average computer users can employ the following tactics to make their computing usage greener:



Figure 2: Green Logo

Use the hibernate or sleep mode when away from a computer for extended periods.

Activate the power management features for controlling energy consumption.

Make proper arrangements for safe electronic waste disposal.

Turn off computers at the end of each day.

Refill printer cartridges, rather than buying new ones.

Instead of purchasing a new computer, try refurbishing an existing device.

# Practical 3: Free software and Open Source Software

## What is Free Software?

“Free software” means software that respects users' freedom and community. Roughly, it means that the users have the freedom to run, copy, distribute, study, change and improvethe software.

A program is free software if the program's users have the four essential freedoms:

* The freedom to run the program as you wish, for any purpose (freedom 0).
* The freedom to study how the program works, and change it so it does your computing as you wish (freedom 1). Access to the source code is a precondition for this.
* The freedom to redistribute copies so you can help others (freedom 2).
* The freedom to distribute copies of your modified versions to others (freedom 3). By doing this you can give the whole community a chance to benefit from your changes. Access to the source code is a precondition for this.

The freedom to deploy the software for any use case without any restrictions. For example, saying that the license of a program expires after 30 days makes it non-free.

The freedom to study how the software works and modify it according to your needs and preferences.

The freedom to freely re-distribute the software to assist someone in need. The redistribution can be done at a cost or at no cost.

The freedom to enhance the performance of the software and release your enhancements for the community to benefit—both programmers and non-programmers. You can do this at a cost or at no cost.

## 

## What is open source software?

Open source software is software with source code that anyone can inspect, modify, and enhance.

"Source code" is the part of software that most computer users don't ever see; it's the code computer programmers can manipulate to change how a piece of software—a "program" or "application"—works. Programmers who have access to a computer program's source code can improve that program by adding features to it or fixing parts that don't always work correctly.

Control. Many people prefer open source software because they have more control over that kind of software. They can examine the code to make sure it's not doing anything they don't want it to do, and they can change parts of it they don't like.



Figure 3 : Open Source Software

Training**.** Other people like open source software because it helps them become better programmer. Because open source code is publicly accessible, students can easily study it as they learn to make better software. Students can also share their work with others, inviting comment and critique, as they develop their skills.

Security. Some people prefer open source software because they consider it more secure and stable than proprietary software. Because anyone can view and modify open source software, someone might spot and correct errors or omissions that a program's original authors might have missed.

Stability**.** Many users prefer open source software to proprietary software for important, long-term projects. Because programmers publically distribute the source code for open source software, users relying on that software for critical tasks can be sure their tools won't disappear or fall into disrepair if their original creators stop working on them.

### Free redistribution of the software.

* The source code should be publicly available.
* The software can be modified and distributed in a different format from the original software.
* The software should not discriminate against persons or groups.
* The software should not restrict the usage of other software.

## Freeware

Typically, freeware refers to a software that you can use without incurring any costs. Unlike open source software and free software, freeware offers minimal freedom to the end user.

Whereas it can be used free of charge, often modification, redistribution, or other improvements cannot be done without getting permission from the author.

As such, freeware is often shared without including its source code, which is atypical to open source software or free software.

## Difference between Free Software and Open Source Software

The terms “free software” and “open source” stand for almost the same range of programs. However, they say deeply different things about those programs, based on different values.

The free software movement campaigns for freedom for the users of computing; it is a movement for freedom and justice. By contrast, the open source idea values mainly practical advantage and does not campaign for principles. This is why we do not agree with open source, and do not use that term.

Table 1: Difference between Freeware and Open Source Software

|  |  |
| --- | --- |
| Freeware | Open Source Software |
| You don't get source code. You only get to use this software for free. You don't get the source code and hence you can't modify it. | You get source that you can build, modify and redistribute. You however also needs to distribute your source code modifications to the world. |
| Free software is a software that you don't need to spend money to obtain and use it. But, you can't modify the software because you don't have the source code. | Open source application are developed using Open source software such PHP, MYSQL, APACHE, Linux and etc. Which means you don’t required to buy any software to run your programs. |
| Free software just means the author doesn’t charge for it. It could still be proprietary. | Open source software actually is a community project, and anyone can work on the code or even fork it to another purpose. |

# Practical 4: PEP-8

# Standard way of writing a Program

## Rules

### [Indentation](https://www.python.org/dev/peps/pep-0008/#id17):

Use 4 spaces per indentation level.

