**DAY-4 ASSIGNMENT**

Name: Madhura B

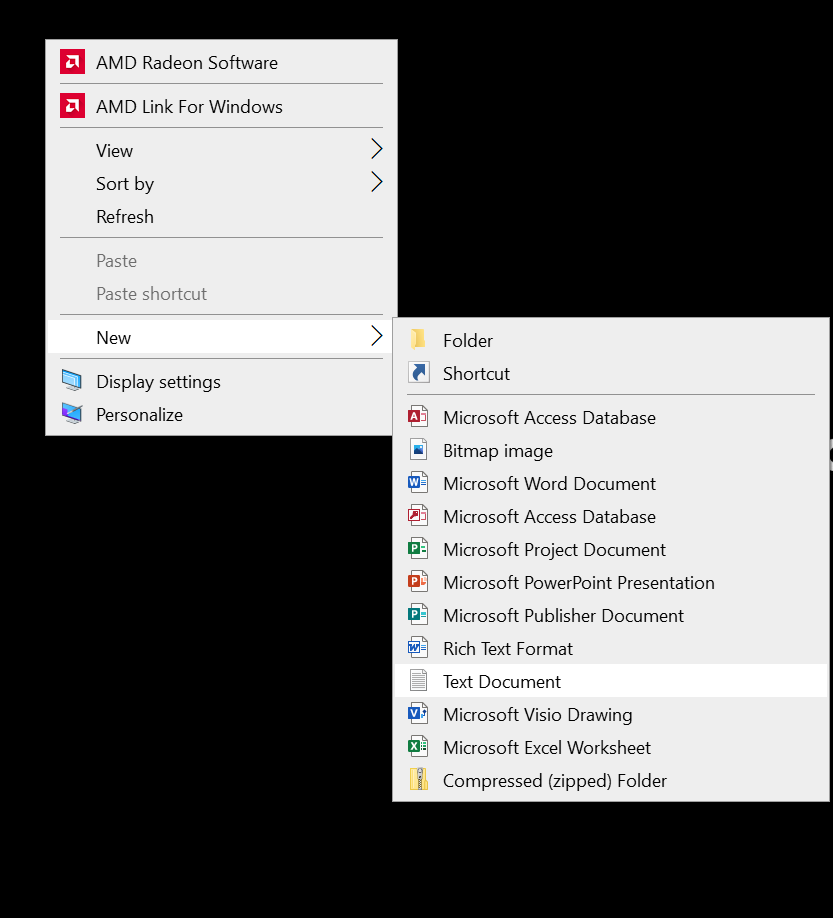
Email: [madhura23madhu@gmail.com](mailto:madhura23madhu@gmail.com)

