



**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: Ms. Priya Umesh Gupta

Roll No: 20 Programme: BSc IT 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 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)

Name: - Priya Gupta

Roll No:- 20

Sr. No.	DATE	TITLE	SIGN
1.	02/02/2021	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.	09/02/2021	Creating account, repository on GitHub and Cloning repository in GitHub Page	
3.	16/02/2021	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.	23/02/2021	WRITING EMAIL	
5.	25/02/2021	Using practical examples, describe green computing. List and explain the steps that you take to contribute to green computing	
6.	02/03/2021	WRITING BLOGS	
7.	09/03/2021	Implementing coding practices in Python using PEP8.	
8.	18/03/2021	PRESENTATION: PEP 8 STYLE GUIDE	

NAME: Priya Gupta

ROLL NO.: 20

CLASS: FYIT

1. Description about Wikipedia and its features

Wikipedia is a free, open content online encyclopedia created through **the** collaborative effort of a community of users known as Wikipedians. Anyone registered on **the** site can create an article for publication; registration is not required to edit articles. **Wikipedia** was **the** only non-commercial site of **the** top ten.

You may not have realized, but you've probably already used a **Wiki**. The most famous example most people have engaged with is **Wikipedia**, the free internet encyclopedia that anyone can edit.

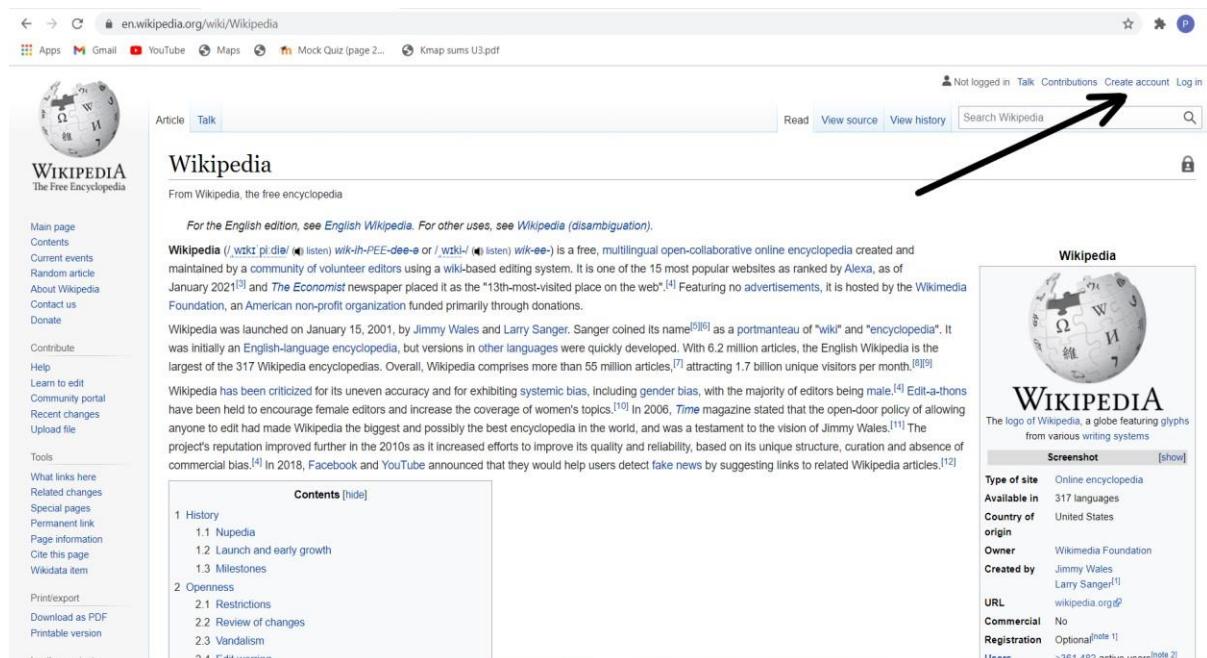
Link between pages

- Full page search.
- Hierarchical page **display**.
- A list of the most recently edited pages.

Searching with tags.

a) Creating Account on Wikipedia

Step 1: To create your account click on create account.



The screenshot shows the English Wikipedia homepage. At the top right, there is a navigation bar with links for 'Not logged in', 'Talk', 'Contributions', 'Create account', and 'Log in'. An arrow points to the 'Create account' link. The main content area features the Wikipedia logo, the title 'Wikipedia', and a brief introduction. On the left side, there is a sidebar with various links like 'Main page', 'Contents', 'Current events', etc. The right side has a sidebar with information about the Wikipedia logo and a 'Screenshot' section with details such as 'Type of site: Online encyclopedia', 'Available in: 317 languages', and 'Owner: Wikimedia Foundation'.

NAME: Priya Gupta

ROLL NO.: 20

CLASS: FYIT

Step 2: Put Unique Username Here.

The screenshot shows the 'Create account' page on Wikipedia. The 'Username' input field is highlighted with a red arrow. To the right of the form, there's a summary of Wikipedia statistics: 1,002,153,299 edits, 6,249,600 articles, and 148,564 recent contributors. Below the statistics are icons for a pen, a book, and two people.

en.wikipedia.org/w/index.php?title=Special>CreateAccount&returnto=Wikipedia

Random article About Wikipedia Contact us Donate Contribute Help Learn to edit Community portal Recent changes Upload file Tools Upload file Special pages Printable version Languages

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

mendspedro

Enter the text you see on the image

Can't see the image? Request an account

Create your account

1,002,153,299 edits
6,249,600 articles
148,564 recent contributors

Step 3 : create your password and confirm it once.

The screenshot shows the 'Create account' page on Wikipedia. The 'Password' input field is highlighted with a red arrow. To the right of the form, there's a summary of Wikipedia statistics: 1,002,153,299 edits, 6,249,600 articles, and 148,564 recent contributors. Below the statistics are icons for a pen, a book, and two people.

en.wikipedia.org/w/index.php?title=Special>CreateAccount&returnto=Wikipedia

Random article About Wikipedia Contact us Donate Contribute Help Learn to edit Community portal Recent changes Upload file Tools Upload file Special pages Printable version Languages

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

mendspedro

Enter the text you see on the image

Can't see the image? Request an account

Create your account

1,002,153,299 edits
6,249,600 articles
148,564 recent contributors

Step 4: Enter same text which is seen in the table

NAME: Priya Gupta

ROLL NO.: 20

CLASS: FYIT

The screenshot shows the Wikipedia account creation form. On the left, there's a sidebar with links like Random article, About Wikipedia, and Help. The main form has fields for Username, Password, Confirm password, and Email address (optional). Below these is a CAPTCHA section with an image containing the text "mendspedro" and a text input field labeled "Enter the text you see on the image". A large black arrow points from the text "Step 5: click on Create account to create the account." down to the "Create your account" button at the bottom of the form.

en.wikipedia.org/w/index.php?title=Special>CreateAccount&returnto=Wikipedia

Random article About Wikipedia Contact us Donate Contribute Help Learn to edit Community portal Recent changes Upload file Tools Upload file Special pages Printable version Languages

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

mendspedro

Enter the text you see on the image

Can't see the image? Request an account

Create your account

1,002,153,299 edits

6,249,600 articles

148,564 recent contributors

Step 5: click on Create account to create the account.

This screenshot is identical to the one above, showing the account creation form on Wikipedia. It includes the sidebar, form fields, and CAPTCHA section. A large black arrow points from the text "Step 5: click on Create account to create the account." down to the "Create your account" button at the bottom of the form.

en.wikipedia.org/w/index.php?title=Special>CreateAccount&returnto=Wikipedia

Random article About Wikipedia Contact us Donate Contribute Help Learn to edit Community portal Recent changes Upload file Tools Upload file Special pages Printable version Languages

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

mendspedro

Enter the text you see on the image

Can't see the image? Request an account

Create your account

1,002,153,299 edits

6,249,600 articles

148,564 recent contributors

NAME: Priya Gupta

ROLL NO.: 20

CLASS: FYIT

a) Creating your page on Wikipidea

Step 1: Click on view source

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

Submit an edit request

Step 2: Click on Submit an edit request for edit and create a page.