The 4-space rule is optional for continuation lines.

### [Maximum Line Length](https://www.python.org/dev/peps/pep-0008/#id19):

Limit all lines to a maximum of 79 characters.

The Python standard library is conservative and requires limiting lines to 79 characters

### Line Break Before and After Operators:

In Python code, it is permissible to break before or after a binary operator, as long as the convention is consistent locally.

### [Blank Lines](https://www.python.org/dev/peps/pep-0008/#id21)

Surround top-level function and class definitions with two blank lines.

Method definitions inside a class are surrounded by a single blank line.

### [Imports](https://www.python.org/dev/peps/pep-0008/#id23)

Imports should usually be on separate lines

### [String Quotes](https://www.python.org/dev/peps/pep-0008/#id25)

In Python, single-quoted strings and double-quoted strings are the same.

When a string contains single or double quote characters, however, use the other one to avoid backslashes in the string. It improves readability.

### [Whitespace in Expressions and Statements](https://www.python.org/dev/peps/pep-0008/#id26)

Avoid extraneous whitespace in the following situations:

* Immediately inside parentheses, brackets or braces.
* Between a trailing comma and a following close parenthesis.
* Immediately before a comma, semicolon, or colon.
* Immediately before the open parenthesis that starts the argument list of a function call.
* More than one space around an assignment (or other) operator to align it with another.
* Don't use spaces around the = sign when used to indicate a keyword argument.

### Comments:

You should use two spaces after a sentence-ending period in multi- sentence comments, except after the final sentence.

Comments should be complete sentences. The first word should be capitalized, unless it is an identifier that begins with a lower case letter (never alter the case of identifiers!).

### [Class Names](https://www.python.org/dev/peps/pep-0008/#id41)

Class names should normally use the Cap Words convention.

### [Exception Names](https://www.python.org/dev/peps/pep-0008/#id43)

Because exceptions should be classes, the class naming convention applies here. However, you should use the suffix "Error" on your exception names

### [Function and Variable Names](https://www.python.org/dev/peps/pep-0008/#id45)

Function names should be lowercase, with words separated by underscores as necessary to improve readability.

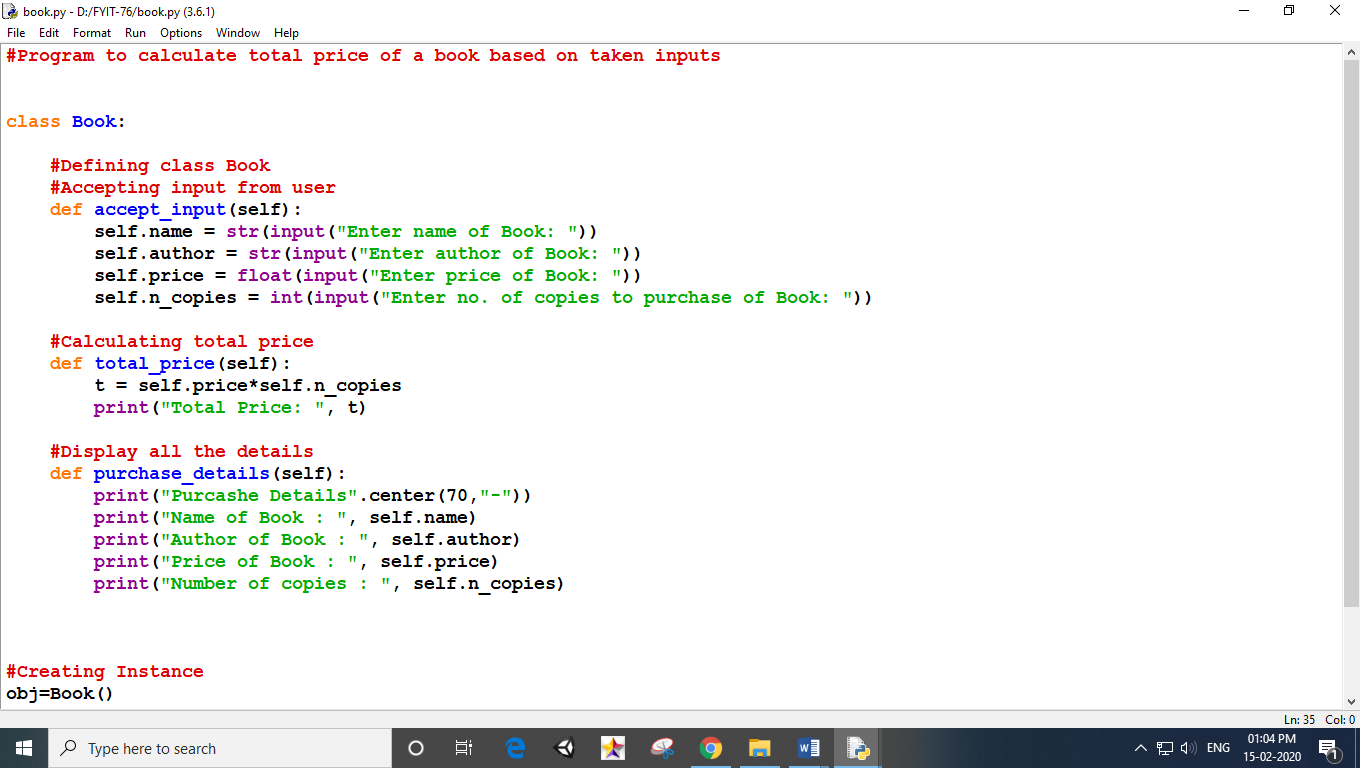
Variable names follow the same convention as function names.

