



**MALAD KANDIVALI EDUCATION SOCIETY'S
NAGINDAS KHANDWALA COLLEGE OF COMMERCE,
ARTS & MANAGEMENT STUDIES & SHANTABEN NAGINDAS
KHANDWALA COLLEGE OF SCIENCE
MALAD [W], MUMBAI – 64
(AUTONOMOUS)**

**(Reaccredited 'A' Grade by NAAC)
(AFFILIATED TO UNIVERSITY OF MUMBAI)
(ISO 9001:2015)**

CERTIFICATE

Name: Mr./Ms. VISHWAKARMA SURYASEN KAILASH

Roll No:139 **Programme: BSc IT/CS** **Semester: II**

This is certified to be a bonafide record of practical works done by the above student in the college laboratory for the course **IT platforms, Tools and Practices** (Course Code: **2026UISTP**) for the partial fulfillment of Second Semester of BSc IT/CS during the academic year 2020-2021.

The journal work is the original study work that has been duly approved in the year 2020-2021 by the undersigned.

External Examiner

Subject-In-Charge (Ms.Sweety Garg)

Date of Examination: (College Stamp)

Sr. No.	DATE	TITLE	SIGN
1.	2/2/21	INTRODUCTION and CONTRIBUTING TO WIKIPEDIA a) What is Wikipedia? b) Steps to Create Account on Wikipedia c) Creating Page on Wikipedia d) Edit your page	
2.	9/2/21	Creating account, repository on GitHub and Cloning repository in GitHub Page	
3.	16/2/21	BASIC UNDERSTANDING ON FREE AND OPEN-SOURCE SOFTWARE a) Describe Open-Source Software with Example. b) Describe Free Software with Example c) Difference between Free and Open-Source Software.	
4.	23/2/21	WRITING EMAIL	
5.	25/2/21	Using practical examples, describe green computing. List and explain the steps that you take to contribute to green computing	
6.	2/3/21	WRITING BLOGS	
7.	9/3/21	Implementing coding practices in Python using PEP8.	
8.	9/3/21	PRESENTATION: PEP8	

Name: - suryasen vishwakrma

Roll No: 139

Class: FYBSCIT

Practical1: Introduction and Contribution to Wikipedia

a) Description about Wikipedia and its features.

- **Overview**

Wikipedia's articles allow the user to use links that are related to the pages for additional information.

It allows you to edit articles for free.

It allows you to put your own article on the site.

Wikipedia has 270 languages.

Wikipedia was founded by Nupedia to produce a free encyclopedia.

- . Quick Information

Wikipedia is a free encyclopedia service available over the internet.

Depending on the topic, wiki's information can be general or very detailed.

Wikipedia appeals to many different types of people. Even though most teachers frown upon it, wiki is used as a starting point for many students when doing a project or presentation.

- **Features and Benefits**

Wikipedia's most known and controversial feature is its editing tool.

You must be a registered user of the site to make changes to any articles.

The editing model is only available in the English edition.

Free editing can be very helpful to the sites users.

If a fact is incorrect it can be changed within seconds by one of Wikis register users.

This can also create issues, because anyone can edit the site can tend to have some news posted that is not fact based.

Wikipedia Features

- Wikipedia has many features for the users on the web. Wiki provides pieces of answers from different resources, to help the user see it from different perspectives. If someone doesn't understand what one editor says, there're always a few more editors with the same answer but with a different response.
- More Features...
 - Users around the world are able to add or delete an answer to make the best response. Wiki added a new feature called vector, which is stylish tabs at the top of each page telling the user whether they're viewing a document or a discussion page.
 - This feature also reminds them if they're reading or editing an entry. Wikipedia is a community effort to create the best answers possible; there are different viewpoints for the user to read.
- Supported Applications
 - Many applications are necessary in order for Wikipedia to function properly.
 - Javascript must be installed on your computer during your research process for the language to be visible and understandable to the reader.
- How Wikipedia is Used
 - Wikipedia can be used in various ways. Companies such as Red Ant, a Sydney, Australia

Name: - suryasen vishwakarma

Roll No: 139

Class: FYBSCIT

based web design and development firm, and Sun Microsystems all benefit from using Wikipedia.

Wikipedia is used as a focal point of the web for employees and customers.

Employees use Wikipedia to attach files, update web pages and integrate information for their job.

- Where Can It Be Purchased

Wikipedia is a free online encyclopedia directed more towards the younger generation.

Anyone can easily access wikipedia by logging on to <http://www.wikipedia.org/>

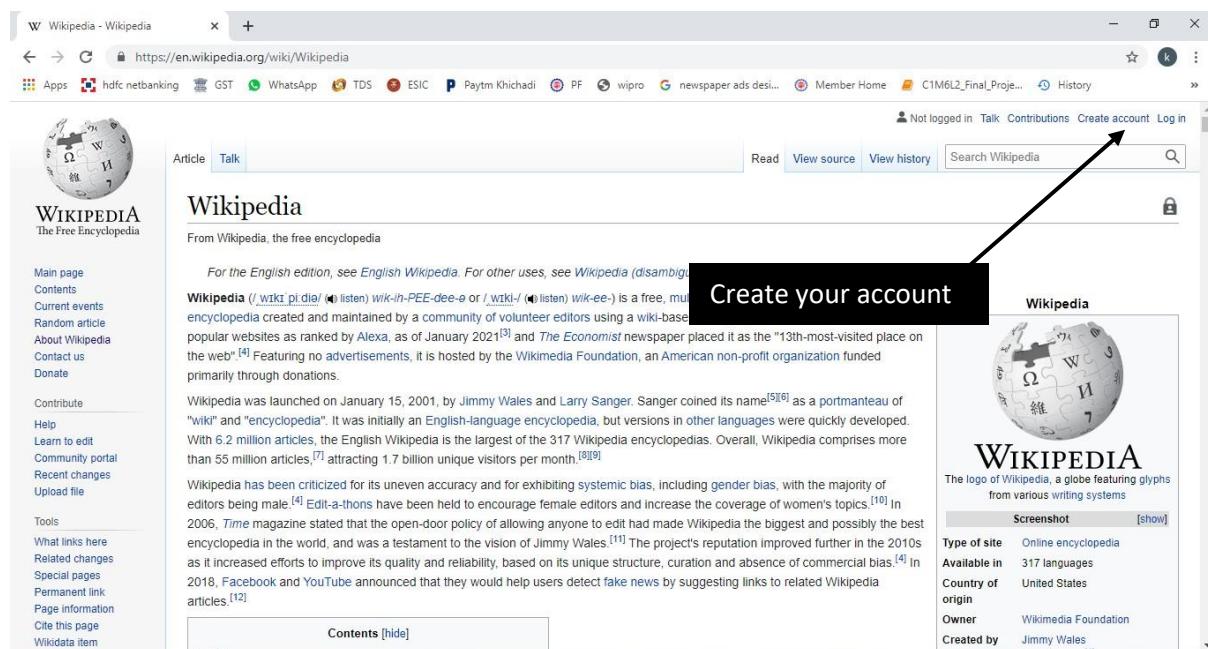
Recommended **Explore professional development books with Scribd** Explore professional development books with Scribd Scribd - Free 30 day trial Wiki powerpoint Wiki powerpoint Penfield Central Schools Wikipedia Wikipedia bookbird Wikipedia Wikipedia ger1989

Wikipedia Wikipedia Laura Wilson Wikipedia Wikipedia rene1517 Wiki ppt Wiki ppt donald.smith KNVI-IP inspiratiemiddag over Wikipedia - presentatie Richard Rogers KNVI-IP inspiratiemiddag over Wikipedia - presentatie Richard Rogers marjobakker Encyclopedias Encyclopedias soher hassan Slowing Growth of Wikipedia and Models of its Dynamic (Presented at Wikimedia Foundataion and WikiSym 2009) Slowing Growth of Wikipedia and Models of its Dynamic (Presented at Wikimedia... Ed Chi Editing Wikipedia Editing

Wikipedia Kathy Gill

b) Creating Account on Wikipedia

Step1: Click on Create account button to create your account.



Name: - suryasen vishwakarma

Roll No: 139

Class: FYBSCIT

Step2: Fill all the information carefully and after that click on the Create your account button to create account.

Your username will be public.
Please consider using an anonymous username, and not your real name, unless you are comfortable with your identity being public for the entire internet to see and identify you.
Once an account has been created, it is essentially impossible to hide the original username should you later want to change it for privacy reasons.

Username (help me choose)
Enter your username

Password
Enter a password

It is recommended to use a unique password that you are not using on any other website.

Confirm password
Enter password again

Email address (optional)
Enter your email address

To protect the wiki against automated account creation, we kindly ask you to enter the words that appear below in the box (more info).
CAPTCHA Security check

zincbirth
C Refresh
Enter the text you see on the image
Can't see the image? Request an account

1,001,033,864 edits
6,245,902 articles
146,633 recent contributors

Create account

c) Creating your page on Wikipedia

Step1: click on the view source button then you will reach on this page then click on submit an edit request button.

View source for Wikipedia - Wikipedia

Read View source View history Search Wikipedia

This page is currently semi-protected so that only established, registered users can edit it.

Why is the page protected?

- While most articles can be edited by anyone, semi-protection is sometimes necessary to prevent vandalism to popular pages.
- The reason for protection can be found in the protection log. If there are no relevant entries in the protection log, the page may have been moved after being protected.

What can I do?

- If you have a user account, log in first. If you do not yet have an account, you may create one; after 4 days and 10 edits, you will be able to edit semi-protected pages.
- Discuss this page with others.
- For move-protected pages, see requested moves.
- Request that the page's protection level be reduced.
- Find out more about how to get started editing Wikipedia.
- If you have noticed an error or have a suggestion for a simple, non-controversial change, you can submit an edit request, by clicking the button below following instructions. An established user may then make the change on your behalf. Please check the talk page first in case the issue is already being discussed.

Submit an edit request

This article relates to Wikipedia itself. Please note that links to non-article namespaces should be treated as external links and not included in the body. Wikipedia is not a reliable source, so references to it must comply with WP:ABOUTSELF.

Request to edit the page

Name: - suryasen vishwakarma

Roll No: 139

Class: FYBSCIT

Step2: Click on New section to create your page.

The screenshot shows a Microsoft Edge browser window with the URL <https://en.wikipedia.org/w/index.php?title=Talk%3AWikipedia&preload=Template%3ASubmit+an+edit+request%2Fpreload&action=edit§ion=new&editintro=Template...>. The page title is 'Editing Talk:Wikipedia (new section)'. At the top right, there are tabs for 'Read', 'Edit source', 'New section', 'View history', and a search bar. A large black arrow points to the 'New section' tab. On the left, there's a sidebar with links like 'Main page', 'Contents', 'Current events', etc. The main content area has sections for 'About edit requests', 'What an edit request IS NOT for', 'What an edit request IS for', and 'When submitting your request'. A note at the bottom says 'This is a talk page. Please respect the talk page guidelines, and remember to sign your posts by typing four tildes ~~~~.'

d) Editing your page on Wikipedia

Step1: Click on Create source button and after that click on User page then you will reach on your page. Here you can Edit your page.

The screenshot shows a Microsoft Edge browser window with the URL <https://en.wikipedia.org/w/index.php?title=User%3AKomalmenariacpm&action=edit>. The page title is 'Creating User:Komalmenariacpm'. At the top right, there are tabs for 'User page', 'Talk', 'Create source', and a search bar. A large black arrow points to the 'Create source' tab. On the left, there's a sidebar with links like 'Main page', 'Contents', 'Current events', etc. The main content area has a note: 'Wikipedia does not have a user page with this exact title. In general, this page should be created and edited by User:Komalmenariacpm.' It also says 'If you want to draft an article, please create a userspace draft instead of creating it here.' A note at the bottom says 'Content that violates any copyrights will be deleted. Encyclopedic content must be verifiable. Any work submitted to Wikipedia can be edited, used, and redistributed—by anyone—subject to certain terms and conditions.'

Name: - suryasen vishwakarma

Roll No: 139

Class: FYBSCIT

Step2: Here you can fill necessary information and at the end click on Publish changes button to Publish your page information.