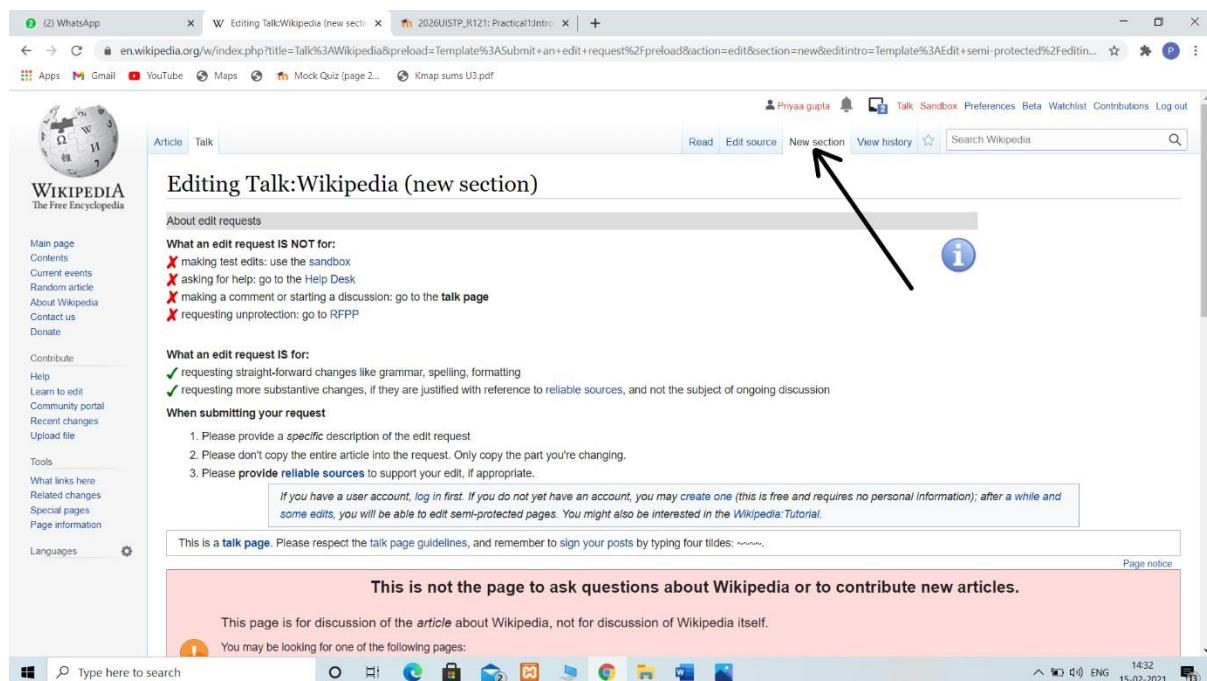
Submit an edit request

Step 3: Click on New section and here we v=can create our page

NAME: Priya Gupta

ROLL NO.: 20

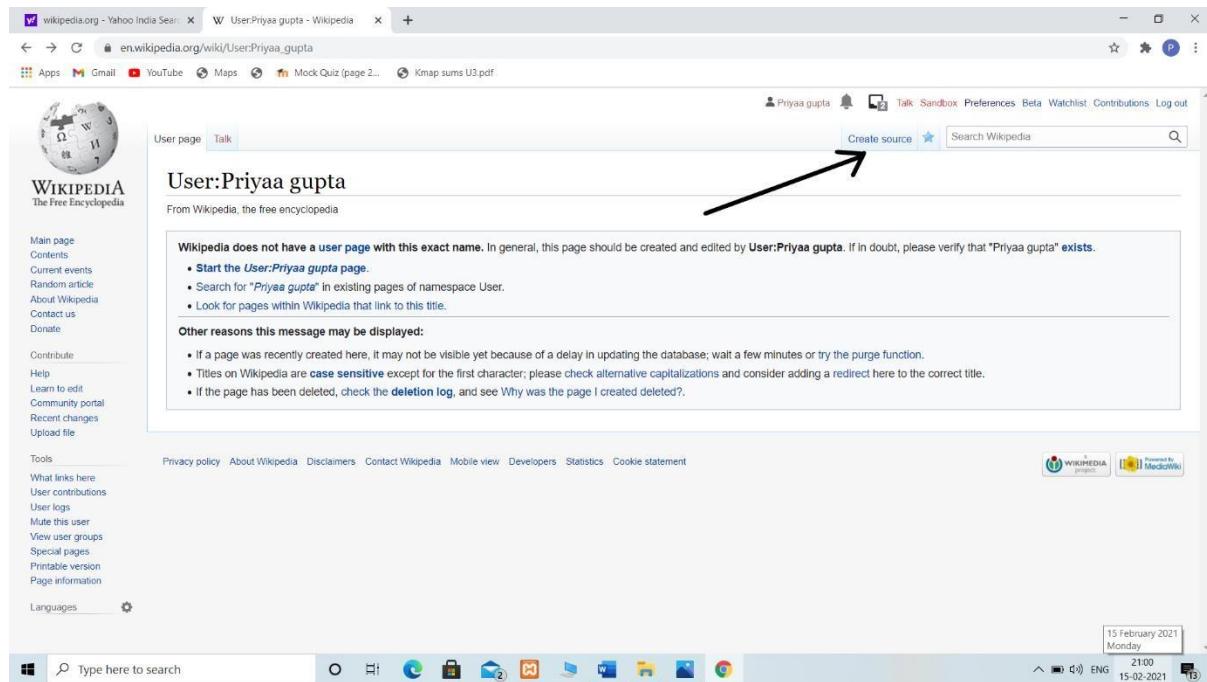
CLASS: FYIT



The screenshot shows a Microsoft Edge browser window with three tabs open: 'WhatsApp', 'Editing Talk:Wikipedia (new section)', and '2026UISTP_R121: Practical1Intro'. The main content area displays the 'Editing Talk:Wikipedia (new section)' page. At the top, there are tabs for 'Article' and 'Talk'. Below them is a navigation bar with links for 'Read', 'Edit source', 'New section' (which has a black arrow pointing to it), 'View history', and 'Search Wikipedia'. The 'New section' link is highlighted with a blue background. To the right of the navigation bar is a search bar and a magnifying glass icon. The page content includes sections for 'About edit requests', 'What an edit request IS NOT for', 'What an edit request IS for', 'When submitting your request', and a note about talk pages. A large red box at the bottom states: 'This is not the page to ask questions about Wikipedia or to contribute new articles.' Below this, a smaller note says: 'This page is for discussion of the article about Wikipedia, not for discussion of Wikipedia itself.' A status bar at the bottom of the screen shows the date and time as 15-02-2021 14:32.

b) Editing your page on Wikipedia

Step 1: Click on Create source button



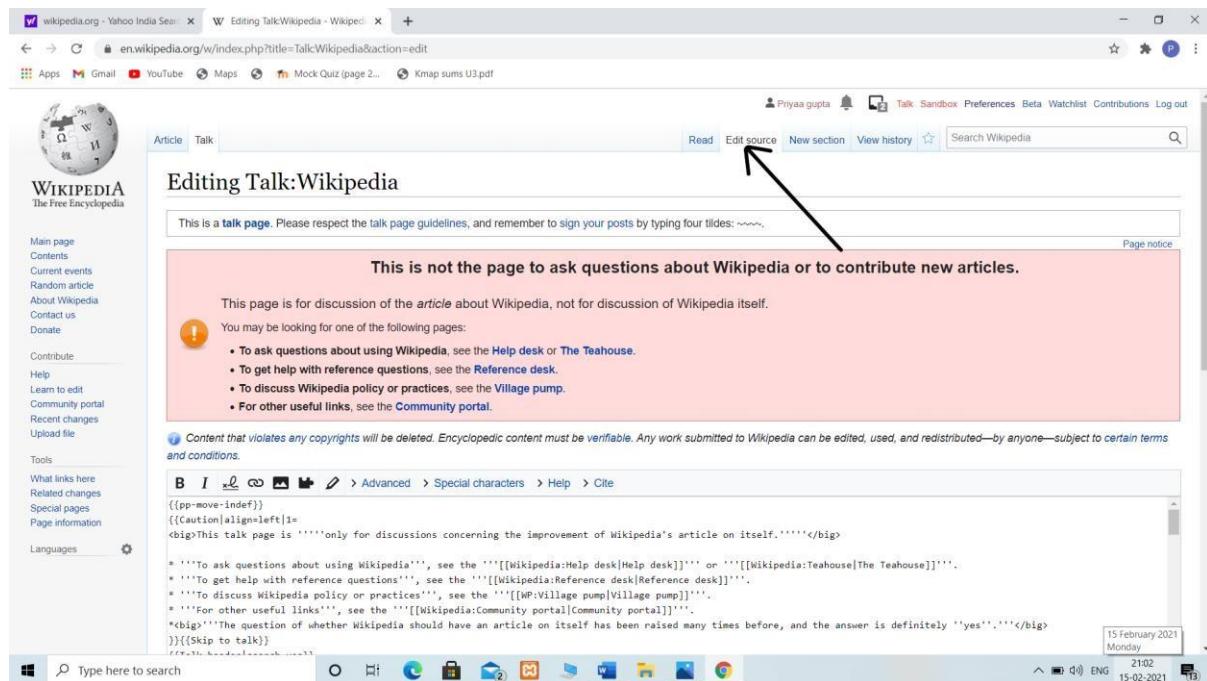
The screenshot shows a Microsoft Edge browser window with three tabs open: 'wikipedia.org - Yahoo India Search', 'User:Priyaa gupta - Wikipedia', and 'en.wikipedia.org/wikilinks'. The main content area displays the 'User:Priyaa gupta' page. At the top, there are tabs for 'User page' and 'Talk'. Below them is a navigation bar with links for 'Priyaa gupta', 'Talk', 'Sandbox', 'Preferences', 'Beta', 'Watchlist', 'Contributions', and 'Log out'. The 'Create source' link is highlighted with a blue background and has a black arrow pointing to it. The page content includes a note about the user page name and other reasons why the message might be displayed. A status bar at the bottom of the screen shows the date and time as 15 February 2021 Monday 21:00 15-02-2021.

Step 2: Go to the User page and edit your page.

NAME: Priya Gupta

ROLL NO.: 20

CLASS: FYIT



wikipedia.org - Yahoo India Search | W Editing Talk:Wikipedia - Wikipedia | +

en.wikipedia.org/w/index.php?title=Talk:Wikipedia&action=edit

Apps Gmail YouTube Maps Mock Quiz (page 2...) Kmap sums U3.pdf

Priyaa gupta Talk Preferences Beta Watchlist Contributions Log out

WIKIPEDIA The Free Encyclopedia

Main page Contents Current events Random article About Wikipedia Contact us Donate Contribute Help Learn to edit Community portal Recent changes Upload file Tools What links here Related changes Special pages Page information Languages

Article Talk Read Edit source New section View history Search Wikipedia

Editing Talk:Wikipedia

This is a talk page. Please respect the talk page guidelines, and remember to sign your posts by typing four tildes: ~~~~.