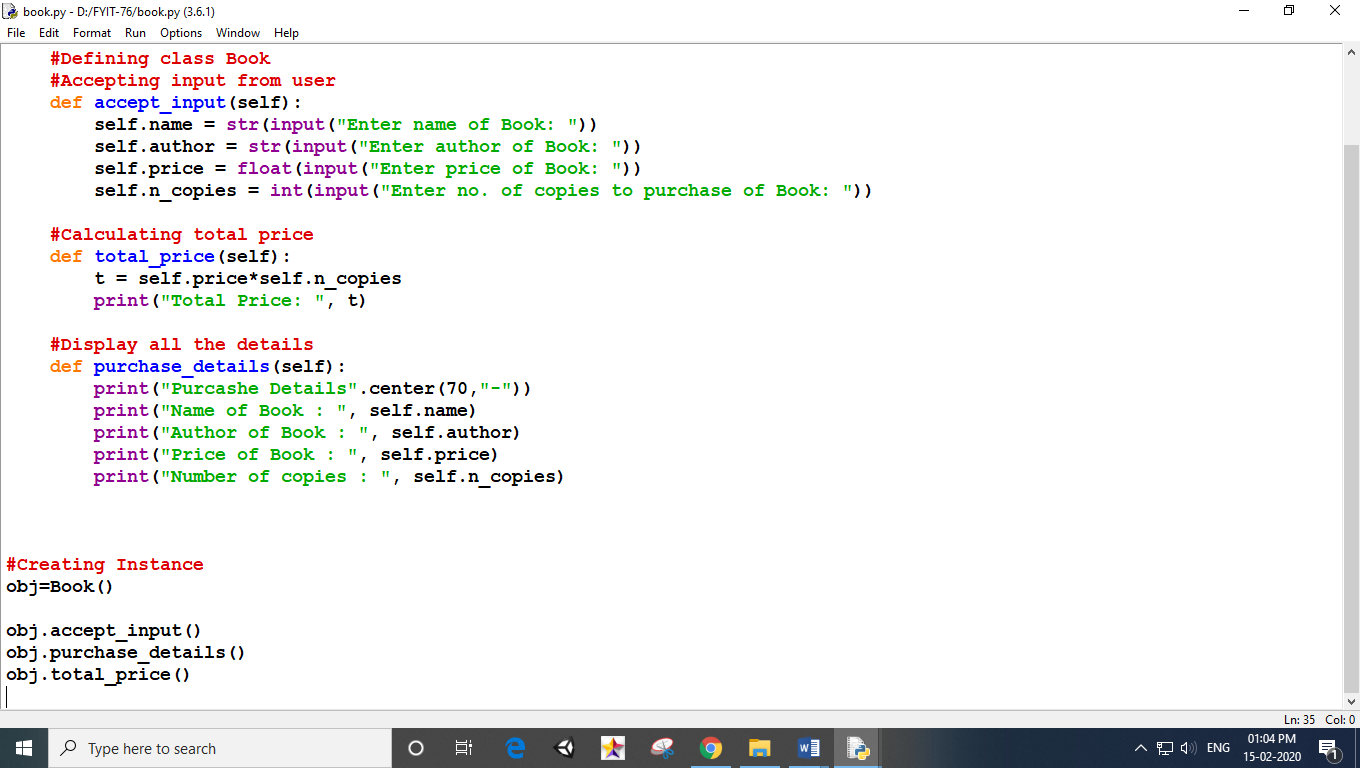
### [Names to Avoid](https://www.python.org/dev/peps/pep-0008/#id38)