Editing User:Komalmenariacpm

Content that violates any copyrights will be deleted. Encyclopedia content must be verifiable. Any work submitted to Wikipedia can be edited, used, and redistributed—by anyone—subject to certain terms and conditions.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Tutorial 6 on php variables</title>
</head>
<body>
    <div class="container">
        <div>More on php variables</div>
        <div>Rules for creating variables in php</div>
        <div>Start with $ sign</div>
        <div>Must start with letter or an underscore character</div>
        <div>Can only contain alphanumeric character and underscore</div>
        <div>Variables in php are case sensitive like ($name,$Name,$NAME they are different variables)</div>
    </div>
</body>
</html>
```

\$name = "komal";
echo '\$name';
echo "This is more on php";
// below type of variable give error

By publishing changes, you agree to the Terms of Use, and you irrevocably agree to release your contribution under the CC BY-SA 3.0 License and the GFDL. You agree that a hyperlink or URL is sufficient attribution under the Creative Commons license.

This is a minor edit Watch this page Permanent Preview

Publish changes Show preview Show changes Cancel

Step3: Click on the read button and you will reach on your edited page.

User page Talk

Wikimedia Wikimeet India 2021!
Registration open for Wikimedia Wikimeet India 2021 until 16 February 2021!
Main Event: February 19 - 21, 2021

User:Komalmenariacpm

From Wikipedia, the free encyclopedia

```
<!DOCTYPE html> <html lang="en"> <head>
```

More on php variables

Rules for creating variables in php

- Start with \$ sign
- Cannot start with numbers
- must Start with letter or an underscore character
- Can only contain alphanumeric character and underscore
- variables in php are case sensitive like (\$name,\$Name,\$NAME they are different variables)