This is not the page to ask questions about Wikipedia or to contribute new articles.

This page is for discussion of the article about Wikipedia, not for discussion of Wikipedia itself.

You may be looking for one of the following pages:

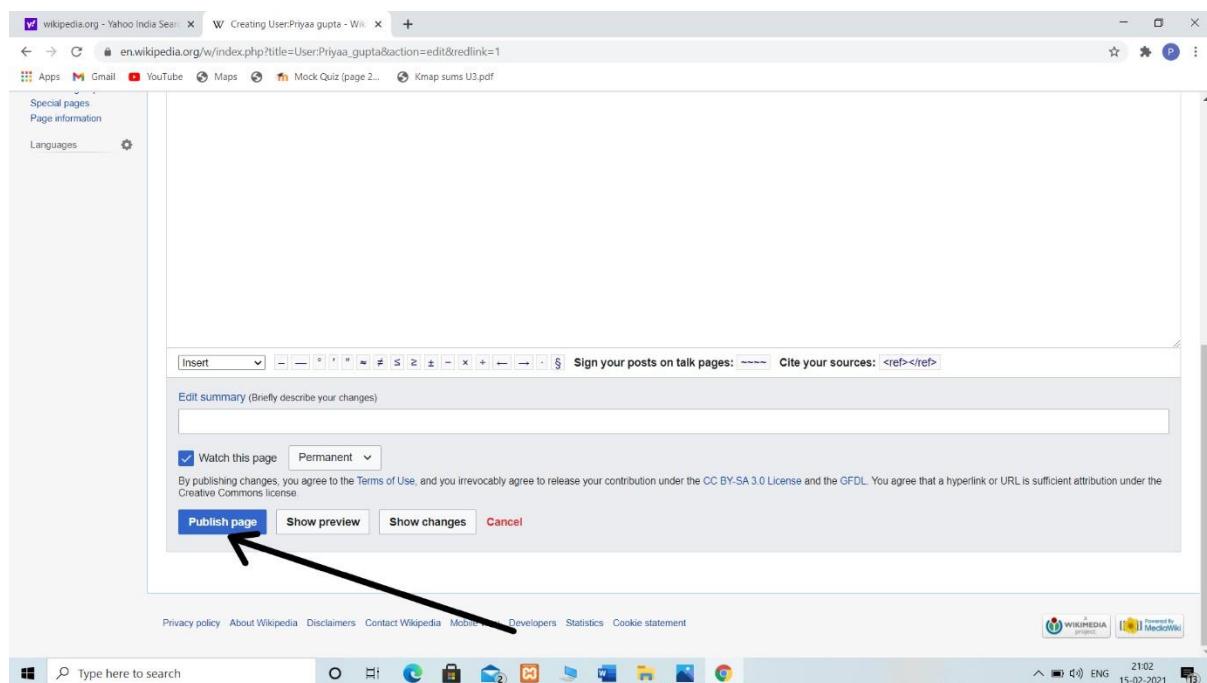
- To ask questions about using Wikipedia, see the [Help desk](#) or [The Teahouse](#).
- To get help with reference questions, see the [Reference desk](#).
- To discuss Wikipedia policy or practices, see the [Village pump](#).
- For other useful links, see the [Community portal](#).

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.

B I ~~L~~ ∞ Advanced > Special characters > Help > Cite

15 February 2021 Monday 2102 15-02-2021

Step 3: By clicking on Publish changes you can Publish your page.



wikipedia.org - Yahoo India Search | W Creating UserPriyaa gupta - Wikipedia | +

en.wikipedia.org/w/index.php?title=User:Priyaa_gupta&action=edit&redlink=1

Apps Gmail YouTube Maps Mock Quiz (page 2...) Kmap sums U3.pdf

Special pages Page information Languages

Sign your posts on talk pages: ~~~~ Cite your sources: <ref></ref>

Edit summary (Briefly describe your changes)

Watch this page Permanent

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.

Publish page Show preview Show changes Cancel

Privacy policy About Wikipedia Disclaimers Contact Wikipedia Mobile Developers Statistics Cookie statement

WIKIMEDIA project Powered by MediaWiki

Type here to search 2102 15-02-2021

We can read our page by clicking on read.

NAME: Priya Gupta

ROLL NO.: 20

CLASS: FYIT

wikipedia.org - Yahoo India Search | en.wikipedia.org - Wikipedia | 2026UISTP_R121:Practical1Intro | +

Priya gupta Talk Sandbox Preferences Beta Watchlist Contributions Log out

Article Talk Read View source View history Search Wikipedia

Wikipedia

From Wikipedia, the free encyclopedia

For the English edition, see [English Wikipedia](#). For other uses, see [Wikipedia \(disambiguation\)](#).

Wikipedia (/*wɪkɪ/ pl. /wɪkɪz/ or /wɪkɪ/ /wɪkɪ/) is a free, multilingual open-collaborative online encyclopedia created and maintained by a community of volunteer editors using a wiki-based editing system. It is one of the 15 most popular websites as ranked by Alexa, as of January 2021^[3] and *The Economist* newspaper placed it as the "13th-most-visited place on the web".^[4] Featuring no advertisements, it is hosted by the Wikimedia Foundation, an American non-profit organization funded primarily through donations.*

Wikipedia was launched on January 15, 2001, by Jimmy Wales and Larry Sanger. Sanger coined its name^{[5][6]} as a portmanteau of "wiki" and "encyclopedia". It was initially an English-language encyclopedia, but versions in other languages were quickly developed. With 6.2 million articles, the English Wikipedia is the largest of the 317 Wikipedia encyclopedias. Overall, Wikipedia comprises more than 55 million articles,^[7] attracting 1.7 billion unique visitors per month.^{[8][9]}

Wikipedia has been criticized for its uneven accuracy and for exhibiting systemic bias, including gender bias, with the majority of editors being male.^[4] Edit-a-thons have been held to encourage female editors and increase the coverage of women's topics.^[10] In 2006, *Time* magazine stated that the open-door policy of allowing anyone to edit had made Wikipedia the biggest and possibly the best encyclopedia in the world, and was a testament to the vision of Jimmy Wales.^[11] The project's reputation improved further in the 2010s as it increased efforts to improve its quality and reliability, based on its unique structure, curation and absence of commercial bias.^[12] In 2018, Facebook and YouTube announced that they would help users detect fake news by suggesting links to related Wikipedia articles.^[12]

Contents [hide]

- 1 History
 - 1.1 Nupedia
 - 1.2 Launch and early growth
 - 1.3 Milestones
- 2 Openness
 - 2.1 Restrictions
 - 2.2 Review of changes
 - 2.3 Vandalism
 - 2.4 Edit warning

Screenshot [show]

Type of site Online encyclopedia

Available in 317 languages

Country of origin United States

Owner Wikimedia Foundation

Created by Jimmy Wales
Larry Sanger^[13]

URL wikipedia.org^[14]

Commercial No

Registration Optional^[note 1]

Users >361,482 active users^[note 2]

15-02-2021 2133 14

NAME: Priya Gupta

Roll No.: 20

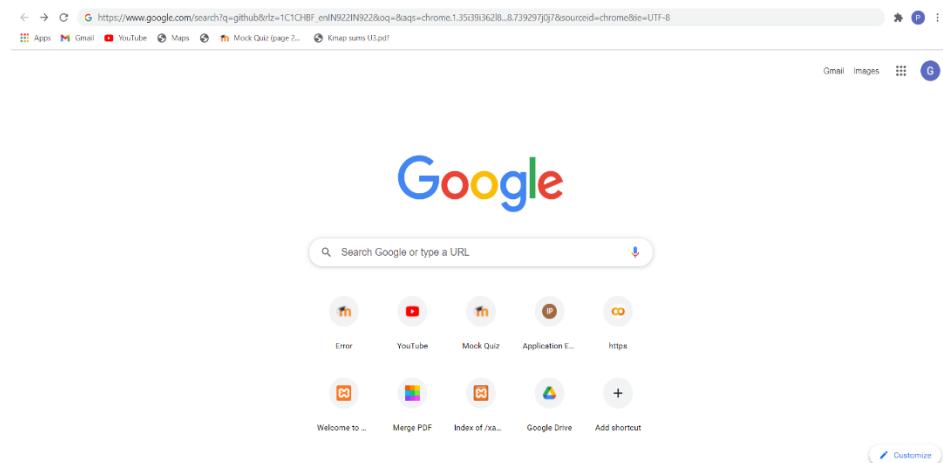
Class: FYIT

Practical 2

Creating account, repository on Github and Cloning repository in Github.

a) Creating Account:

Step 1: Go to <https://github.com/join> in a web browser. You can use any web browser on your computer, phone or tablet to join.