Employee ID: 11718307

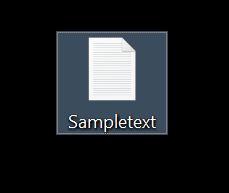
Candidate ID: CANDIDATE-3-1538817

1. Explain File handling in python and attach the screenshots.?

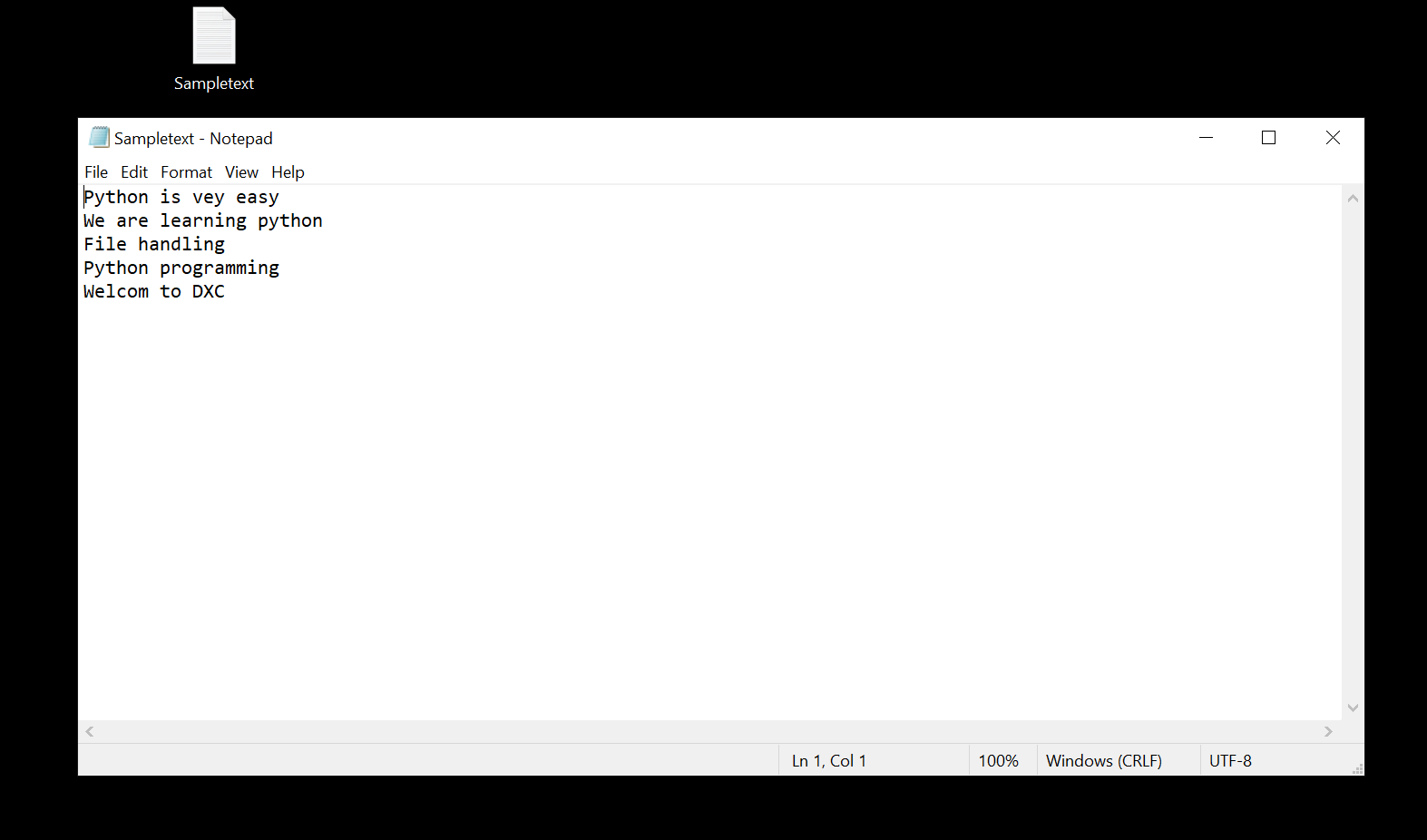
Ans: Step1: Create a sample text file document and enter some data within it and save.



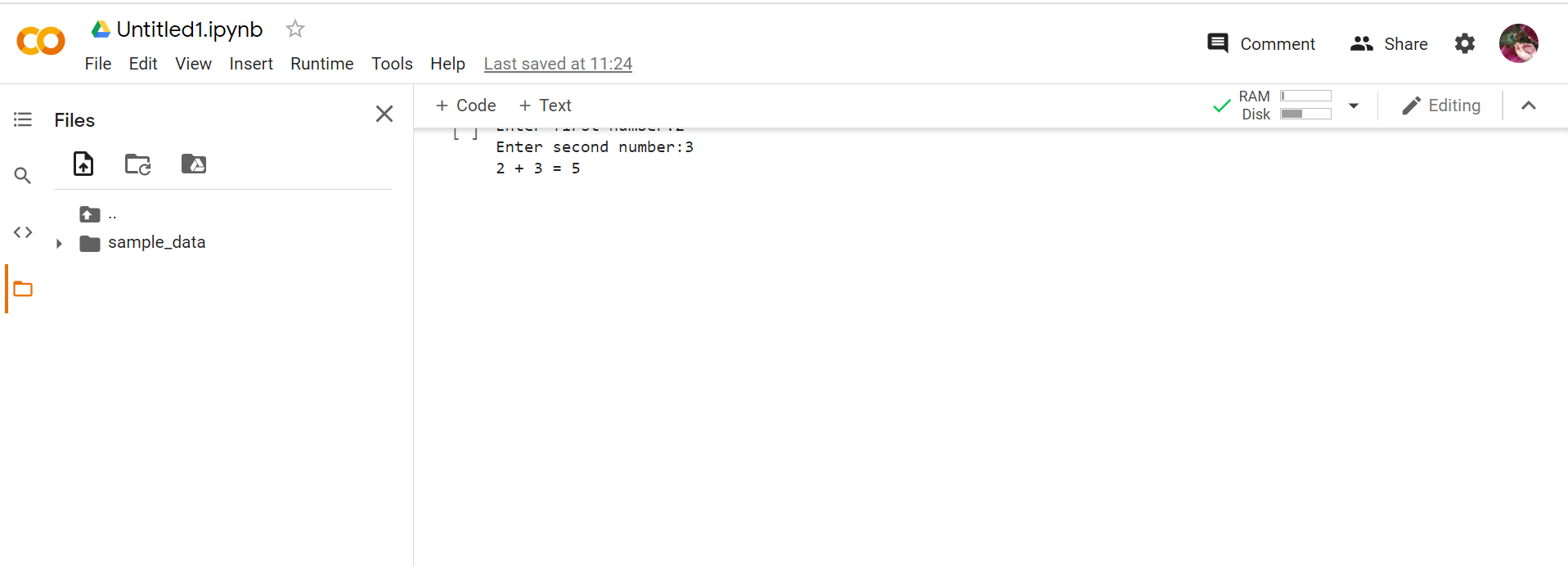
Step2: The sample text file has been created.