```
<php $name = "komal";>  
echo '$name';  
"echo "This is more on php";  
// below type of variable give error // $name% = "komal"; // echo "$name%"; ?>
```

Name: - suryasen vishwakrma

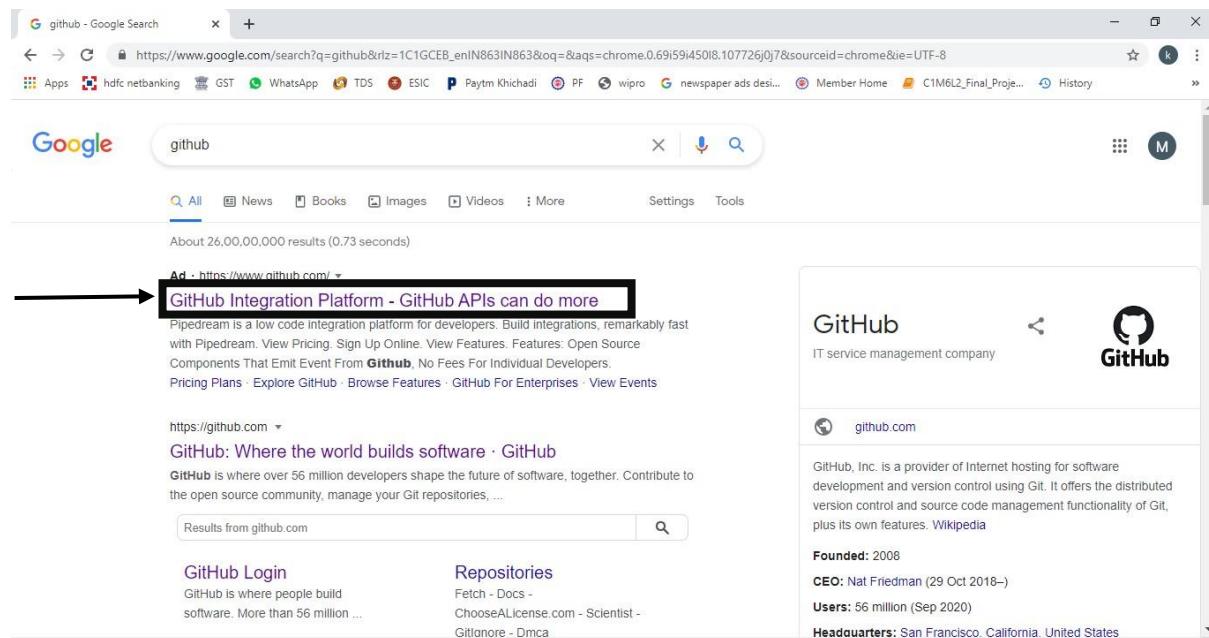
Roll No: 139

Class: FYBSCIT

PRACTICAL 2: Creating account, repository on Github s and cloning repository in Github

a) Creating account: -

Step1: - Search Github on Google and click on the official link of Github.

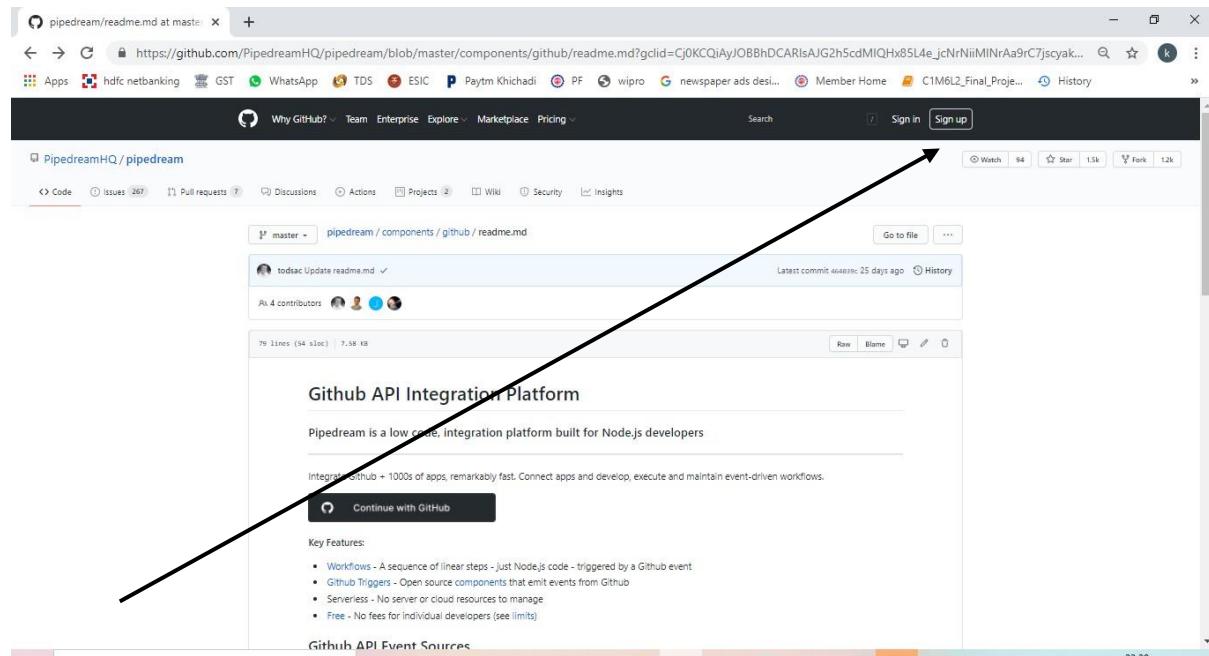


Name: - suryasen vishwakrma

Roll No: 139

Class: FYBSCIT

Step 2: Click on the sign up button to create an account on Github.



Step 3: When you click on the signup button you will reach on this page

here you have to fill all necessary information which is star marked.

A screenshot of a web browser displaying the 'Create your account' form on GitHub. The URL is https://github.com/join?ref_cta=Sign+up&ref_loc=header+logged+out&ref_page=%2F<user-name>%2F<repo-name>%2Fblob%2Fshow&source=header-repo&sour... A large black arrow points from the top right towards the 'Create account' button at the bottom left of the form. To the right of the form, there are four black boxes with white text and arrows pointing to specific fields:

- INSERT YOUR NAME (points to the 'Username *' field)
- INSERT YOUR EMAIL ID (points to the 'Email address *' field)
- INSERT PASSWORD (points to the 'Password *' field)
- VERIFY YOUR ACCOUNT (points to the CAPTCHA verification area)
- CLICK ON THE CREATE ACCOUNT (points to the 'Create account' button)

Name: - suryasen vishwakrma

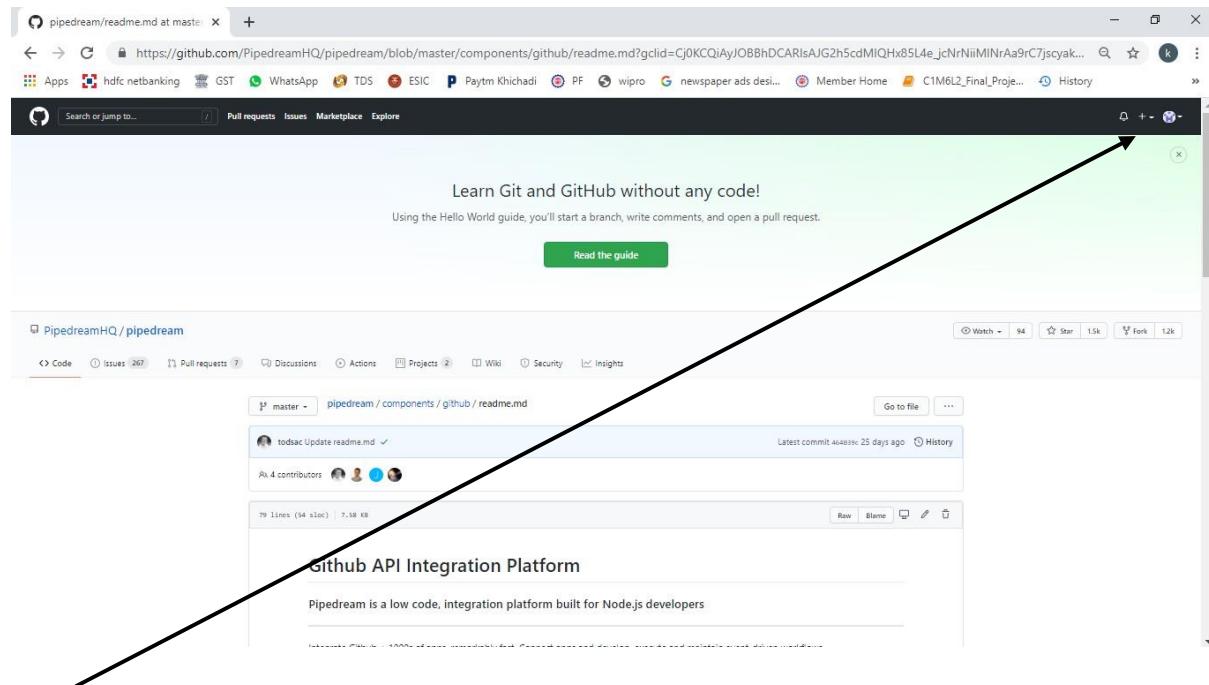
Roll No: 139

Class: FYBSCIT

Step 4: After fill all the information verify your account and click on the Create Account button. Now your account is created.

b) Creating repository:

Step 1: To create repository click on the + button .

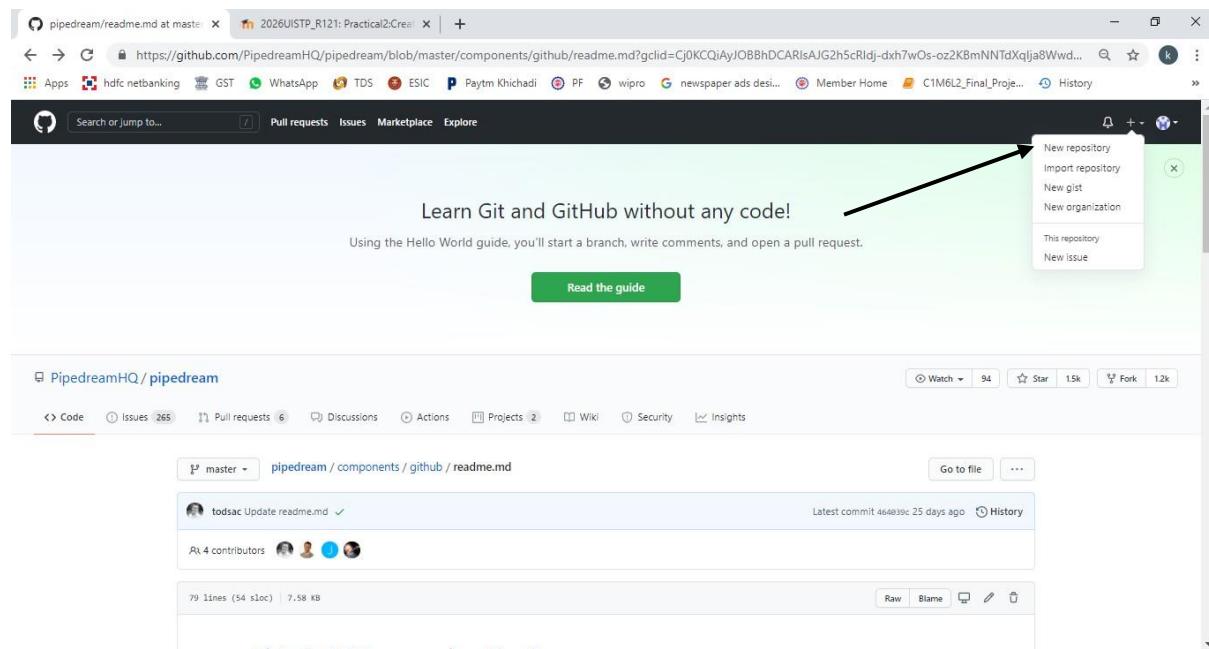


Step 2: Click on New repository to make your repository.

Name: - suryasen vishwakarma

Roll No: 139

Class: FYBSCIT



Step 3: After clicking on the New repository you reach on this page here you have to give your Repository name, click on the private or public button.

Description: It is optional, so if you want to describe something about your repository then fill the description field with necessary information

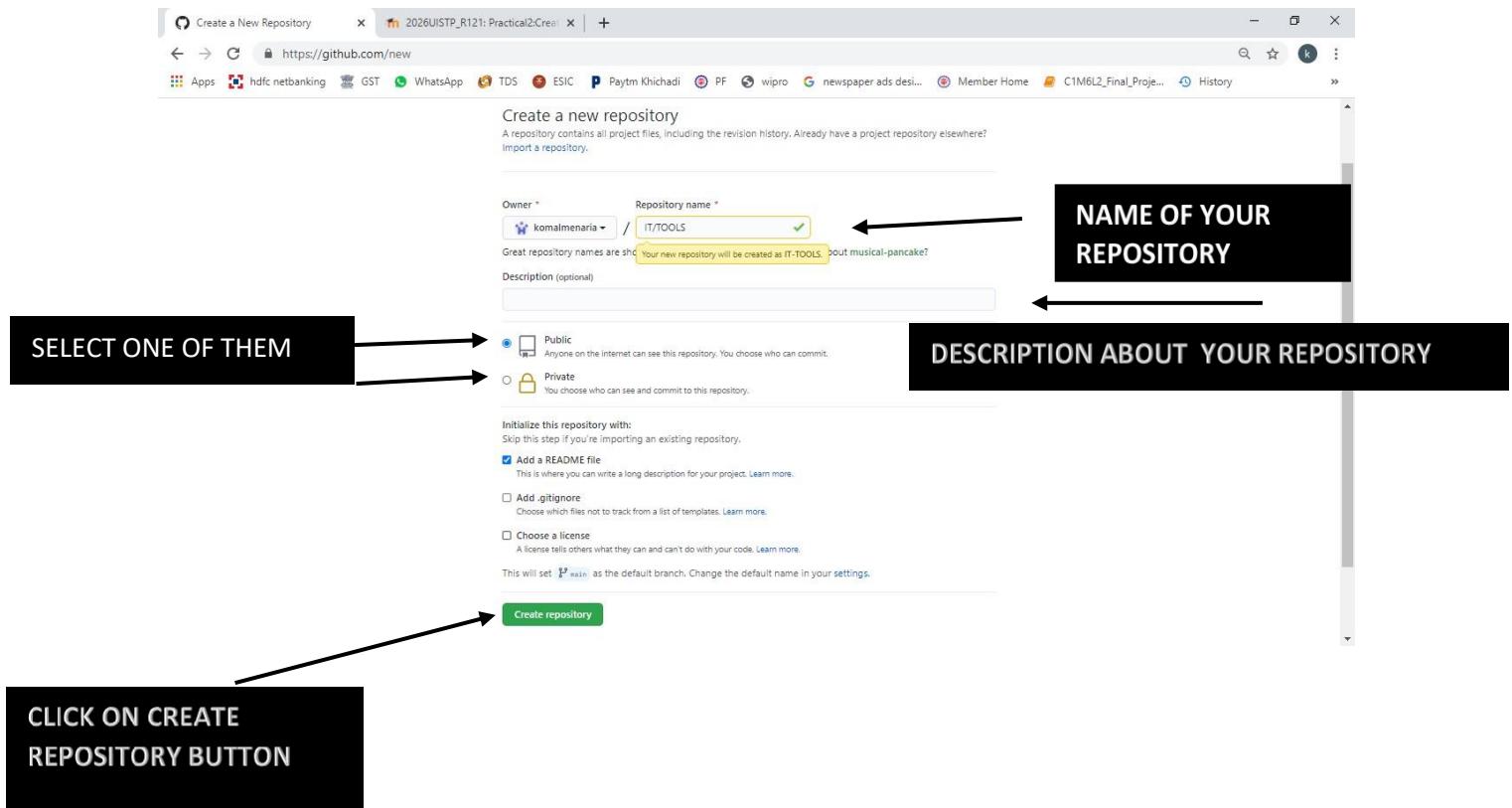
1 Private: If you want no one can access your repository then click on private button.

2 Public: If you want other people can access your repository then click on public button.

Name: - suryasen vishwakarma

Roll No: 139

Class: FYBSCIT



Step 4: In the last click on the create repository button.

c) Cloning repository:

After clicking on the create repository button you will reach on this page

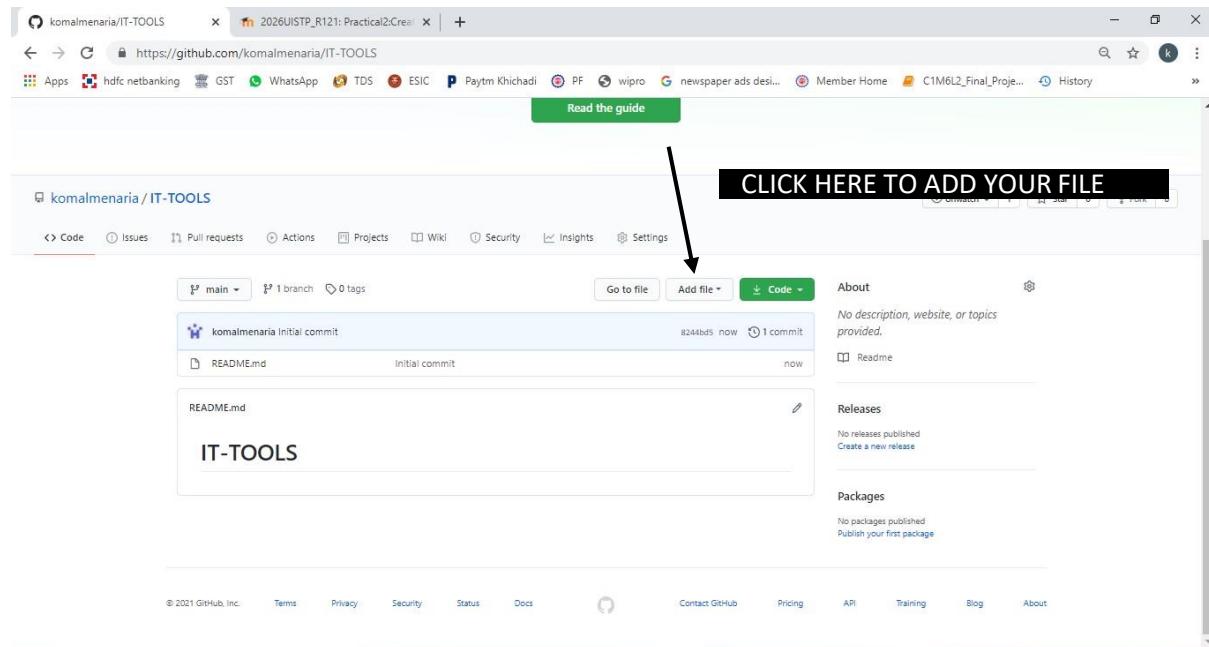
Step 1: Click on the Add file button .



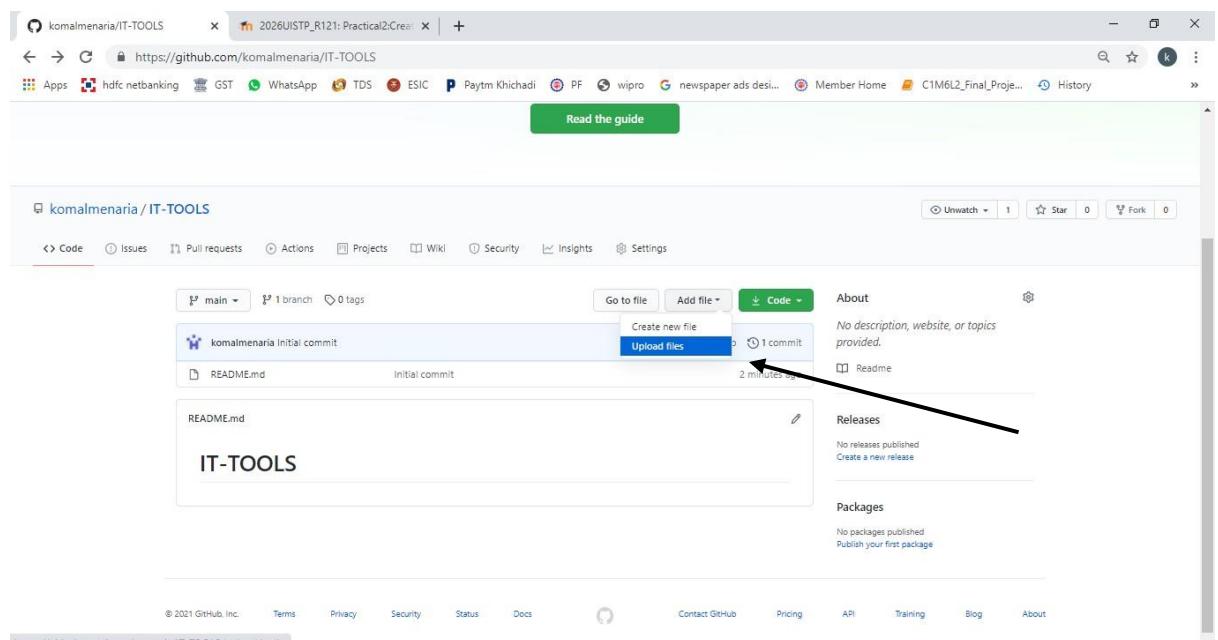
Name: - suryasen vishwakarma

Roll No: 139

Class: FYBSCIT



Step 2 click on upload files here you can add any no of file in the repository.

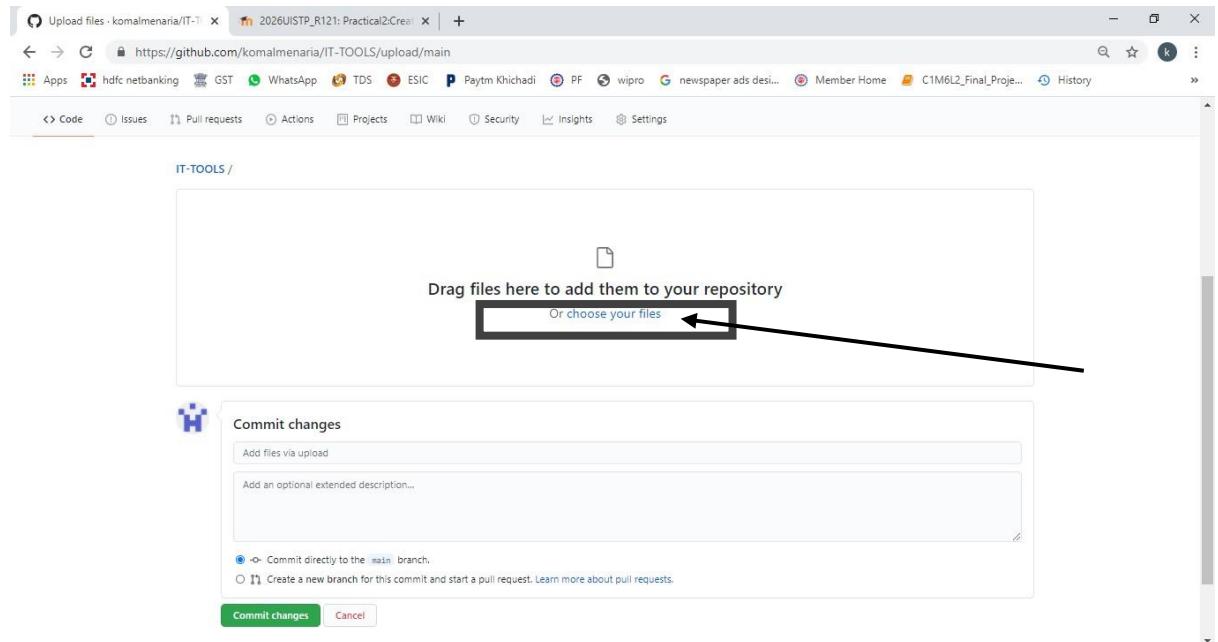


Step 3: Click on choose your files to select your file.

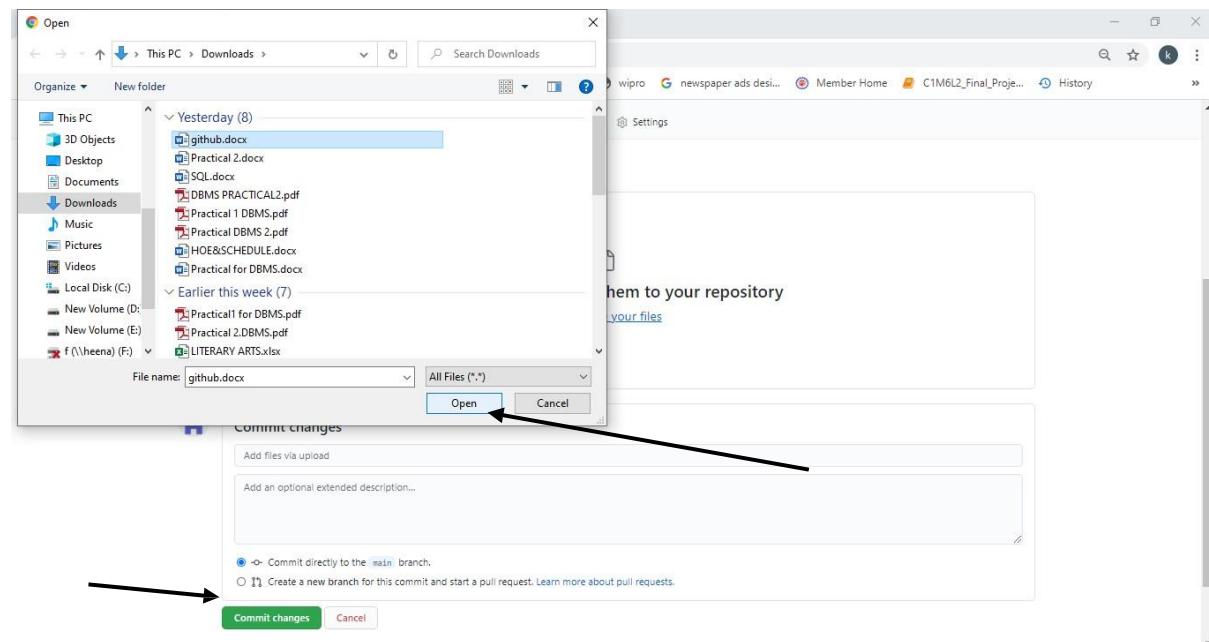
Name: - suryasen vishwakarma

Roll No: 139

Class: FYBSCIT



Step 4: Select the file which you want and click on open button and atlast click on the commit changes button to save the changes.



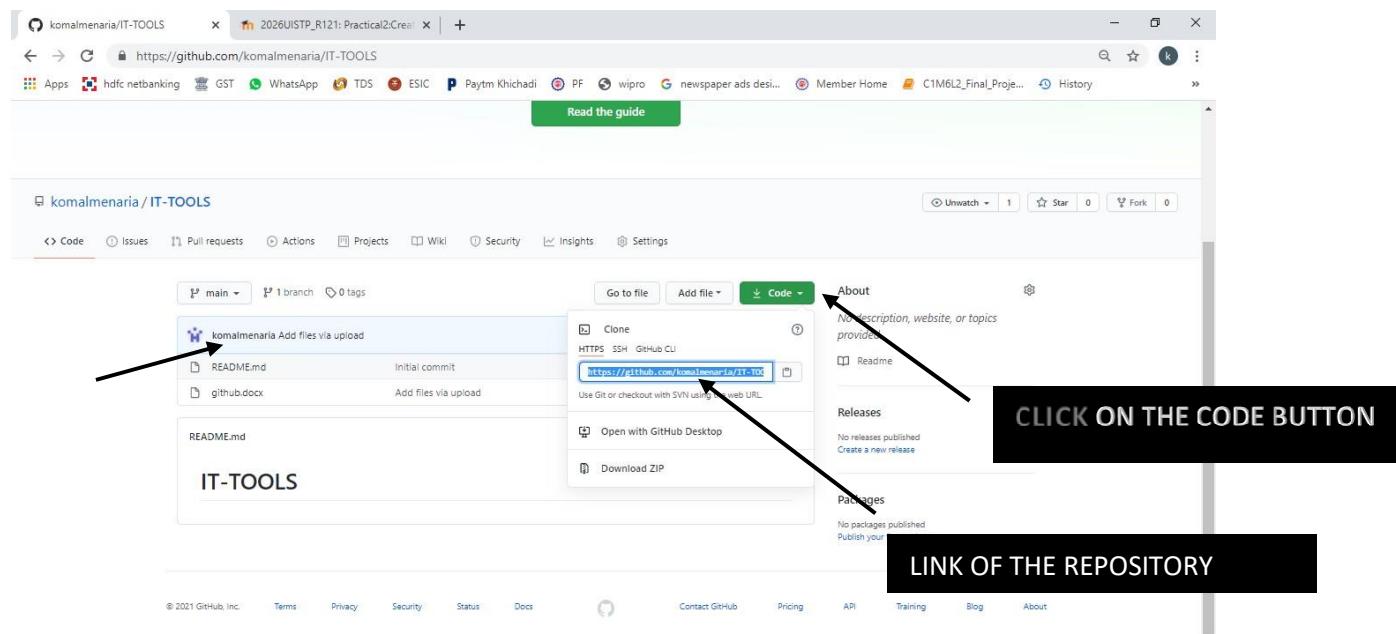
Name: - suryasen vishwakarma

Roll No: 139

Class: FYBSCIT

Step 4: click on the code button to get your repository link and copy the link

To share your repository.



PRACTICAL 3: BASIC UNDERSTANDING ON FREE AND OPEN-SOURCE SOFTWARE

a) Describe open source Software with Example.

Open-source software is a type of computer software in which source code is released under a license in which the copyright holder grants users the rights to use, study, change, and distribute the software to anyone and for any purpose. Open-source software may be developed in a collaborative public manner.

Open source Software is released through a specific kind of license that makes its source code legally available to end-users. The source code can be repurposed into other new software, meaning anyone can take the source code and distribute their own program from it.

Some examples of open source software:

Linux operating system

Android by Google

Open office

Firefox browser

VCL media player

Moodle

b) Describe Free Software with Example

The concept of free software is the brainchild of Richard Stallman, head of the GNU Project. The best known example of free software is Linux, an operating system that is proposed as an alternative to Windows or other proprietary operating systems. Debian is an example of a distributor of a Linux package.

Some examples of free source software:

- The Linux kernel, of course! The Linux kernel is protected by the GPL, and is used daily by millions of people throughout the world.
...
- Apache, the most widely used web server in the world. ...
- The Gimp is a powerful bitmap mode digital creation program. ...
- PostgreSQL is an object-relational database.

c) Difference between Free and open source Software.

SR.NO	Free Source Software	Open source Software
1	Software is an important part of people's lives.	Software is just software. There are no ethics associated directly to it.
2	Software freedom translates to social freedom.	Ethics are to be associated to the people not to the software.
3	Freedom is a value that is more important than any economical advantage.	Freedom is not an absolute concept. Freedom should be allowed, not imposed.
4	Examples: The Free Software Directory maintains a large database of free-software packages. Some of the best-known examples include the Linux kernel, the BSD and Linux operating systems, the GNU Compiler Collection and C library; the MySQL relational database; the Apache web server; and the Sendmail mail transport agent.	Examples: Prime examples of open-source products are the Apache HTTP Server, the e-commerce platform osCommerce, internet browsers Mozilla Firefox and Chromium (the project where the vast majority of development of the freeware Google Chrome is done) and the full office suite LibreOffice.

Practical 4-

Job application - tonudon86@gmail.com | Job application - vishwakarma.su | 2026UISTP_R121: PRACTICAL 4 | formal letter for job application

mail.google.com/mail/u/0/#inbox/

Compose

Job application

SURYASEN VISHWAKARMA

To me

Recruiter

Rogers Consulting
901 Main Street
New York, NY 10001

Dear Ms. Jenkins,

I am reaching out to you regarding the posting for the human resources consultant position I found on Indeed.com. I have a great interest in this position and would appreciate your consideration as a candidate for the role.

In my previous experience, I worked in human resources departments to provide support across several different industries. I have worked in my current role as a human resources generalist for the past four years. Prior to this job, I worked as a human resources assistant for two years, which shows my ability to advance in my career.

I have a strong passion for helping others, which is why I have found such fulfillment in human resources, providing support to my fellow employees and assisting them in ways that benefit them both personally and professionally. I also enjoy looking for solutions to common HR problems, which I feel would be a great asset in the position with your company. Since this consultant position works directly with multiple clients, assisting them in their human resources needs, I believe my innovative nature and strong skill set will help me succeed.

I have strong communication skills, which are vital to success in the HR field. I also have a bachelor's degree in human resources from Arizona State University. Throughout my education, I worked with skilled human resources professionals who have shared their insights and experience with me. Some of my strongest skills include my ability to increase employee retention through the improvement of company culture and to develop training and education programs to ensure all employees have access to the information they need to succeed and comply with legal requirements.

I appreciate your time in reviewing this letter and hope to hear from you in regard to the next steps in the hiring process. If you have any questions or need any additional information, please don't hesitate to contact me.

Sincerely,
suryasen.vishwakarma

Reply Forward

PRACTICAL 4 email.pdf

Type here to search

Show all

ENG IN 08:42 PM 16-03-2021

Name:suryasen vishwakrma

Roll no : 139

Class : FY BSC IT

1. Explain Green Computing with its advantages.

Green Computing refers to durable computing of the environment. This reduces the use of electricity as well as power and reduces environmental waste when we are using a computer. It Computing has the same goal with green chemistry, which is now the life of the product and makes the product more energy efficient, the abandoned product and factory waste are more easily recycled and to be biodegradable, less Dangerous Use Content.

Advantages of Green Computing:

Here different benefits of green computing are

Lessened vitality utilization by green registering advances converts into low carbon dioxide emanations, which emerge because of the absence of petroleum derivatives utilized as a part of intensity plants and transportation.

Conservation of resources means less energy is required to produce, use and dispose of products.

Saving energy and resources saves money.

Green processing includes changing government arrangement to empower reusing by people and organizations and to lessen vitality utilization.

Reduce existing exposure in laptops such as chemical, cancer, nerve damage, and is known due to immune responses in humans.



2. What is E-waste? What can be done to reduce the impact of E-waste.