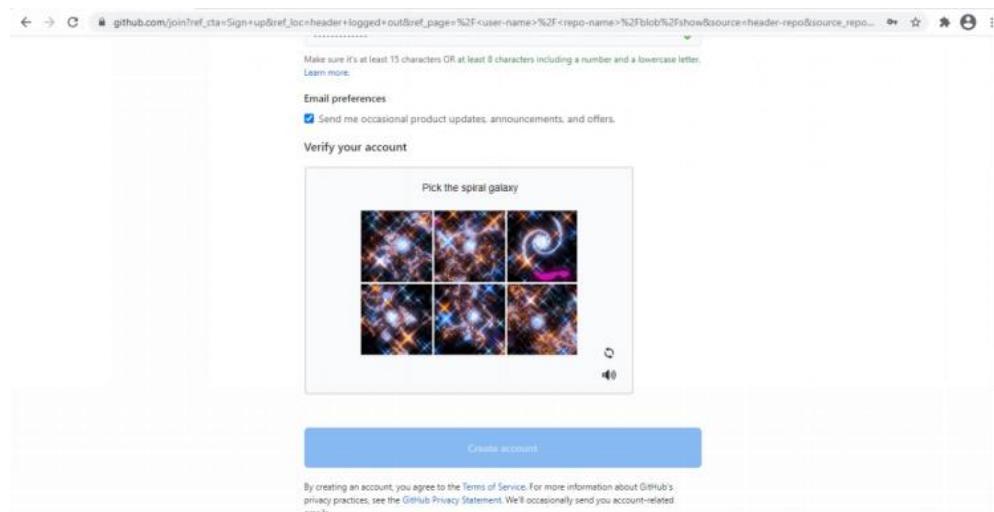
Step 2: Enter your personal details and complete the CAPTCHA puzzle.

A screenshot of the GitHub 'Create your account' form. The URL in the address bar is github.com/join. The main heading is 'Create your account'. A pink box at the top says 'There were problems creating your account.' Below it, there are three input fields: 'Username' (priyaagupta), 'Email address' (ugupta9363@gmail.com), and 'Password' (*****). Below the password field is a note: 'Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter.' A 'Learn more.' link is provided. Under 'Email preferences', there is a checked checkbox for 'Send me occasional product updates, announcements, and offers.' At the bottom, there is a 'Verify your account' section with a CAPTCHA image labeled 'Pick the spiral galaxy' showing a grid of four spiral galaxy icons.

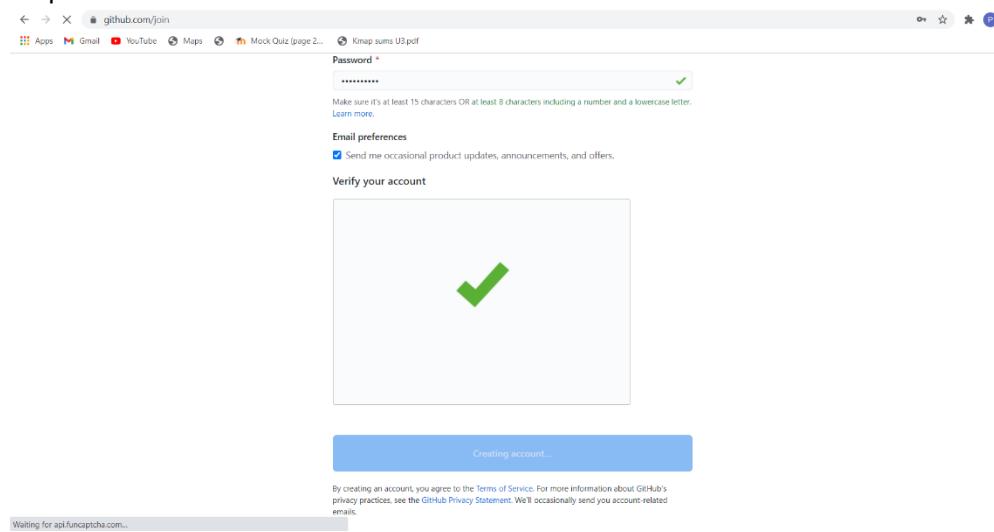
NAME: Priya Gupta

Roll No.: 20

Class: FYIT



Step 3: Click the 'Create account' button below the form.



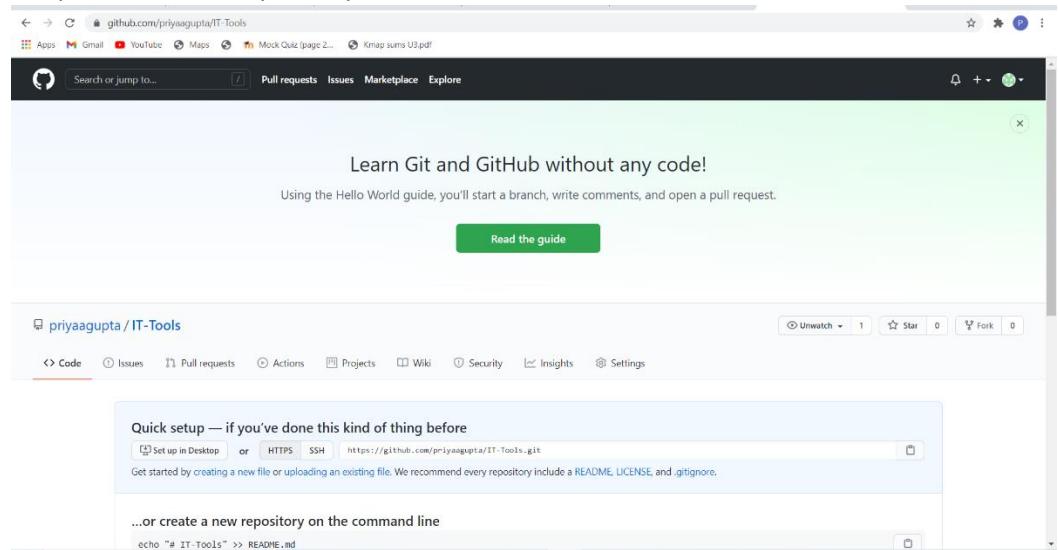
NAME: Priya Gupta

Roll No.: 20

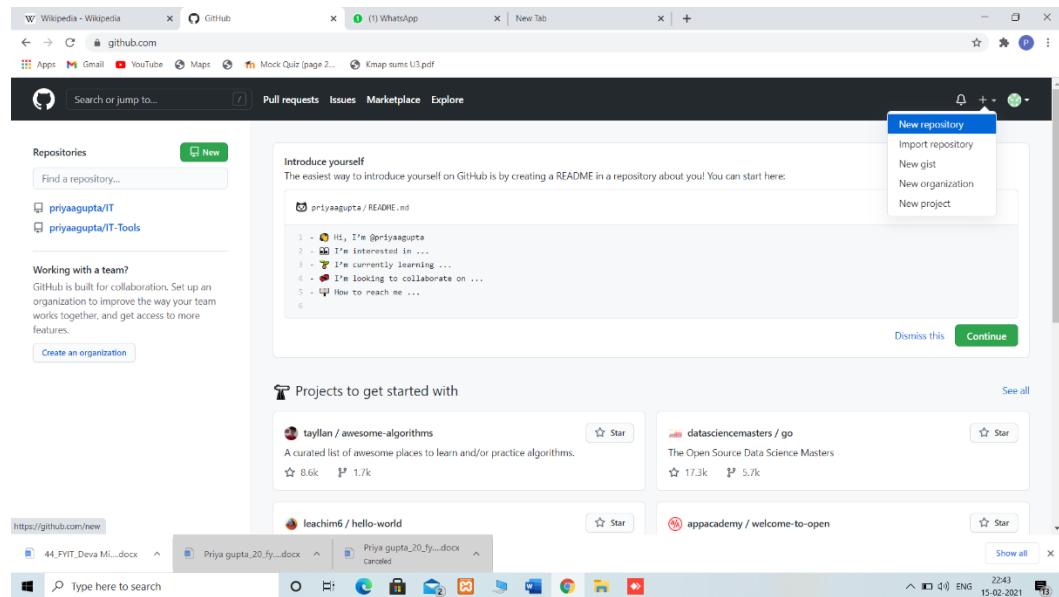
Class: FYIT

b) Creating Repository:

Step 1: To create repository click on + Button.



Step 2: Click on new repository.