Never use the characters 'l' (lowercase letter el), 'O' (uppercase letter oh), or 'I' (uppercase letter eye) as single character variable names.

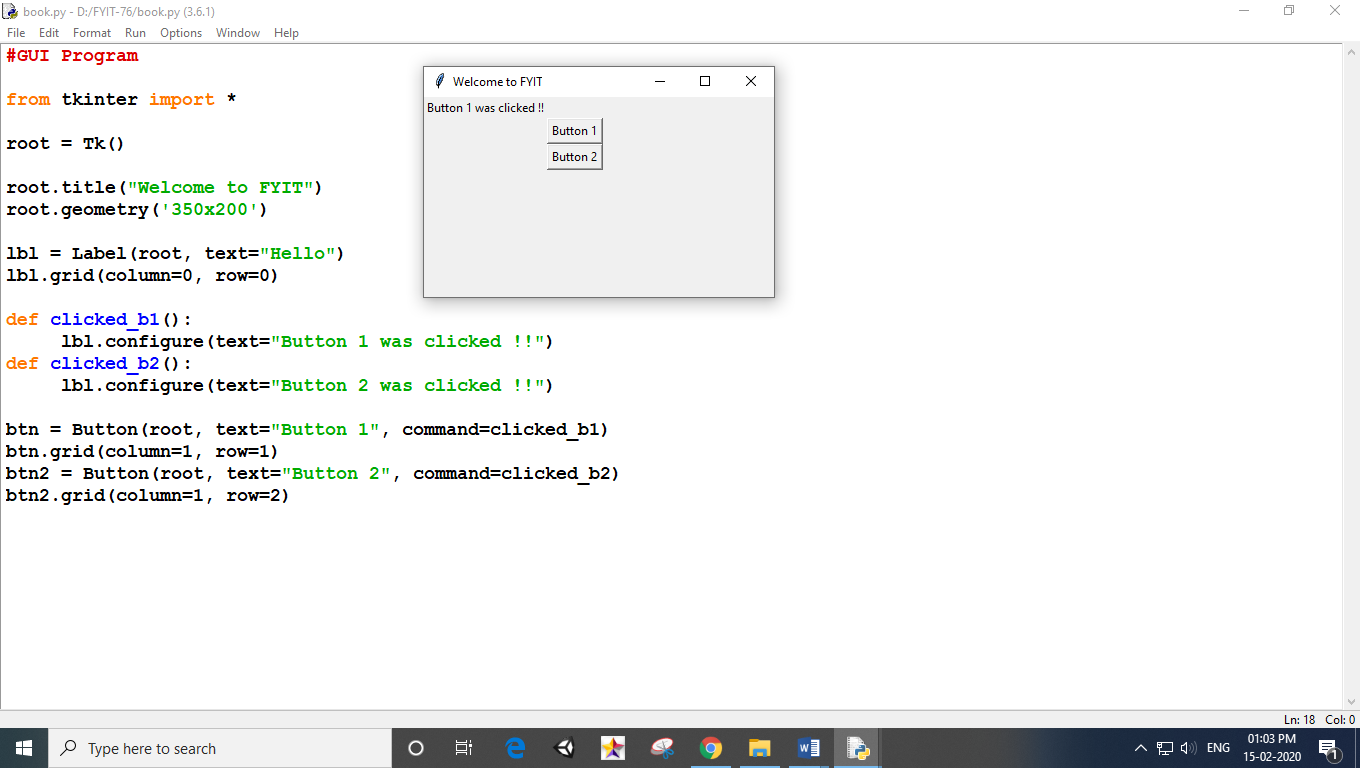
Example

### Book program





### GUI Program



# Practical 5: GitHub

## Steps to create a GitHub repository:

Step1:Go to <https://github.com/join> in a web browser. You can use any web browser on your computer, phone, or tablet to join.