E-waste, or electronic waste, encompasses electrical and electronic equipment that's outdated, unwanted, or broken. That means everything from smartphones to end-of-life refrigerators. Basically, anything that runs on electricity that you've decided to get rid of.

Name:suryasen vishwakrma

Roll no : 139

Class : FY BSC IT

Globally, we only recycle 10% of our e-waste, a number that's as shocking as it is depressing. As for the 90% we don't recycle, it ends up getting landfilled, incinerated, or illegally traded.

Reduce: The easiest way to solve the e-waste crisis is to produce less e-waste. I know, easier said than done. Companies are constantly rolling out new products—like Apple, for instance, with its iPhone. Newer products look and function better than their predecessors, but novelty comes at a price. Instead of buying that flashy new gadget, stick to what you've got. Also, by taking care of your electronics you can ensure that they last longer. When you don't have to replace them as often, you end up saving money.

Reuse: Instead of tossing out that old television set or gaming console, consider regifting, selling, or donating it. You could also hold on to it. Who knows, it might end up being worth something someday. Look at the Original Apple 1. It's sold at auction for upwards of \$905,000.

Repair: People often throw out and replace broken electronics instead of getting them repaired. True, repairs can be expensive, but for those who aren't afraid of a DIY project, it's a cheap fix. Online resources like , iFixit a website that boasts free repair guides for everything, provide reliable information that'll help you get your tech back in working order. Always remember to be safe, though.

Recycle: As a last resort, you can always recycle your e-waste—just make sure you're doing it correctly! Many communities have e-waste recycling events and drop-off depots that handle these materials. If your city is currently a member of our network, download the Recycle Coach app to find out how. Organizations like TerraCycle accept e-waste in the U.S. and Canada. So do some manufacturers and retailers, like Apple and Best Buy.



3. What are the benefits of going paperless.

A paperless approach offers a much higher level of **security** for your and your customers' sensitive data. Numerous safeguards, encryption, and banking-level **security** measure work together to protect your documents. In a paper-based office, everyone has access to all the information.

Name:suryasen vishwakrma

Roll no : 139

Class : FY BSC IT

Going paperless helps to reduce C02 (carbon dioxide) emissions. Turning a single tree into 17 reams of paper results in around 110 lbs of C02 being released into the **atmosphere**. Additionally, trees are also 'carbon sinks' and every tree that is not cut down for paper usage is able to absorb C02 gasses.

Benefits of Going Paperless

- Saves Time. Time spent filing, organizing, and searching for paper documents is time that could be spent on more productive tasks. ...
- Saves Space. ...
- Saves **Money**. ...
- Eases Transfer of Information. ...
- Promotes the Environment. ...
- Boosts **Security**.

1. Document organization

The ability to quickly locate and disseminate information may enhance your company's efficiency and professional image. Spending time hunting through piles of paper slows down response time in an age when most answers are only a few keystrokes away. By scanning electronic copies of receipts and invoices, documents can be sorted, filed, and organized for quick retrieval when it matters most.

2. Client communication is faster and less expensive

By maintaining a customer email list, you can instantaneously communicate sales and special offers without incurring postage and printing expenses. With the advanced technology of smart devices, most people have immediate access to emails. While it increases efficiency, electronic communication also decreases storage costs as the amount of paper copies littering your office will begin to dwindle.

3. Paperless files are easily saved and retrieved on the go

With the advent of photo-scanning apps, business travelers can easily back up expense reports without needing to save a pile of papers to bring back to the office. Electronic files can also be shared with coworkers over a network or via email. Shifting to paperless documentation also makes the transportation of data more efficient, without the need for cumbersome fax machines or document couriers.

4. Automatic backups

When you accidentally throw out an important paper, it's usually gone forever. However, maintaining electronic files allows for multiple backup points. Data can be saved on flash drives, in the cloud, or to an external hard drive. For vitally important financial data, cloud

Name:suryasen vishwakrma

Roll no : 139

Class : FY BSC IT

based accounting systems provide automatic backups on a pre-scheduled basis, which eliminates the need for small business owners to set aside time for manual backups.

5. Data security

Customers will always be concerned about privacy and data protection, which requires companies to respond by implementing proper data security procedures beyond locked filing cabinets and paper shredders. Many of today's cloud-based accounting systems offer bank-level data security to protect financial and customer information, which is more than most small companies with limited technology staff can afford to build in-house.

6. Environmental friendliness

According to the Environmental Paper Network's most recent State of the Paper Industry report, paper usage in North America is decreasing while the amount of paper recovered for recycling is increasing. Companies are striving to recycle, yet office copy paper alone still accounts for over 20 percent of the total paper usage in the United States. But being green is more than just reducing paper production. A paperless environment may also mean less energy consumption. Small businesses use less energy when printers, faxes, and copiers are inactive.

7. Financial benefits

The savings of going paperless extends beyond just the cost of the paper, which can be substantial. The cost of other office supplies like ink cartridges also decreases. Additional upgrades or replacements to expensive office equipment such as copiers and fax machines may also decrease in a paperless office.

The shift toward a paperless environment increases each year as new technology becomes available to improve data storage and electronic communication. Taking action to reduce paper usage may help your business be more efficient and enhance the level of security that guards your most valuable information.



Name:suryasen vishwakrma

Roll no : 139

Class : FY BSC IT

4. What is Github? Give advantages of using Github.

GitHub is a code hosting platform for version control and collaboration. It lets you and others work together on projects from anywhere. This tutorial teaches you **GitHub** essentials like repositories, branches, commits, and Pull Requests.

GitHub today announced that all of its core features are now available for **free** to all users, including those that are currently on **free** accounts.



Advantages

1. It makes it easy to contribute to your open source projects

To be honest, nearly every open-source project uses GitHub to manage their project. Using GitHub is free if your project is open source and includes a wiki and issue tracker that makes it easy to include more in-depth documentation and get feedback about your project. If you want to contribute, you just fork a project, make your changes and then send them a pull request using GitHub web interface.

2. Documentation

By using GitHub, you make it easier to get excellent documentation. Their help section and guides have articles for nearly any topic related to git that you can think of.

3. Showcase your work

Are you a developer and wishes to attract recruiters? GitHub is the best tool you can rely on for this. Today, when searching for new recruits for their project, most companies look into the GitHub profiles. If your profile is available, you will have a higher chance of being recruited even if you are not from a great university or college.

4. Markdown

Markdown allows you to use a simple text editor to write formatted documents. GitHub has revolutionized writing by channeling everything through Markdown: from the issue tracker,

Name:suryasen vishwakrma

Roll no : 139

Class : FY BSC IT

user comments, everything. With so many other programming languages to learn for setting up projects, it's really a big benefit to have your content inputted in a format without having to learn yet another system.

5. GitHub is a repository

This was already mentioned before, but it's important to note, GitHub is a repository.

What this means that it allows your work to get out there in front of the public. Moreover, GitHub is one of the largest coding communities around right now, so it's wide exposure for your project.

6. Track changes in your code across versions

When multiple people collaborate on a project, it's hard to keep track revisions—who changed what, when, and where those files are stored. GitHub takes care of this problem by keeping track of all the changes that have been pushed to the repository. Much like using Microsoft Word or Google Drive, you can have a version history of your code so that previous versions are not lost with every iteration.

7. Integration options

GitHub can integrate with common platforms such as Amazon and Google Cloud, services such as Code Climate to track your feedback, and can highlight syntax in over 200 different programming languages.

5. Write a program using PEP8 rules.

PEP 8 exists to improve the readability of Python code. Code is read much more often than it is written.

Name:suryasen vishwakrma

Roll no : 139

Class : FY BSC IT