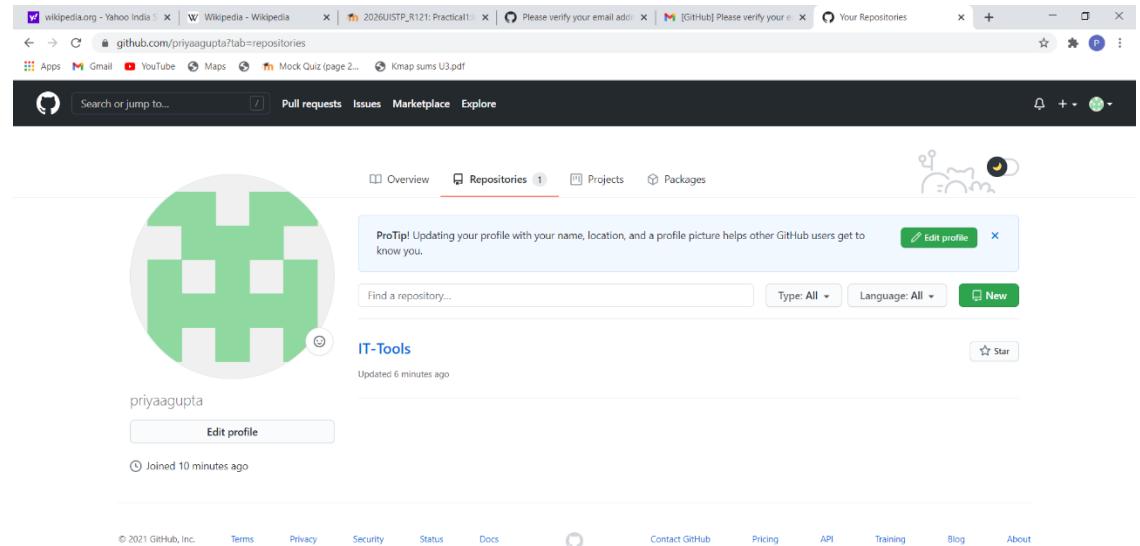


NAME: Priya Gupta

Roll No.: 20

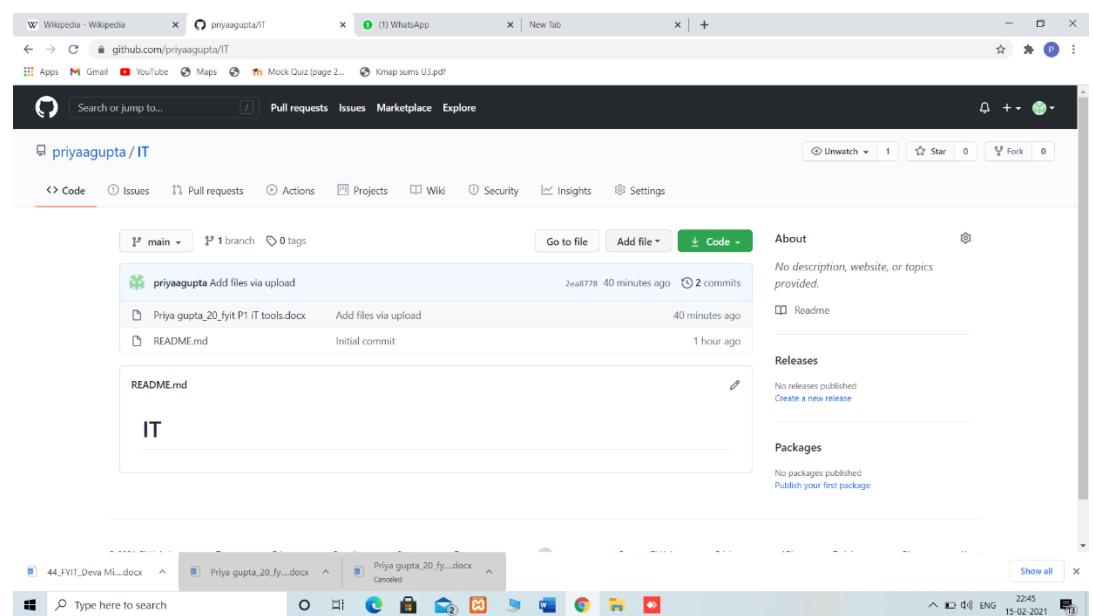
Class: FYIT

Step 3: Give name to your repository and choose one from private or public. Then click on create repository.



c) Cloning Repository:

Step 1: Click on add file button.

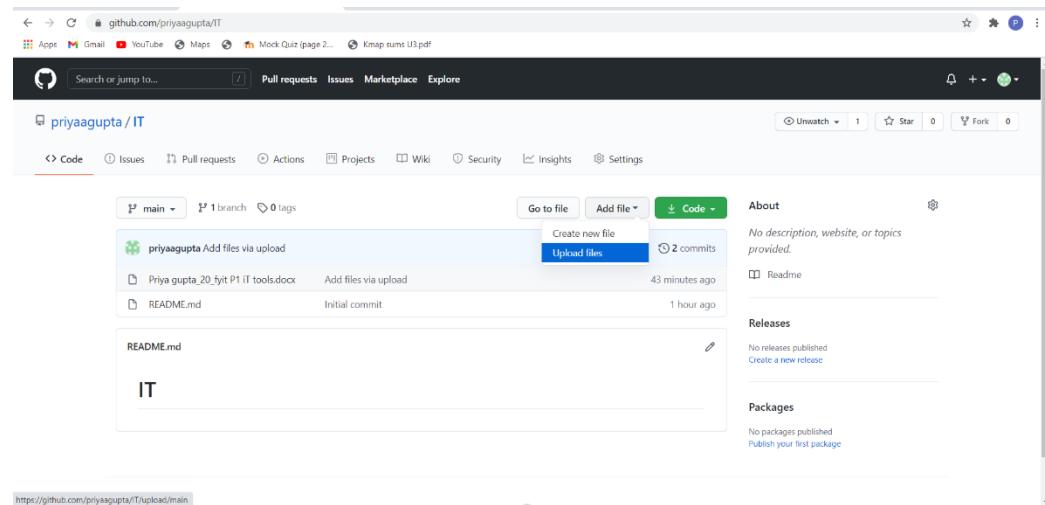


NAME: Priya Gupta

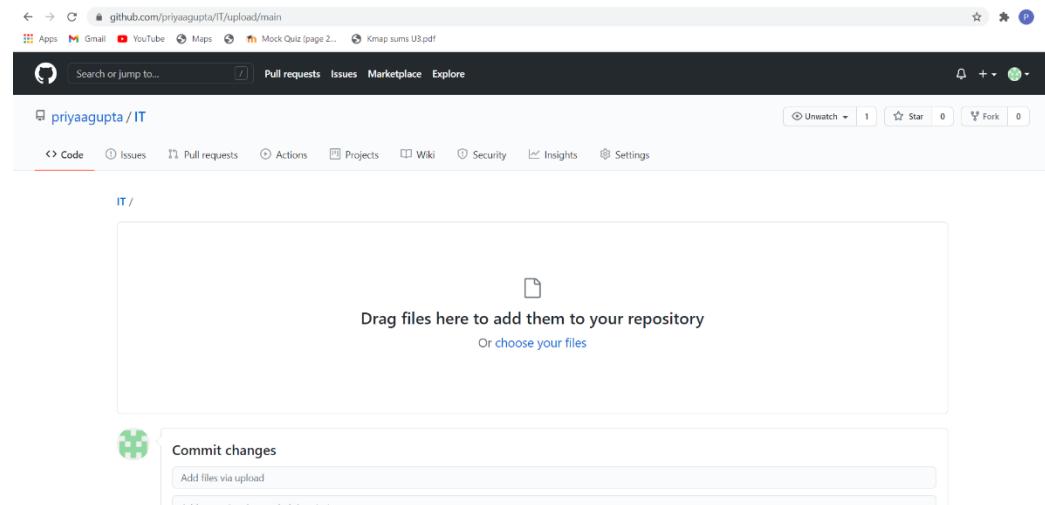
Roll No.: 20

Class: FYIT

Step 2: Click on upload file button.



Step 3:- Click on choose your files.