Step2:

Enter your personal details. In addition to creating a username and entering an email address, you'll also have to create a password. Your password must be at least 15 characters in length or at least 8 characters with at least one number and lowercase letter.

Step3:

Click the green **Create an account** button. It's below the form.

Step4:

Complete the CAPTCHA puzzle. The instructions vary by puzzle, so just follow the on-screen instructions to confirm that you are a human.

Step5:

Click the Choose button for your desired plan. One you select a plan, GitHub will send an email confirmation message to the address you entered

Step6:

Click the Verify email address button in the message from GitHub**.** This confirms your email address and returns you to the sign-up process.

Step7:

Review your plan selection and click Continue. You can also choose whether you want to receive updates from GitHub via email by checking or unchecking the "Send me updates" box.

Step8:

Select your preferences and click Submit. GitHub displays a quick survey that can help you tailor your experience to match what you're looking for. Once you make your selection, you'll be taken to a screen that allows you to set up your first repository.

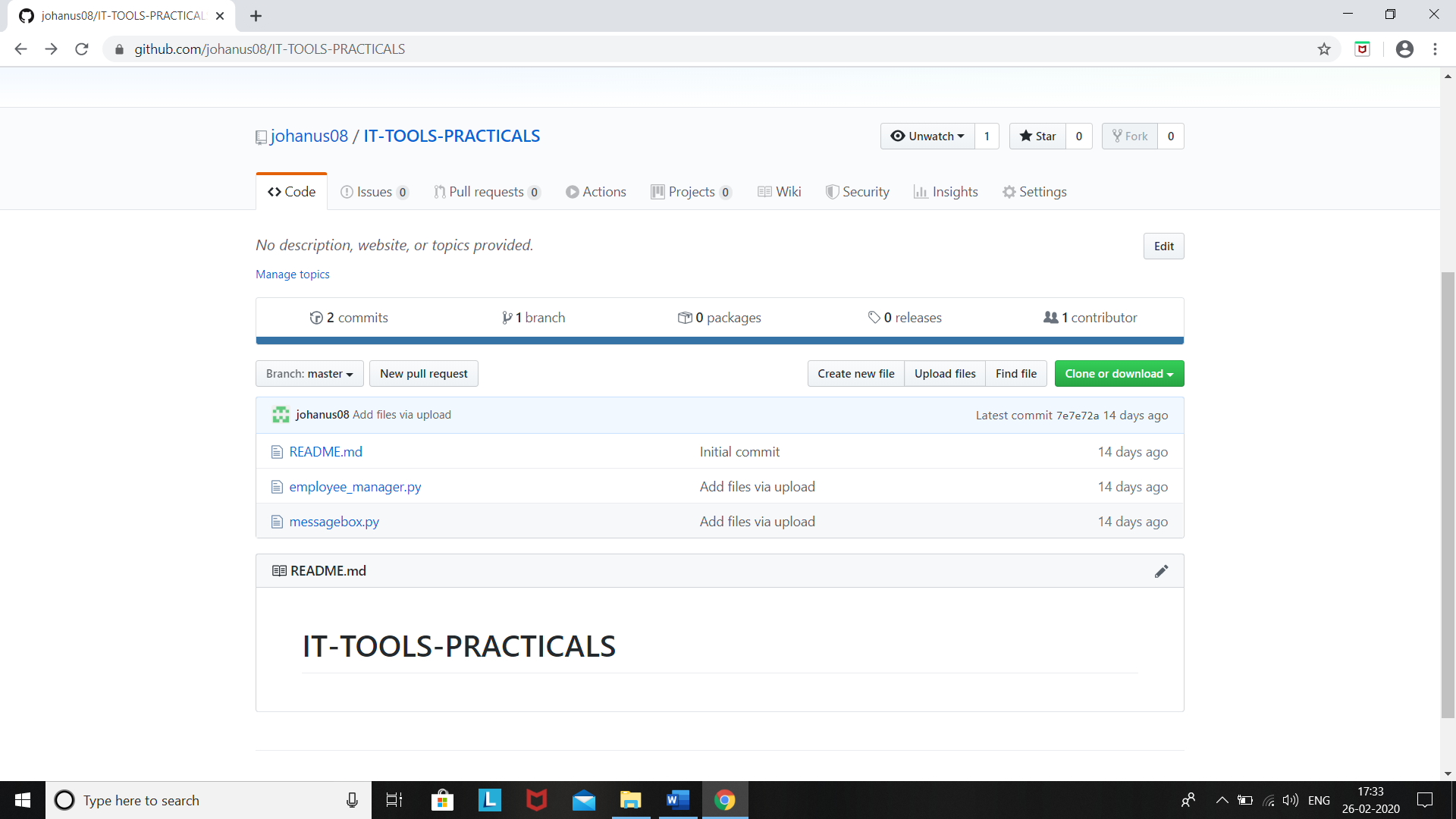


Figure 4 Repository account

# Practical: 6

# BLOG: FOOD

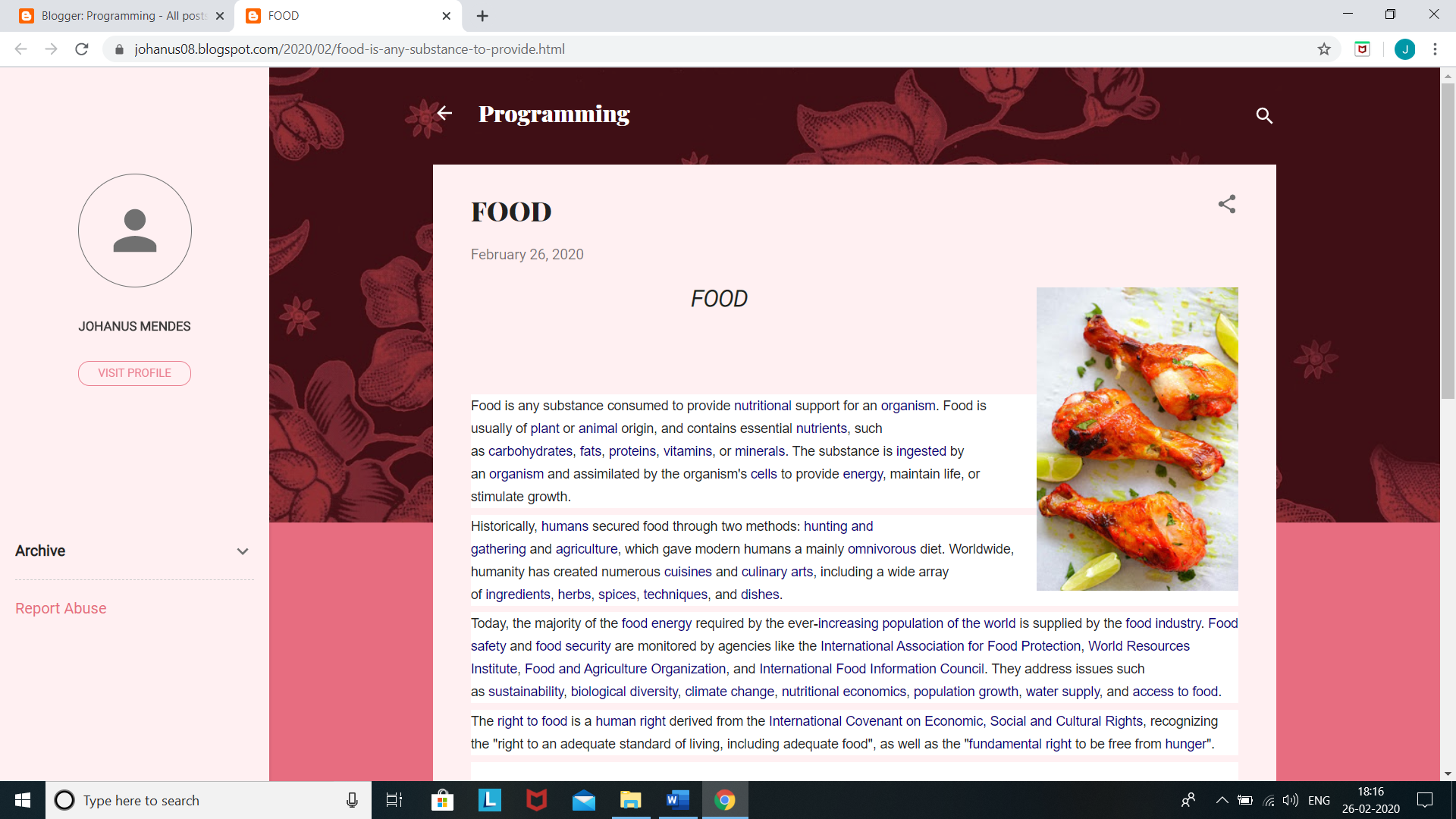
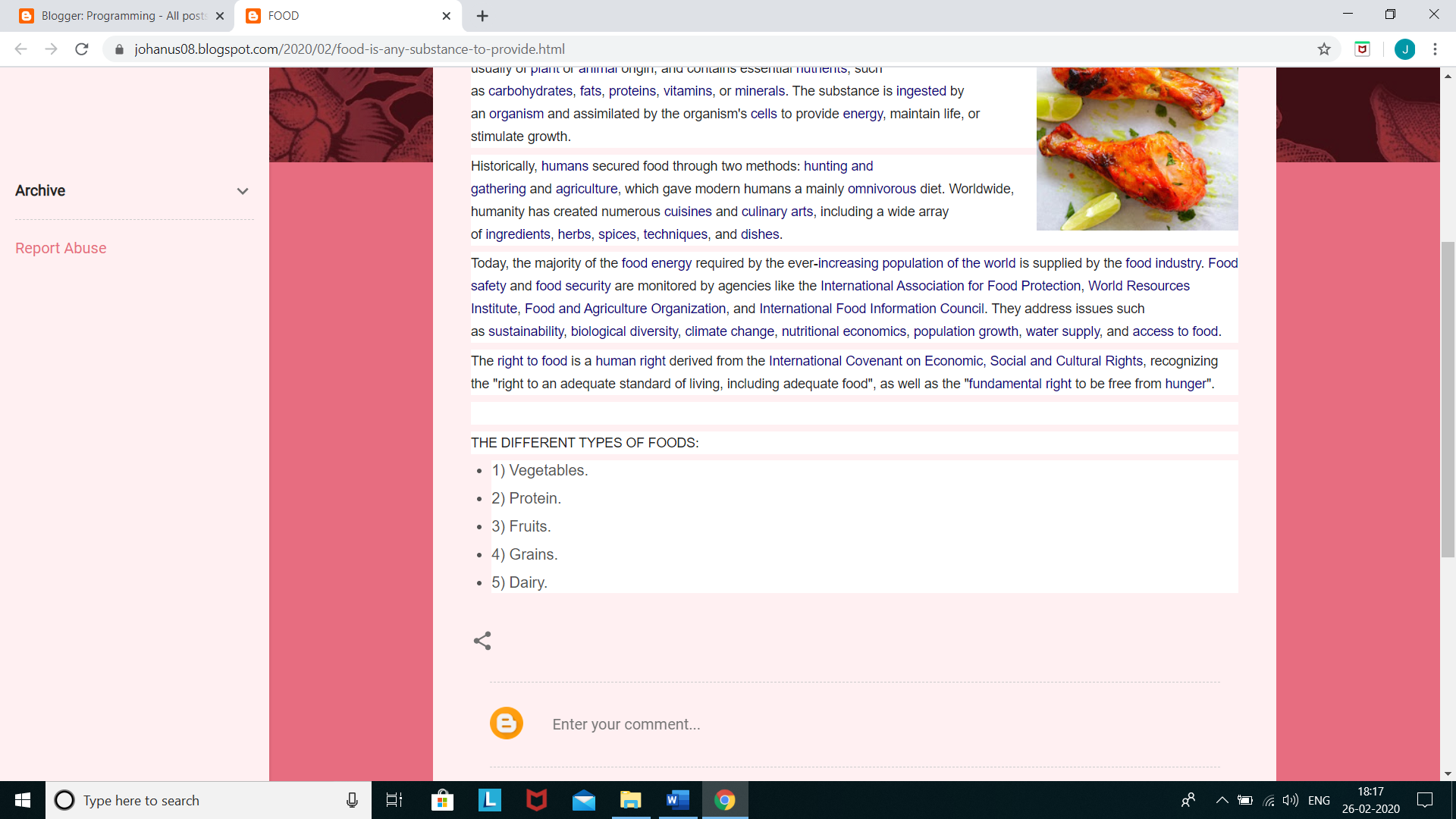


Figure 5 blog food



# 