```
practical 5b oop.py - C:\Users\HP\practical 5b oop.py (3.8.8)
File Edit Format Run Options Window Help
# to demonstrate the use of getter and setter method using property decorator.
class property_decorator_demo:
    def __init__(self):
        self._age = 0

    @property
    def get_age(self):
        print("Getter method called. ")
        return self._age

    @get_age.setter
    def age(self,a):
        if(a < 18):
            raise ValueError("Sorry your age is below eligibility criteria.")
        print("Setter method called.")
        self._age = a

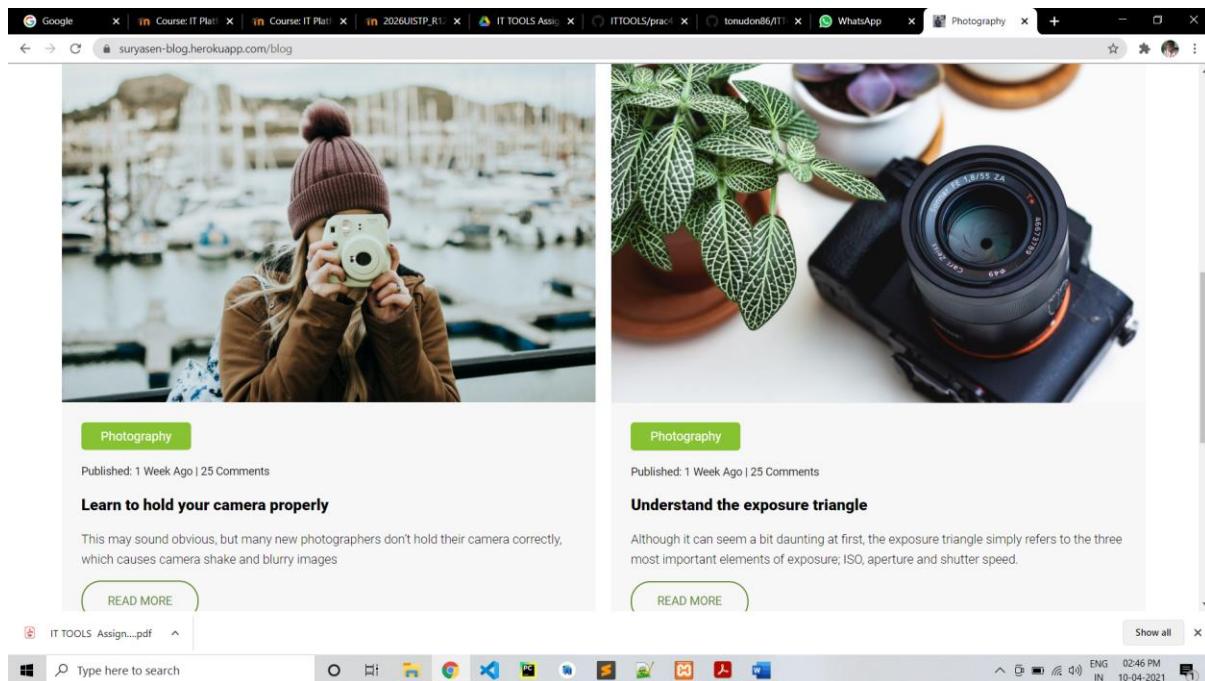
# Creating a object of class.
mark = property_decorator_demo()

#Calling method through object.
mark.age = 19
print(mark.age)
```

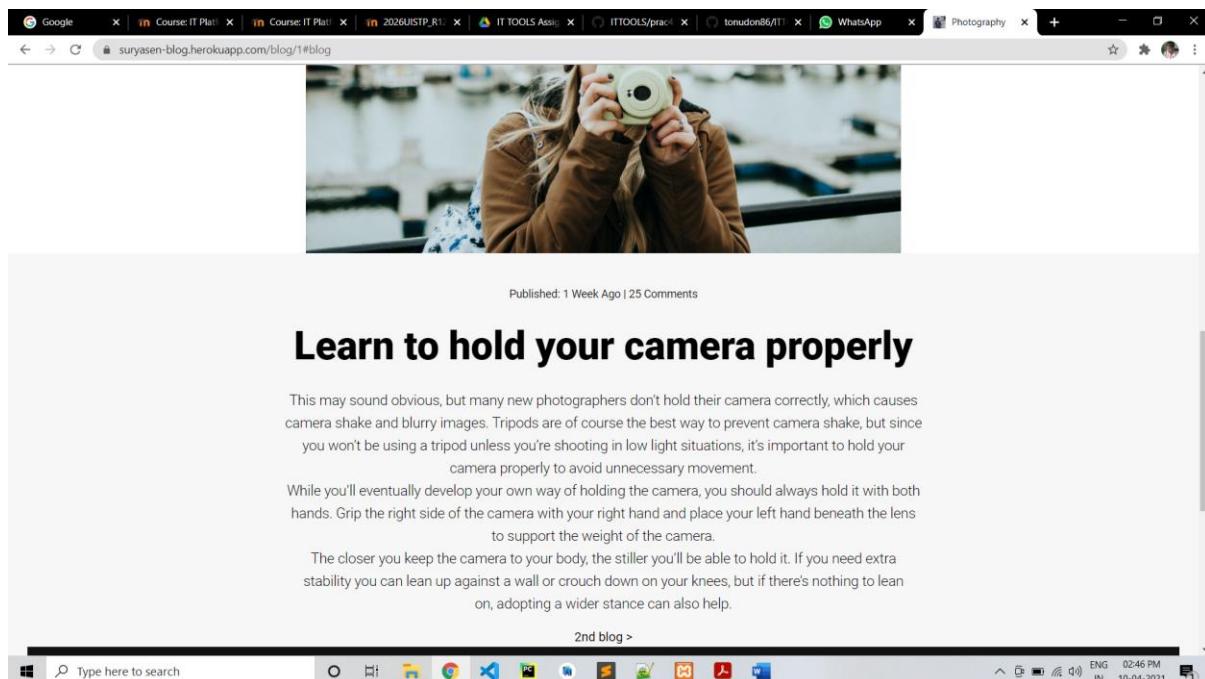
Output:

```
IDLE Shell 3.8.8
File Edit Shell Debug Options Window Help
Python 3.8.8 (tags/v3.8.8:024d805, Feb 19 2021, 13:18:16) [MSC v.1928 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
=====
RESTART: C:\Users\HP\practical 5b oop.py =====
Setter method called.
Getter method called.
19
>>>
```

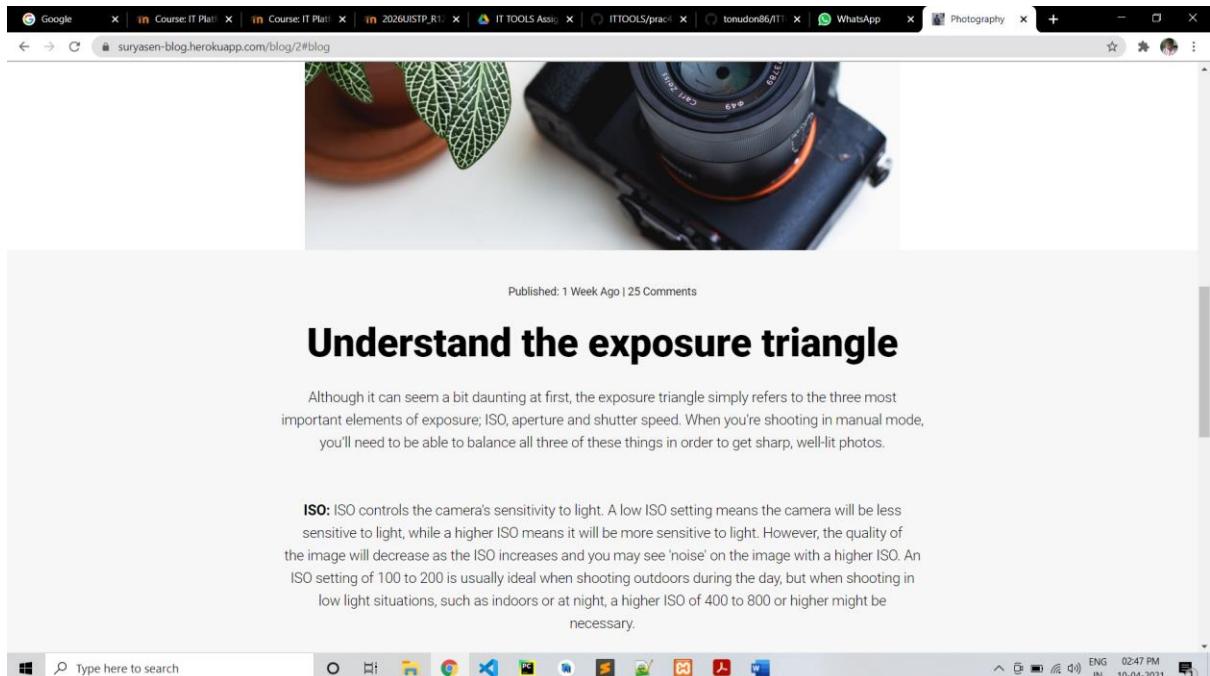
PRACTICAL 6: WRITING BLOGS ..



1st blog



2nd blog



PRACTICAL7: Implementing coding practices in Python using PEP8

PEP 8 exists to improve the readability of Python code.

1: Naming Conventions

When you write Python code, you have to name a lot of things: variables, functions, classes, packages, and so on. Choosing sensible names will save you time and energy later. You'll be able to figure out, from the name, what a certain variable, function, or class represents. You'll also avoid using inappropriate names that might result in errors that are difficult to debug.



A screenshot of a code editor window titled "PEP8 PROGRAM.py". It shows the line of code "p = 2 # This may look like you're trying to reassign 2 to zero". A yellow warning icon is positioned next to the line, and a tooltip message "A 2 ^ v" is visible above the code area.

2: How to Choose Names

When naming variables, you may be tempted to choose simple, single-letter lowercase names, like x. But, unless you're using x as the argument of a mathematical function, it's not clear what x represents

3: Code Layout

Blank Lines

Vertical whitespace, or blank lines, can greatly improve the readability of your code. Code that's bunched up together can be overwhelming and hard to read. Similarly, too many blank lines in your code makes it look very sparse, and the reader might need to scroll more than necessary. Below are three key guidelines on how to use vertical whitespace.

Surround top-level functions and classes with two blank lines.

Top-level functions and classes should be fairly self-contained and handle separate functionality. It makes sense to put extra vertical space around them, so that it's clear they are separate:



```
PEP8 PROGRAM.py
1 class MyFirstClass:
2     pass
3
4
5 class MySecondClass:
6     pass
7
8
9 def top_level_function():
10    return None
```

Surround method definitions inside classes with a single blank line.

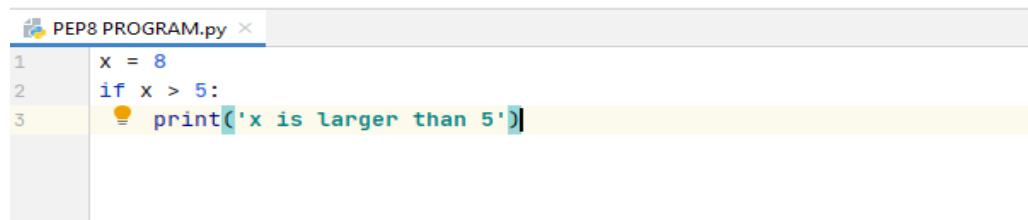
Inside a class, functions are all related to one another. It's good practice to leave only a single line between them:



```
PEP8 PROGRAM.py
1 class MyClass:
2     def first_method(self):
3         return None
4
5     def second_method(self):
6         return None
```

4: Indentation

Indentation, or leading whitespace, is extremely important in Python. The indentation level of lines of code in Python determines how statements are grouped together.



```
PEP8 PROGRAM.py
1 x = 8
2 if x > 5:
3     print('x is larger than 5')
```

5: Comments

You should use comments to document code as it's written. It is important to document your code so that you, and any collaborators, can understand it. When you or someone else reads a comment, they should be able to easily understand the code the comment applies to and how it fits in with the rest of your code.

Here are some key points to remember when adding comments to your code:

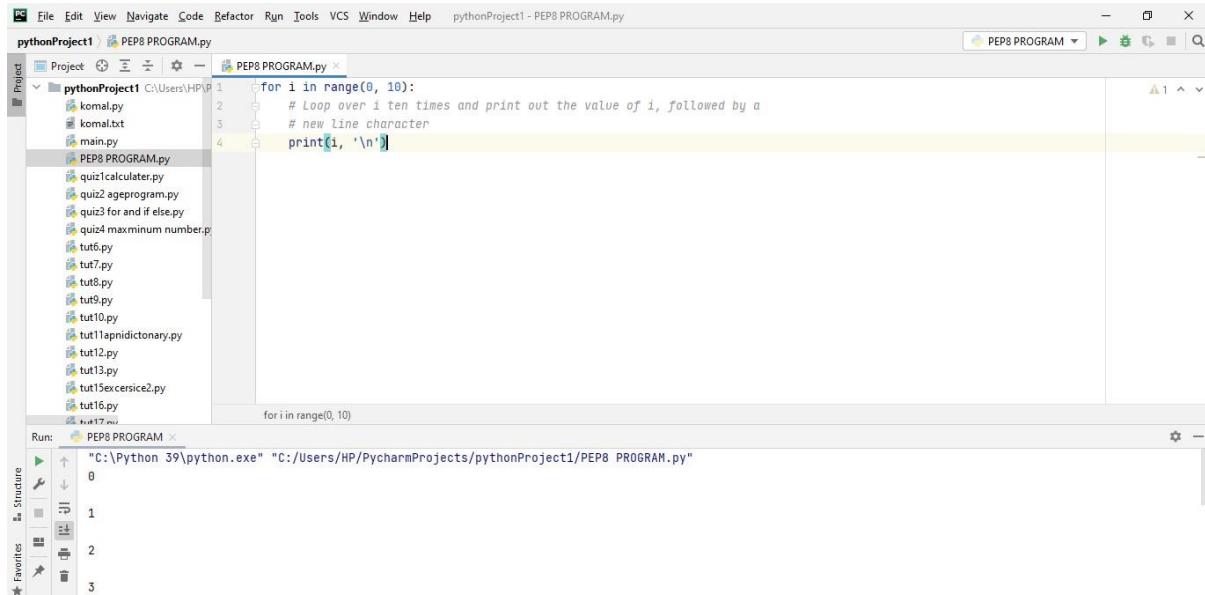
- Limit the line length of comments and docstrings to 72 characters.
- Use complete sentences, starting with a capital letter.
- Make sure to update comments if you change your code.

Block Comments

Use block comments to document a small section of code. They are useful when you have to write several lines of code to perform a single action, such as importing data from a file or updating a database entry. They are important as they help others understand the purpose and functionality of a given code block.

PEP 8 provides the following rules for writing block comments:

- Indent block comments to the same level as the code they describe.
- Start each line with a # followed by a single space.
- Separate paragraphs by a line containing a single #.



```
for i in range(0, 10):
    # Loop over i ten times and print out the value of i, followed by a
    # new line character
    print(i, '\n')
```

The screenshot shows the PyCharm IDE interface. The top menu bar includes File, Edit, View, Navigate, Code, Refactor, Run, Tools, VCS, Window, Help, and pythonProject1 - PEP8 PROGRAM.py. The Project tool window on the left lists files in the 'pythonProject1' directory, including 'PEP8 PROGRAM.py'. The main editor window displays the Python code. The bottom Run tool window shows the command: "C:\Python 39\python.exe" "C:/Users/HP/PycharmProjects/pythonProject1/PEP8 PROGRAM.py". The output pane shows the numbers 0 through 9, each followed by a new line character.

PEP 8 STYLE GUIDE

GROUP No. 8

ANNE VERONICA	1
PRIYA GUPTA	20
SHUBH PATEL	60
BHAVANA PRAJAPATI	66
POOJA PRAMANIK	67
HEETA TALAVIYA	92
GLORY LITHIYAL	95
NITESH GUPTA	111
AMAN UPADHYAY	132
DEEPAK KESHRI	134
SURYASEN VISHWAKARMA	139

PEP8

PEP8 is a style guide for python code.

- PEP stands for Python Enhancement Proposal, and they describe and document the way python language evolves.
- It was written in 2001 by Guido van Rossum, Barry Warsaw, and Nick Coghlan.
- A PEP is a document that describes new features proposed for Python and documents aspects of Python, like design and style, for the community.
- They also provide a reference point (and a standard) for the pythonic way to write code

→ It also has a lot of programming recommendations and useful tips on various topics, which aim to improve readability and reliability of your code.

→ PEP8 features:-

1. Plugin architecture: Adding new checks is easy.
2. Parseable output: Jump to error location in your editor.
3. Small: Just one Python file, requires only stdlib. You can use just the pep8.py file for this purpose.

Naming Conventions

Naming Conventions:

1.Variable

2.Function

3.Class

4.Method

5.Constant

6.Module

7.Package

\

Variable: A variable is created the moment you first assign a value to it

```
#Wrong Way to Initialize or assigning a name to a variable  
#Name Should not start with a number  
#Name should be intuitive and not too common.
```

```
1variable=2 #Variable name started with a number (Wrong Way)  
print(1variable)
```

```
File "<ipython-input-1-d1860915d72c>", line 5  
 1variable=2  
^
```

```
SyntaxError: invalid syntax
```

```
#Wrong Way to Initialize or assigning a name to a variable  
#Name Should not start with a number  
#Name should be intuitive and not too common.
```

```
x='Bhavana' #Variable name is too common and not intuitive (Not a Good Way)  
print(x)
```

```
Bhavana
```

```
#Wrong Way to Initialize or assigning a name to a variable  
#Name Should not start with a number  
#Name should be intuitive and not too common.
```

```
first_name='Bhavana' #Variable name is self-explanatory and has a readability, and it is seperated using underscores  
print(x)
```

```
Bhavana
```

Function: A function is a block of code which only runs when it is called.

```
#Wrong Way to Initialize or assigning a name to a function
#Name Should not start with a number
#Name should be intuitive and not too common.

def ^function(): #Function name should not be started with a Number or special characters
    print("Not a correct way to represent a function name")

^function()

File "<ipython-input-5-5f84f1733e34>", line 5
    def ^function():
    ^
SyntaxError: invalid syntax
```

```
#Wrong Way to Initialize or assigning a name to a function
#Name Should not start with a number
#Name should be intuitive and not too common.

def x(): #Function name is too generic and it can create a confusion in enterprise programming
    print("Function Name is too generic, you can use it but it is not recommended as it is not self-explanatory and intuitive")

x()

Function Name is too generic, you can use it but it is not recommended as it is not self-explanatory and intuitive
```

```
#Wrong Way to Initialize or assigning a name to a function
#Name Should not start with a number
#Name should be intuitive and not too common.

def display_function(): #Function name is self explanatory
    print("Function Name is self explanatory, name can be more intuitive in case of proper functionality")

display_function()

Function Name is self explanatory, name can be more intuitive in case of proper functionality
```

Class: class definitions begin with a class keyword.

```
#Wrong Way to Initialize or assigning a name to a class
#Name Should not start with a number
#Name should be intuitive and not too common.

1class x:
def display_function(): #Function name is self explanatory
    print("Function Name is self explanatory, name can be more intuitive in case of proper functionality")

display_function()

File <ipython-input-9-0547726683a1>, line 5
 1class x:
 ^
SyntaxError: invalid syntax
```

```
class Employee:
    def accept(self):
        print("Enter Id:")
        self.Id=int(input())
        print("Enter Name:")
        self.name= str(input())
    def display(self):
        print("ID: %d \nName: %s"%(self.Id,self.name))

emp=Employee()
emp.accept()
emp.display()

Enter Id:
66
Enter Name:
bhavana
ID: 66
Name: bhavana
```

Method: A Python method is a label that you can call on an object; it is a piece of code to execute on that object.

```
#Wrong Way to Initialize or assigning a name to a method
#Name Should not start with a number
#Name should be intuitive and not too common.
```

```
1class Method:
    def display(self):
        print("This is method function. ")

c = Method()
c.display()
```

```
File "<ipython-input-26-3e88b14da450>", line 6
 1class Method:
 ^
```

```
SyntaxError: invalid syntax
```

```
#Wrong Way to Initialize or assigning a name to a class
#Name Should not start with a number
#Name should be intuitive and not too common
```

```
class Product:
    def __init__(self):
        self.prod_id = input("Enter the Product ID: ")
        self.prod_name = input("Enter the Product Name: ")
        self.total_no = int(input("Enter the total no. of Items Purchase: "))
        self.unit_price=float(input("Enter the unit Price: "))
    def display(self):
        print("Total Price of %d units of Product %s is: %0.2f" %(self.total_no,self.prod_name,self.total_no*self.unit_price))

p1=Product()
p1.display()
```

```
Enter the Product ID: A20134
Enter the Product Name: Choclate
Enter the total no. of Items Purchase: 7
Enter the unit Price: 75.50
Total Price of 7 units of Product Choclate is: 528.50
```

Constant: A constant is a type of variable whose value cannot be changed.