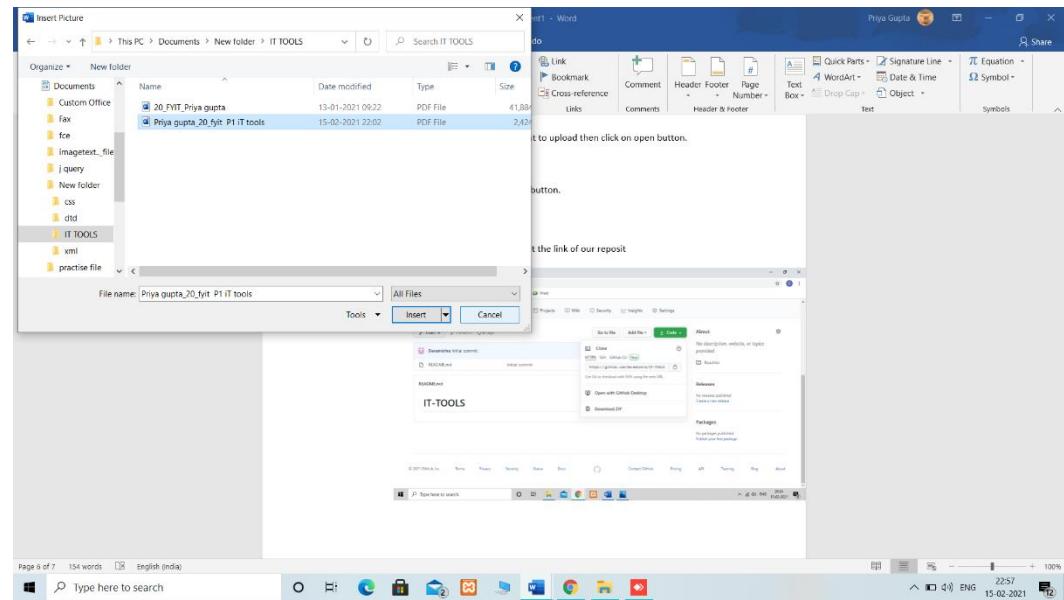


NAME: Priya Gupta

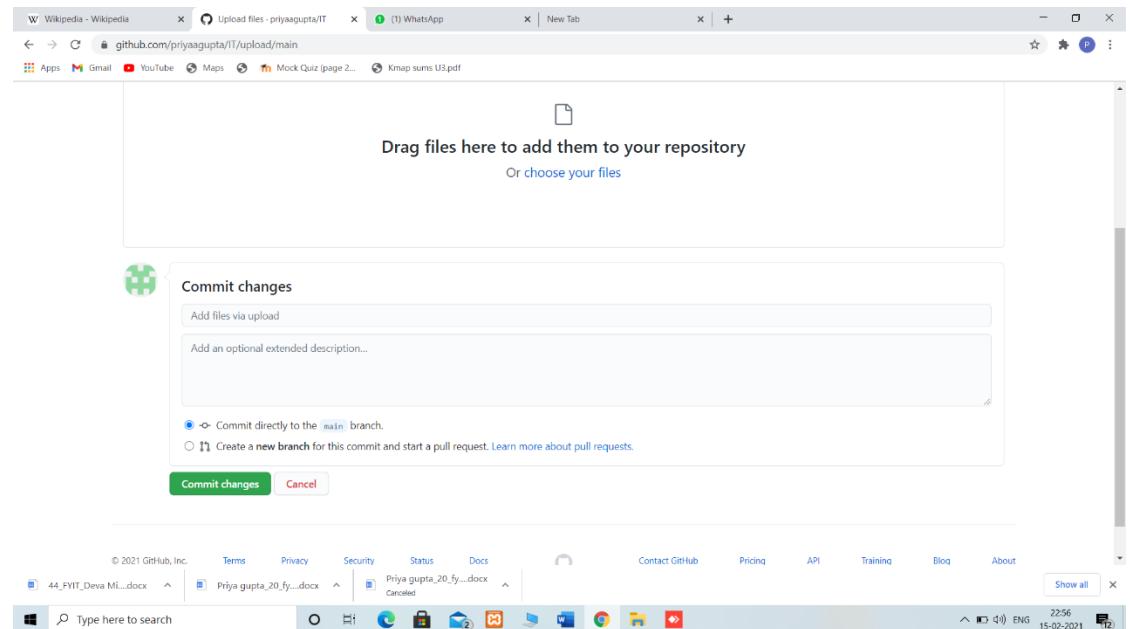
Roll No.: 20

Class: FYIT

Step 4 :- Select file which you want to upload then click on open button.



Step5 :- Click on commit changes button.

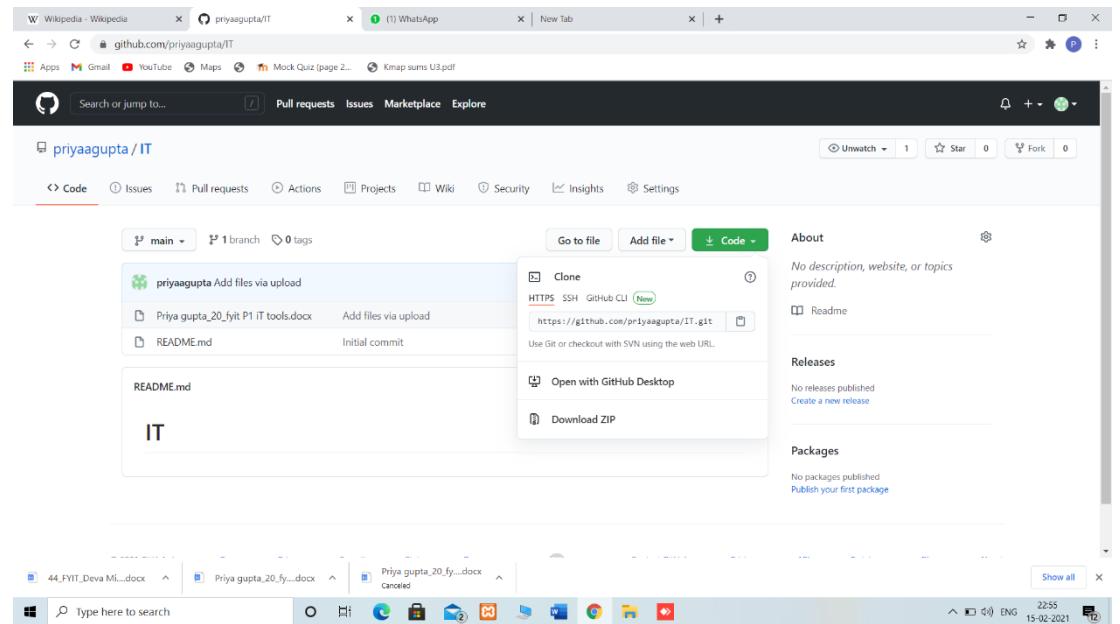


NAME: Priya Gupta

Roll No.: 20

Class: FYIT

Step6:- Click on code button to get the link of our reposit



NAME – Priya Gupta

ROLL NO. – 20

CLASS - FYIT

PRACTICAL 3

BASIC UNDERSTANDING ON FREE AND OPEN-SOURCE SOFTWARE

1. Describe Open Source Software with Example.

- ⑦ Open-source software (OSS) is any computer software that's distributed with its source code available for modification. That means it usually includes a license for programmers to change the software in any way they choose: They can fix bugs, improve functions, or adapt the software to suit their own needs

While its lack of cost is a key advantage, OSS has several additional benefits:

- Its quality can be easily and greatly improved when its source code is passed around, tested, and fixed.
- It offers a valuable learning opportunity for programmers. They can apply skills to the most popular programs available today.
- It can be more secure than proprietary software because bugs are identified and fixed quickly.
- Since it is in the public domain, and constantly subject to updates, there is little chance it can become unavailable or quickly outmoded—an important plus for long-term projects.

Open-source technologies helped establish much of the internet. Furthermore, many of the programs in use every day are based on open-source technologies. Cases in point: Android OS and Apple's OS X are based on the kernel and Unix/BSD open-source technologies, respectively.

Other popular open-source software is:

- Mozilla's Firefox web browser
- Thunderbird email client
- PHP scripting language
- Python programming language
- Apache HTTP web server

2. Describe Free Software with Example

- ⑦ Free software is software that can be freely used, modified, and redistributed with only one restriction: any redistributed version of the software must be distributed with the original terms of free use, modification, and distribution (known as copyleft). The definition of free software is stipulated as part of the GNU Project and by the Free Software Foundation. Free software may be packaged and distributed for a fee; the "free" refers to the ability to reuse it, modified or unmodified, as part of another software package. As part of the ability to modify, users of free software may also have access to and study the source code.

Free software is easily confused with freeware, term describing software that can be freely downloaded and used but which may contain restrictions for modification and reuse.

Best Free software at a glance:

1. LibreOffice
2. VLC Media Player
3. GIMP
4. Shortcut
5. Brave
6. Audacity
7. KeePass
8. Thunderbird
9. FileZilla
10. Linux

3. Difference between Free and Open Source Software.

→ 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 improve the software. The term "free software" is sometimes misunderstood—it has nothing to do with price. It is about freedom.

Open-Source Software:

Open Source Software is something which you can modify as per your needs, share with others without any licensing violation burden. When we say Open Source, source code of software is available publicly with Open Source licenses like GNU (GPL) which allows you to edit source code and distribute it. Read these licenses and you will realize that these licenses are created to help us.

- Coined by the development environments around software produced by open collaboration of software developers on the internet.
- Later specified by the Open Source Initiative (OSI).
- It does not explicitly state ethical values, besides those directly associated to software development.

Sr.No.	Free 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 economic 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 Send mail transport agent.	Examples: Prime examples of open-sources products are the Apache HTTP Server, the e-commerce platform ecommerce, 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

Name: - Priya Gupta

Class: - FYIT 20

Requested for Extension of Time for Assignment

sweety@nkc.ac.in



Requested for Extension of Time for Assignment

Respected ma'am

Good Morning

I am Priya Gupta, student of FYIT Roll no. 20 . I apologize for not being able to complete my assignment on time and missing the due date.

I request you to extend the Submission date for the assignment. I hope you comprehend. I am sure that I will complete it by that time and submit you promptly.

Looking forward for your positive response.

regards

Priya Gupta

FYIT-20

SIGNATURE

NAME: - Priya Gupta.

CLASS: - FYIT 20

Practical-5

Using Practical examples, describe green computing. List and explain the steps that you take to contribute to green computing.

Ans: Green computing is the environmentally responsible and eco-friendly use of computers and their resources. In broader terms, it is also defined as the study of designing, manufacturing/engineering, using, and disposing of computing devices in a way that reduces their environmental impact.

Green computing is the study and practice of environmentally sustainable computing or IT. The goals of green computing are similar to green chemistry reduce the use of hazardous materials, maximize energy efficiency during the product's lifetime, the recyclability or biodegradability of defunct products and factory waste. Green computing is important for all classes of systems, ranging from handheld systems to large-scale data centers. Many corporate IT departments have green computing initiatives to reduce the environmental effect of their IT operations. Many IT manufacturers and vendors are continuously investing in designing energy-efficient computing devices, reducing the use of dangerous materials and encouraging the recyclability of digital devices. Green computing practices came into prominence in 1992, when the Environmental Protection Agency (EPA) launched the Energy Star program. Green computing is also known as green information technology (green IT).

Steps To contribute to green computing:

1. Proclamation of the Green Intentions:

It is always best to begin Green IT initiatives by communicating intentions to adopt an environment-friendly IT infrastructure. The push for energy efficiency should be cascaded down to every staff, setting the stage for collaboration between various departments. Once they learn about the initiatives, they will know that everyone needs to be involved.

2. Appointment of a working Group for Green IT Compliance Assurance:

Once the ball is set to roll, you need to have a committee that will monitor and ensure that the company's plans are adhered to by all members of the organization. One of the most important tasks that the appointed Green IT Committee must focus on is the acquisition of energy efficient IT infrastructure. This team should make sure that the IT groundwork meets all the criteria that are set for the protection of the environment.

3. Measurement of current Carbon Footprints produced by IT Components:

Where the company stands in terms of carbon footprint brought about by information technology services, is an important information to be known. Quickly establish a carbon footprint reference point. Check on the power usage in the IT center and compare it with existing power efficiency standards and metrics for industry.

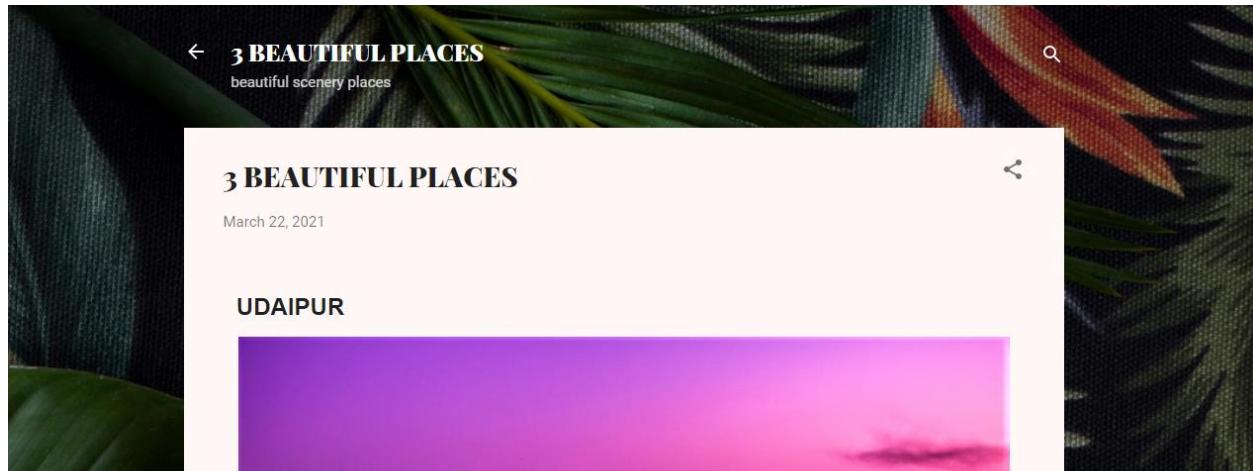
4. Usage of More Efficient Computer Application:

By using more powerful computer applications, your IT systems can better deal with inefficiencies. Besides, faster software spares the servers from regularly operating at maximum capacity, thereby consuming lesser power. If one can only increase the speed of the computer applications that is used, one can have a corresponding positive effect on the energy use and carbon emissions.

5. Usage of More Efficient Cooling System:

By using more powerful computer applications, your IT systems can better deal with inefficiencies. Besides, faster software spares the servers from regularly operating at maximum capacity, thereby consuming lesser power. If one can only increase the speed of the computer applications that is used, one can have a corresponding positive effect on the energy use and carbon emissions.

Practical 6: BLOG



← **3 BEAUTIFUL PLACES**
beautiful scenery places

3 BEAUTIFUL PLACES

March 22, 2021

UDAIPUR



Udaipur which is known for its many lakes and the most famous lake is Pichola also with crystal blue color water that represents every ounce lying on it. There are other places in that city like temples, Ghats, hillsides. In the evening, we can ride the boat on one of the lake.

KHAJJiar



Khajjiar which is in Himachal Pradesh near the Himalayan Mountains. Khajjiar looks even more beautiful because of the green plains, forests, it can be said that it is blessed with green grasslands and forests. This is the perfect place to holiday, here we will be very relaxed if we go on holidays with family or friends.

MANALI



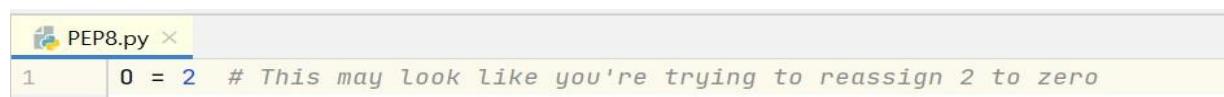
Manali is a snowy place, we can have a lot of fun in the winter season and also spend our holidays with fun. It has an icy slope, and a clear blue sky. Which makes it even more beautiful

PRACTICAL-07: 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.



```
PEP8.py ×
1 0 = 2 # This may look like you're trying to reassign 2 to zero
```

2) How to Choose Names:

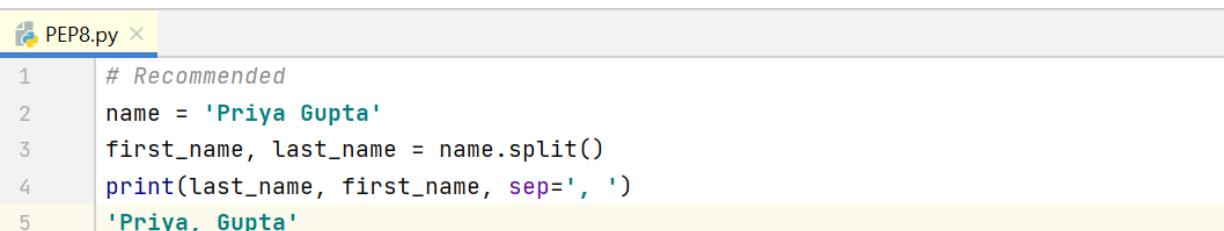
When naming variables, you may be tempted choose simple, single-letter lowercase names, like `x`. But, unless you're using `x` as the argument of a mathematical function, it's not clear what `x` represents.

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. Imagine you are storing a person's name as a string, and you want to use string slicing to format their name differently. You could end up with something like this:



```
PEP8.py ×
1 # Not Recommended
2 x = 'Priya Gupta'
3 y, z = x.split()
4 print(z, y, sep=', ', )
5 'Priya, Gupta'
```

The following example is much clearer. If you come back to this code a couple of days after writing it, you'll still be able to read and understand the purpose of this function:



```
PEP8.py ×
1 # Recommended
2 name = 'Priya Gupta'
3 first_name, last_name = name.split()
4 print(last_name, first_name, sep=', ', )
5 'Priya, Gupta'
```

3) Code Layout:

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.

```
1 #CORRECT
2 # Perform some math
3 a = 1+2
4 b = 3+4
5 c = a+b
6
7 # Read in and Plot some
8 preceip_timeseries = pd.readcsv("precip-2019.csv")
9 preceip_timeseries.plot()
```

```
1 #WRONG
2 a=1+2
3 b=3+4
4 c=a+b
5 date=pd.readcsv("precip=2019csv")
6 date.plot()
```

4) Whitespace in Expressions and Statements:

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

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)

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

```
1 # Not Recommended
2 y = x ** 2 + 5
3 z = (x + y) * (x - y) |
```

5) 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.

- **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

The screenshot shows a code editor window titled "PEP8.py". The code contains two lines of Python:

```
1 def print_name(self):  
2     print(self.name) # comment is correct now needs a space
```

The second line has a comment "# comment is correct now needs a space" starting directly after the opening parenthesis. This is considered an anti-pattern according to PEP 8 style guidelines.

-Best practice

The screenshot shows a code editor window titled "PEP8.py". The code contains two lines of Python:

```
1 def print_name(self):  
2     print(self.name) # comment is correct now
```

The second line has a comment "# comment is correct now" starting with a single space after the opening parenthesis, which is the correct way to format inline comments according to PEP 8.

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: %.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.