```
pi = 3.14          #pi is constant
radius=5
print("Area of circle: %0.2f" %(pi*radius*radius))
```

Area of circle: 78.50

Modules: Modules refer to a file containing Python statements and definitions.

```
# to import standard module math

import math
print("The value of pi is", math.pi)
```

The value of pi is 3.141592653589793

Packages: A package is basically a directory with Python files and a file with the name `__init__.py`



Code layout

WITHOUT SPACE

These conventions lead to text that you can read easily, like this:

This would become increasingly hard to read. For example have a look at the example below

```
howwillitlookifwedonothavethespace
```

WITH SPACE

Now here, we will use space and write it in regular English language, so it will be very easy to read.

How will it look if we do not have the space

Maximum line length and line breaking

PEP 8 guidelines suggest that each line of code (as well as comment lines) should be 79 characters wide or less. This is a common standard that is also used in other languages including R.

CORRECT

```
# Perform some math
a = 1+2
b = 3+4
c = a+b

# Read in and plot some
precip_timeseries = pd.readcsv("precip-2019.csv")
precip_timeseries.plot()
```

#WRONG

```
#Perform some math and do some things
a=1+2
b=3+4
c=a+b
data=pd.readcsv("precip-2019.csv")
data.plot()
```

Should a line break Before or After a Binary Operator

Here, it's harder to see which variable is being added and which is subtracted.

WRONG

```
Total = (Number 1+
          Number 2-
          Number 3)
```

You can immediately see which variable is being added or subtracted, as the operator is right next to the variable being operated on.

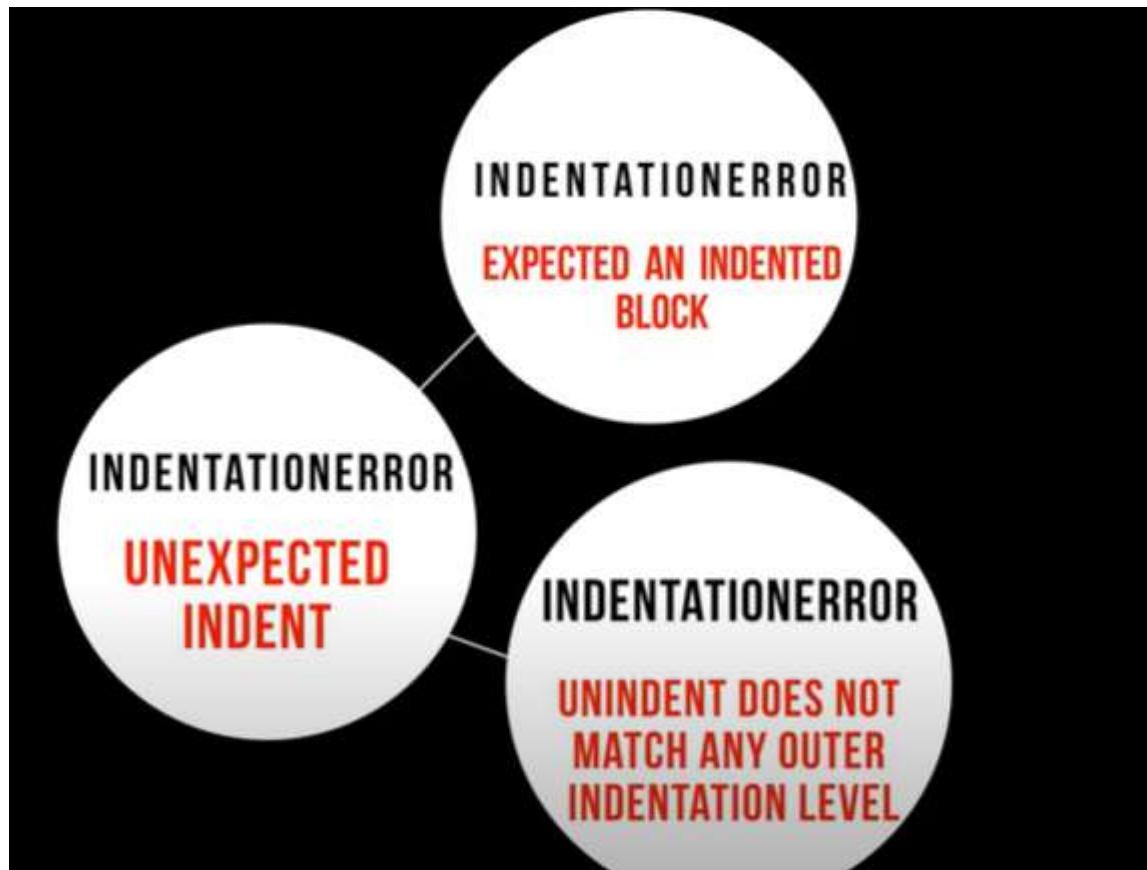
In the below Example

#CORRECT

```
Total = (Number 1
          + Number 2
          - Number 3)
```

Indentation

- Indentation is extremely important in Python.
- The Indentation level of lines of code in python determines how statements are grouped together.



1. Expected an indented block

```
x = 2
if x % 2 == 0:
    print("It is an even number")
```

```
File "<ipython-input-20-d2c95d58e212>", line 3
    print("It is an even number")
          ^
IndentationError: expected an indented block
```

```
x = 2
if x % 2 == 0:
    print("It is an even number")
```

```
It is an even number
```

2. Unexpected Indent



```
x = 2
if x % 2 == 0:
    print("It is an even number")
```



```
File "<ipython-input-24-a296ed44a7f2>", line 2
    if x % 2 == 0:
        ^
IndentationError: unexpected indent
```



```
x = 2
if x % 2 == 0:
    print("It is an even number")
```

It is an even number

3. Unindent does not match any outer indentation level

```
def greeting():
    print("Greetings of the day")
    return

greeting()

File "<ipython-input-30-698032a46f85>", line 3
    return
          ^
IndentationError: unindent does not match any outer indentation level
```

```
def greeting():
    print("Greetings of the day")
    return

greeting()
```

Greetings of the day

Tabs vs. Spaces

➤ Tabs vs. Spaces

The key indentation rules laid out by PEP 8 are the following:

- Use 4 consecutive spaces to indicate indentation.
- Prefer spaces over tabs.

➤ Indentation following line breaks

- Add a comment after the final condition. Due to syntax highlighting in most editors, this will separate the conditions from the nested code:

Not Recommended

```
x = 5  
if (x > 3 and  
    x < 10):  
    print(x)
```

Recommended

```
x = 5  
if (x > 3 and  
    x < 10):  
    # Both conditions satisfied  
    print(x)
```

```
x = 5  
if (x > 3 and  
    x < 10):  
    print(x)
```

Not Recommended

```
var = function(arg_one, arg_two,  
              arg_three, arg_four)
```

Recommended

```
var = function(  
              arg_one, arg_two,  
              arg_three, arg_four)
```

Not Recommended

```
def function(  
            arg_one, arg_two,  
            arg_three, arg_four):  
    return arg_one
```

Recommended

```
def function(  
            arg_one, arg_two,  
            arg_three, arg_four):  
    return arg_one
```

➤ Where to put the closing Braces

Not Recommended

```
▶ list_of_numbers = [ 1, 2, 3,  
                      4, 5, 6,  
                      7, 8, 9]
```

1. Method

```
▶ list_of_numbers = [  
                      1, 2, 3,  
                      4, 5, 6,  
                      7, 8, 9  
]
```

2. Method

```
▶ list_of_numbers = [  
                      1, 2, 3,  
                      4, 5, 6,  
                      7, 8, 9  
]
```

COMMENTS:

Comments are lines that exist in computer programs that are ignored by compilers and interpreters.

Comment begins with a hash mark (#)

Generally, comment looks like this:

this a comment

Because comment does not execute ,when you will run program you will not see any indication of the comment there.

BLOCK COMMENTS:

Each line of block comments starts with a # and a single space

Paragraphs inside a block comment are separated by a line containing a single #.

Anti-pattern

```
#This comment needs a space
def print_name(self):
    print(self.name)
```

Best practice

```
# Comment is correct now
def print_name(self):
    print(self.name)
```

INLINE COMMENTS:

Inline comment should be separated by at least two spaces from the comment.

They should start with a # and a single space

Inline comments are unnecessary and in fact distracting if they state the obvious

Anti-pattern

```
def print_name(self):  
    print(self.name) #This comment needs a space
```

Best practice

```
def print_name(self):  
    print(self.name) # Comment is correct now
```

DOCSTRING COMMENTS:

A docstring is added as a comment string right below the function,module,or object

RULES:

A docstring is either a single line, or a multi-line comment

In latter case, the first line is short description, and after the first line an empty line follows

This is a basic example of what it looks like:

```
def add(value1, value2):
    """Calculate the sum of value1 and value2."""
    return value1 + value2
```

In the Python interactive help system, the docstring is then made available via the `__doc__` attribute.

```
>>> print add.__doc__
Calculate the sum of value1 and value2.
```

Inline Comments vs Block Comments

Inline comments look like this

```
x = x + 1          # Compensate for border
```

While block comments look like this

```
# Compensate for border. These comments  
# often cover multiple lines.  
x = x + 1
```

Whitespace in Expressions and Statements

1) Whitespace Around Binary Operators

Surround the following binary operators with a single space on either side:

- Assignment operators (=, +=, -=, and so forth)
- Comparisons (==, !=, >, <, >=, <=) and (is, is not, in, not in)
- Booleans (and, not, or)

Note: When = is used to assign a default value to a function argument, do not surround it with spaces.

Python

```
# Recommended
def function(default_parameter=5):
    # ...

# Not recommended
def function(default_parameter = 5):
    # ...
```

- Adding space when there is more than one operator in a statement.

Python

```
# Recommended  
y = x**2 + 5  
z = (x+y) * (x-y)
```

```
# Not Recommended
```

```
y = x ** 2 + 5  
z = (x + y) * (x - y)
```

- Adding space to if statements where there are multiple conditions.

Python

```
# Not recommended  
if x > 5 and x % 2 == 0:  
    print('x is larger than 5 and divisible by 2!')
```

Python

```
# Recommended  
if x>5 and x%2==0:  
    print('x is larger than 5 and divisible by 2!')
```

Note : Use the same amount of whitespace either side of the operator.

The following is not acceptable :

Python

```
# Definitely do not do this!
if x >5 and x% 2== 0:
    print('x is larger than 5 and divisible by 2!')
```

When to Avoid Adding Whitespace

- Trailing space
- Immediately inside parentheses, brackets, or braces:

Python

```
# Recommended
my_list = [1, 2, 3]

# Not recommended
my_list = [ 1, 2, 3, ]
```

- Before a comma, semicolon, or colon:

Python

```
x = 5
y = 6

# Recommended
print(x, y)

# Not recommended
print(x , y)
```

Before the open parenthesis that starts the argument list of a function call:

Python

```
def double(x):
    return x * 2

# Recommended
double(3)

# Not recommended
double (3)
```

Before the open bracket that starts an index or slice:

Python

```
# Recommended
list[3]

# Not recommended
list [3]
```

- Between a trailing comma and a closing parenthesis:

```
Python
```

```
# Recommended  
tuple = (1,
```

```
# Not recommended  
tuple = (1, )
```

- To align assignment operators:

```
Python
```

```
# Recommended  
var1 = 5  
var2 = 6  
some_long_var = 7
```

```
# Not recommended  
var1      = 5  
var2      = 6  
some_long_var = 7
```

Programming Recommendations

❖ Two Programming Recommendations by PEP-8

A) # Not recommended

```
my_bool = 6 > 5  
if my_bool == True:  
    return '6 is bigger than 5'
```

B) # Recommended

```
if my_bool:  
    return '6 is bigger than 5'
```

In the above program B is recommended over A by the PEP-8

C) # Not recommended

```
my_list = []  
if not len(my_list):  
    print('List is empty!')
```

```
D) # Recommended  
my_list = []  
if not my_list:  
    print('List is empty!')
```

In the above program D is recommended over C by the PEP-8

Q. When to Ignore PEP-8

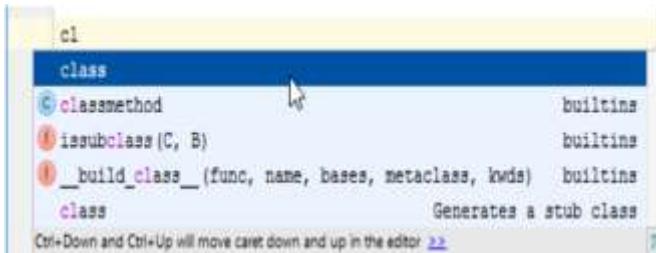
ANSWER: NEVER

Though, there are some guidelines in PEP-8 that are inconvenient in some instances:

- Complying with PEP-8
 - Code surrounding
 - Code compatibility

Tips and Tricks to Help Ensure Your Code Follows PEP 8

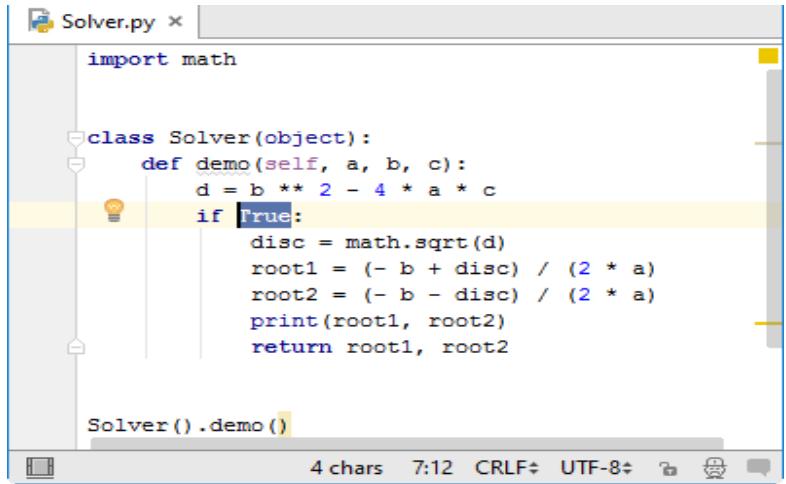
Highlighting code style violations:



(Refer to Code Completion page of the product documentation for details.)

Generating Source code:

Select `if` option from the suggestion list. As you see, PyCharm automatically adds `if True:` and indents the selected lines:



```
import math

class Solver(object):
    def demo(self, a, b, c):
        d = b ** 2 - 4 * a * c
        if True:
            disc = math.sqrt(d)
            root1 = (- b + disc) / (2 * a)
            root2 = (- b - disc) / (2 * a)
            print(root1, root2)
        return root1, root2

Solver().demo()
```

Linter-python-pep8 package

This linter-python-pep8 plugin or Linter provides an interface to pep8. It will be used with files that have the Python syntax.

Installation:

Before using this plugin, you should make sure that pep8 is installed on your system. You can follow following instructions to install pep8:

Install python.

Install pep8 by typing the following in a terminal:

```
pip install pep8
```

Black

Black can be installed by running pip install black. It requires Python 3.6.0+ to run. Once Black is installed, you will have a new command-line tools called black available to you in your shell, and you're ready to start.

```
$ pip install black
```

Format a Single File:

Let's look at this simple example: here are my two python functions in my python file called sample_code.py.

```
def add(a,      b):
    answer = a + b
    return answer

def sub(c,      d):
    answer = c - d
    return answer
```

You can use `black sample_code.py` in the terminal to change the format. After running Black, you will see the following output:

```
reformatted sample_code.py
All done! ✨✨
1 file reformatted.
```

Then you open `sample_code.py` to see formatted python code:

```
def add(a, b):
    answer = a + b

    return answer

def sub(c, d):
    answer = c - d

    return answer
```

Example of code and layout.

With space and without space.

WITH SPACE.

*ex-

MY NAME IS NITESH

WITHOUT SPACE.

*ex-

MYNAMEISNITESH

Maximum line length and line breaking.

Recommended

Python

- Ex-

```
def function(arg_one, arg_two,
             arg_three, arg_four):
    return arg_one
```

Not Recommended

Python

- Ex-

```
from mypkg import example1, \
example2, example3
```

Should a line break Before or After A Binary Operator.

Python

- Ex-

```
# Recommended
total = (first_variable
         + second_variable
         - third_variable)
```

Python

- Ex-

```
# Not Recommended
total = (first_variable +
         second_variable -
         third_variable)
```

Example of comments.

block comment.

Anti-pattern.

Example

```
#This is a comment  
print("Hello, World!")
```

Best-practice.

Example

```
#This is a comment  
#written in  
#more than just one line  
print("Hello, World!")
```

Inline comments.

Anti-pattern.

Python

```
x = 5 # This is an inline comment
```

Best practice.

Python

```
x = 'John Smith' # Student Name
```

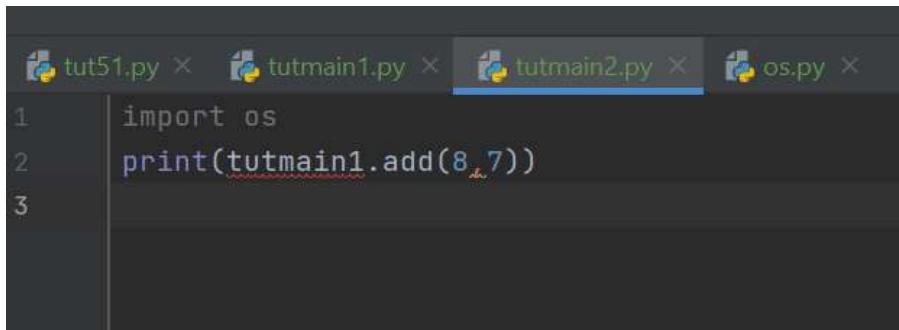
Documentation string comment.

```
"""Return a foobang  
  
Optional plotz says to  
frobnicate the bizbaz first.  
"""
```

EXAMPLE OF NAMING CONVENTION

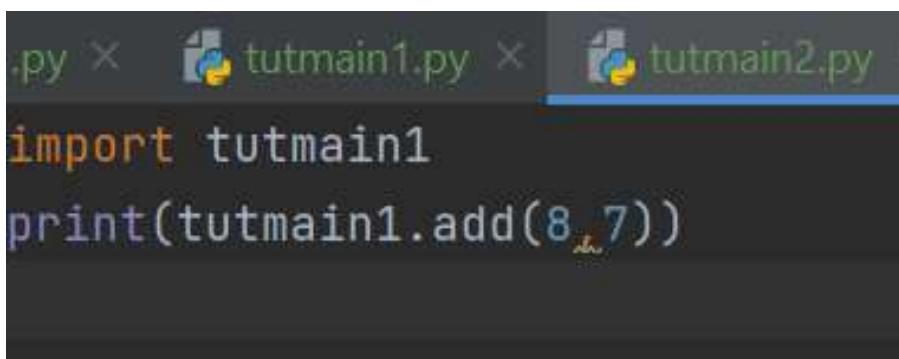
NAMING MODULE WITH HELP OF PEP8

Not recommended



```
tut51.py × tutmain1.py × tutmain2.py × os.py ×  
1 import os  
2 print(tutmain1.add(8,7))  
3
```

Recommended



```
.py × tutmain1.py × tutmain2.py ×  
import tutmain1  
print(tutmain1.add(8,7))
```

NAMING VARIABLE WITH HELP OF PEP8

Variable:

```
>>> # Not recommended  
>>> x = 'John Smith'  
>>> y, z = x.split()  
>>> print(z, y, sep=', ')  
'Smith, John'
```

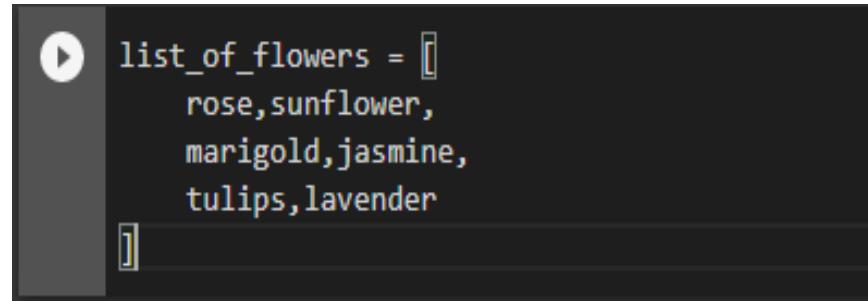
>>>

```
>>> # Recommended  
>>> name = 'John Smith'  
>>> first_name, last_name = name.split()  
>>> print(last_name, first_name, sep=', ')  
'Smith, John'
```

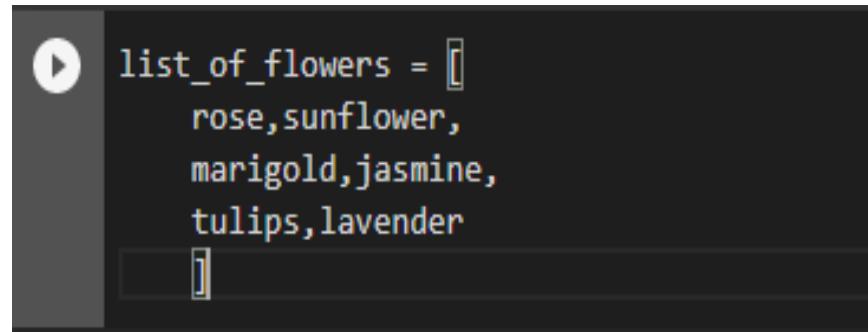
⊕ EXAMPLES OF INDENTATION

❖ METHODS OF WHERE TO PUT CLOSING BRACES:-

Recommended



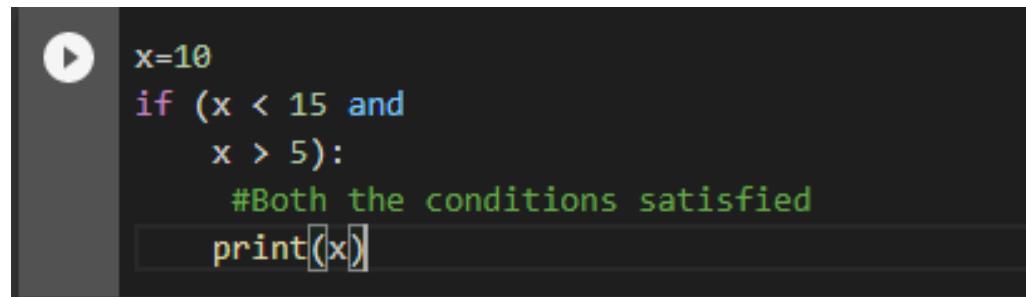
```
list_of_flowers = [rose,sunflower,  
                   marigold,jasmine,  
                   tulips,lavender]  
[]
```



```
list_of_flowers = [rose,sunflower,  
                   marigold,jasmine,  
                   tulips,lavender]  
[]
```

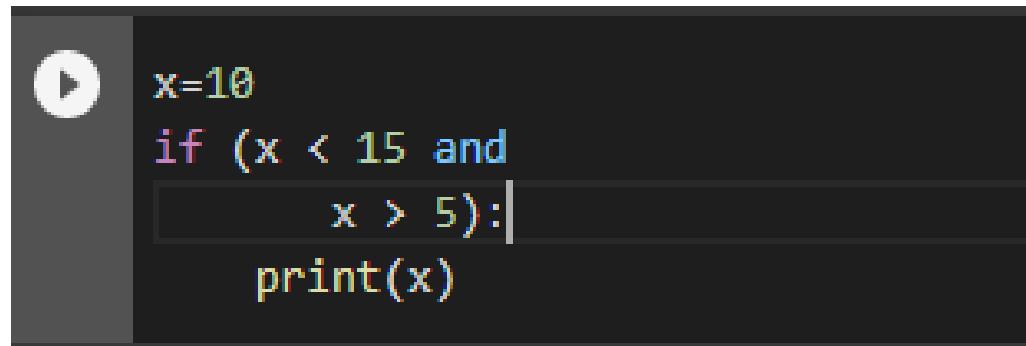
❖ Methods for following line breaks

Recommended



A screenshot of a code editor interface. On the left is a dark sidebar with a white play button icon. The main area has a dark background with light-colored text. It contains the following Python code:

```
x=10
if (x < 15 and
    x > 5):
    #Both the conditions satisfied
    print([x])
```



A screenshot of a code editor interface. On the left is a dark sidebar with a white play button icon. The main area has a dark background with light-colored text. It contains the following Python code:

```
x=10
if (x < 15 and
    x > 5):
    print(x)
```

EXAMPLE OF WHITESPACING

1. Adding space when there is more than one operator in a statement.



```
#recommended  
b = a**8 + 5  
c = (a+b) * (a-b)
```



```
#not recommended  
b = a ** 8 + 5  
c = (a + b) * (a - b)
```

1. Adding space to if statements where there are multiple conditions.

#Recommended

```
if x>8 and x%2== 0:  
    print('x is larger than 8 and divisible by 2!')
```

#not Recommended

```
if x > 8 and x % 2 == 0:  
    print('x is larger than 8 and divisible by 2!')
```

THANK YOU