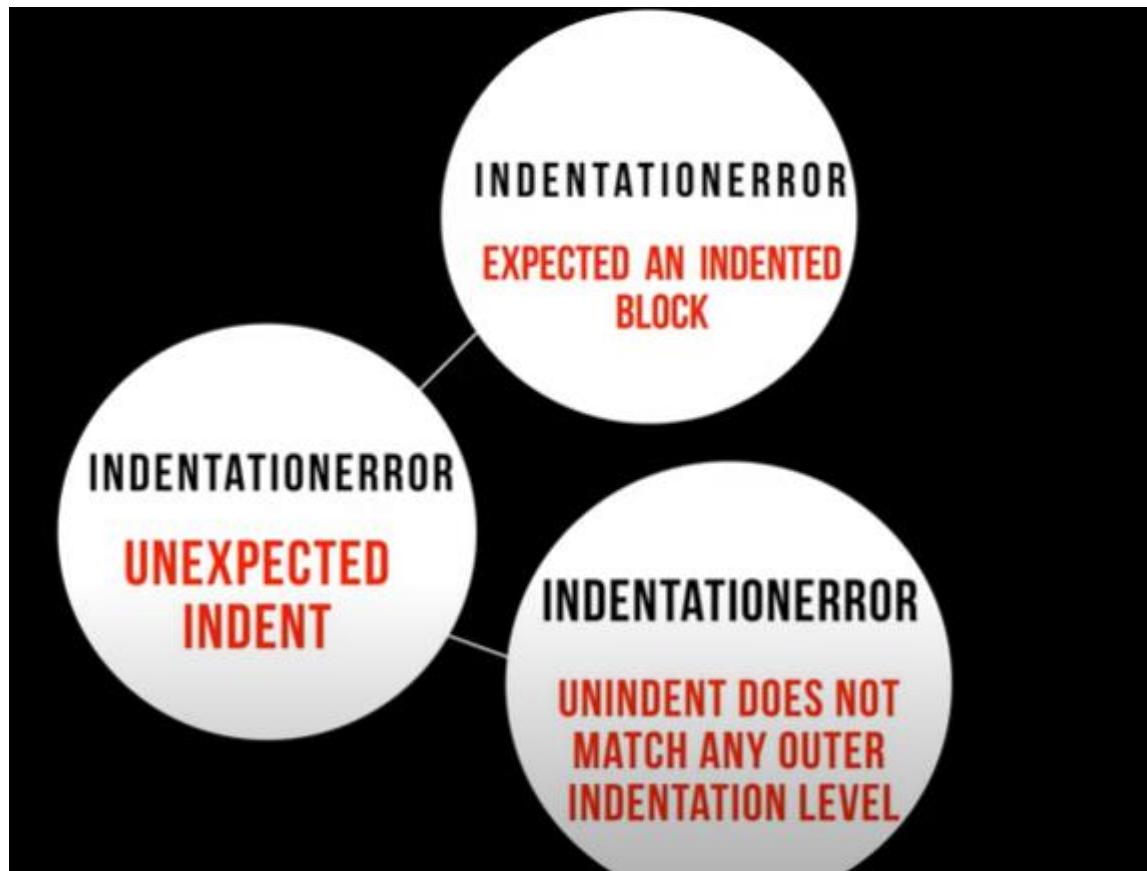
In the below Example

#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

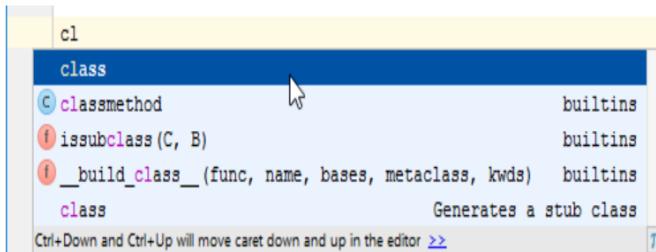
ANSWER: **NEVER**

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

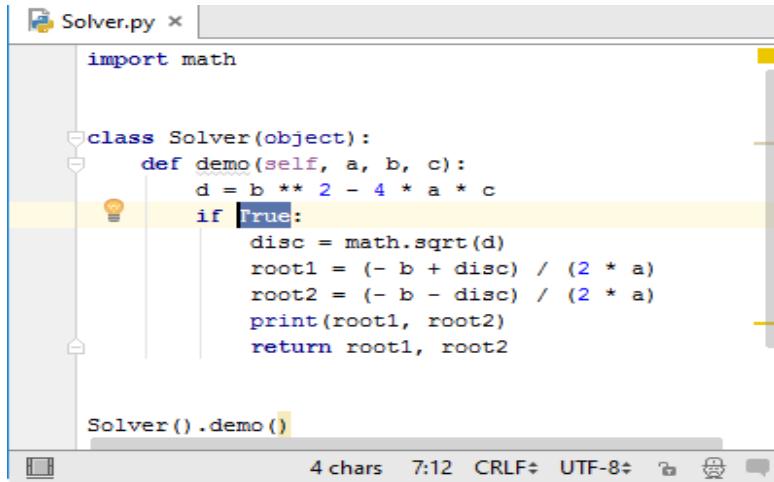
Highliting code style violations:



(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

- Example:

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

Not Recommended

Python

- Example:

```
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
```

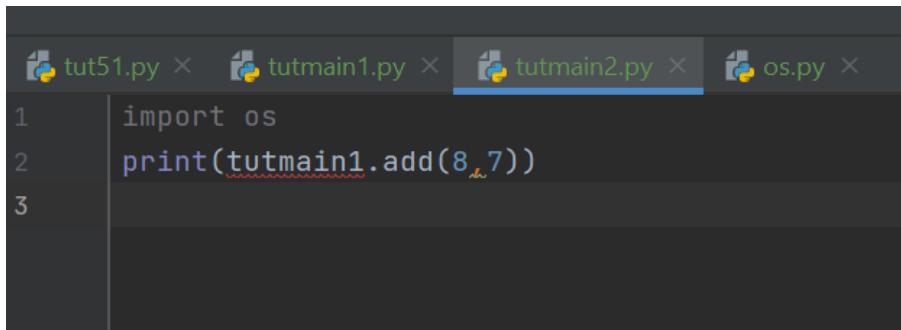
Documentation string comment.

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

EXAMPLE OF NAMING CONVENTION

NAMING MODULE WITH HELP OF PEP8

Not recommended

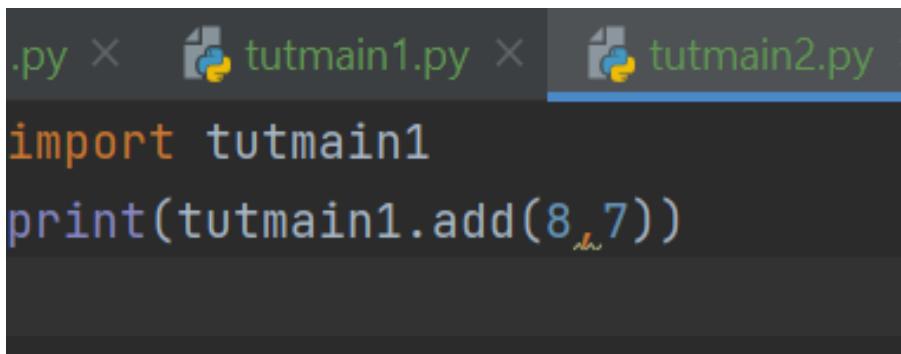


A screenshot of a code editor showing a file named `tutmain2.py`. The code contains:

```
1 import os
2 print(tutmain1.add(8,7))
3
```

The line `print(tutmain1.add(8,7))` is underlined with a red squiggly line, indicating a syntax error or warning.

Recommended



A screenshot of a code editor showing a file named `tutmain2.py`. The code contains:

```
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'
```

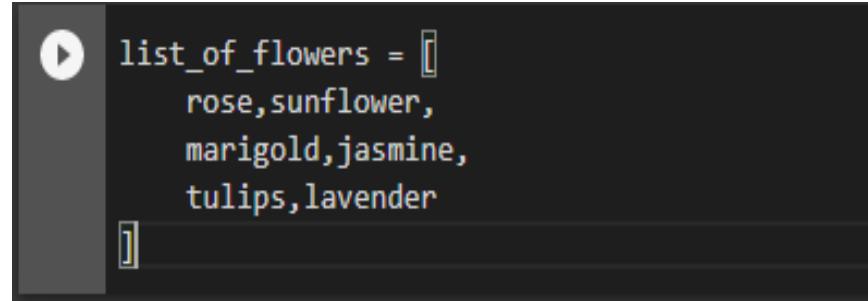
>>>

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

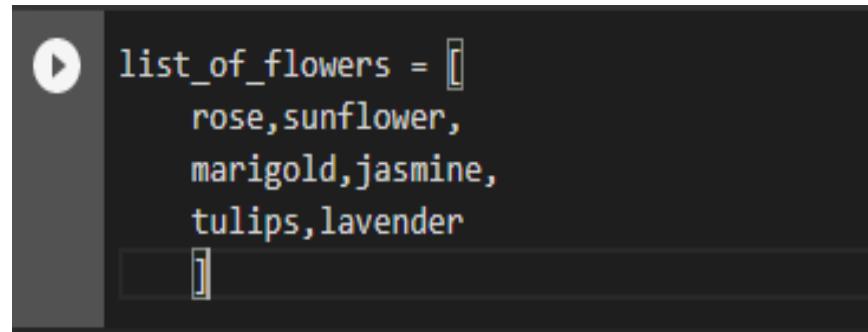
⊕ EXAMPLES OF INDENTATION

❖ METHODS OF WHERE TO PUT CLOSING BRACES:-

Recommended



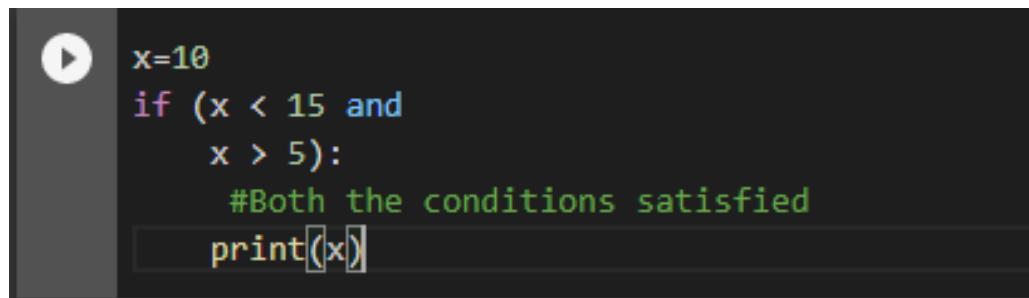
```
list_of_flowers = [rose,sunflower,  
                   marigold,jasmine,  
                   tulips,lavender]  
[]
```



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

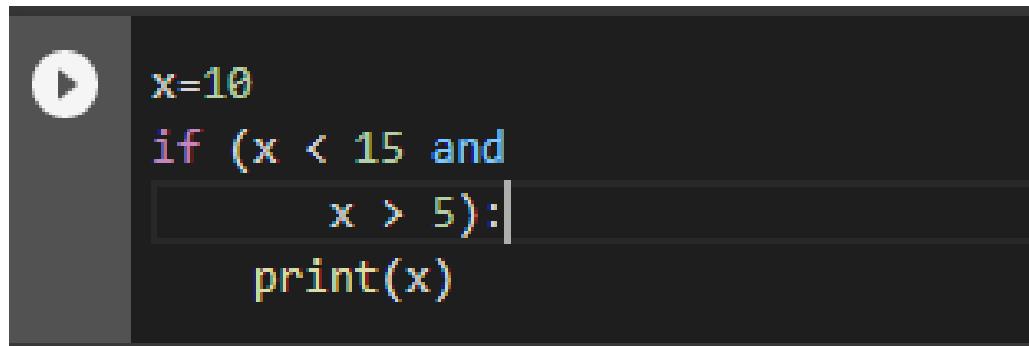
❖ Methods for following line breaks

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 white and light gray 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 white and light gray 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