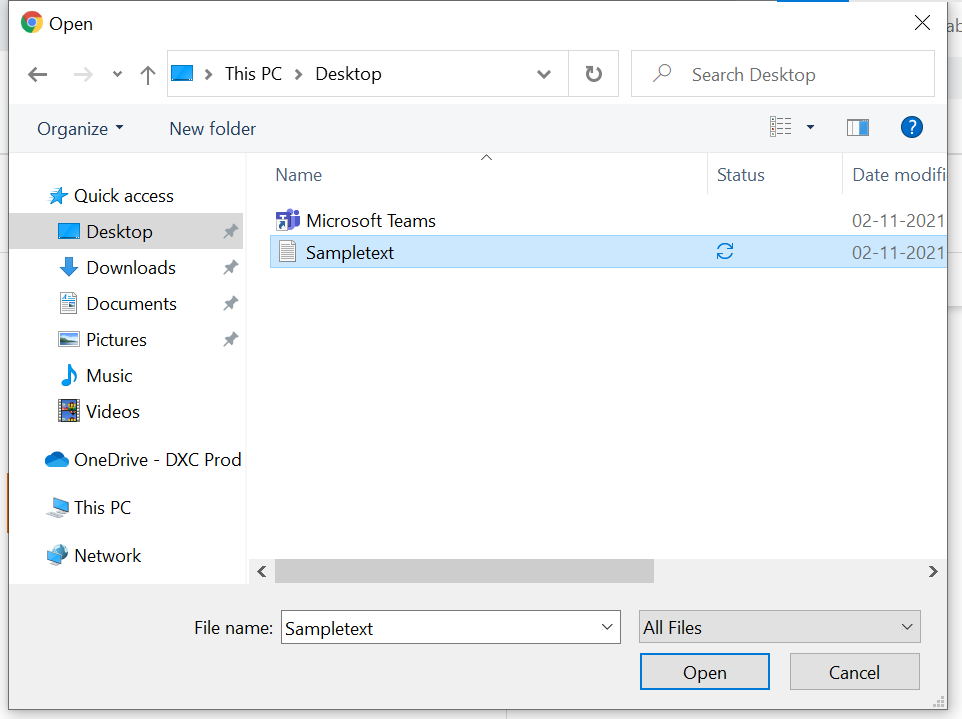
Step3: Now add and write some of the data within that created sample text file.



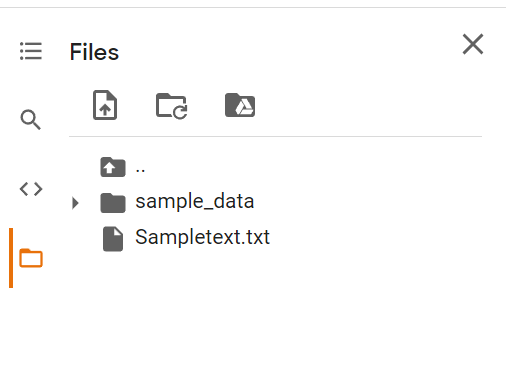
Step4: Now the sample text file which we had created should be uploaded in the below shown as in the snap.



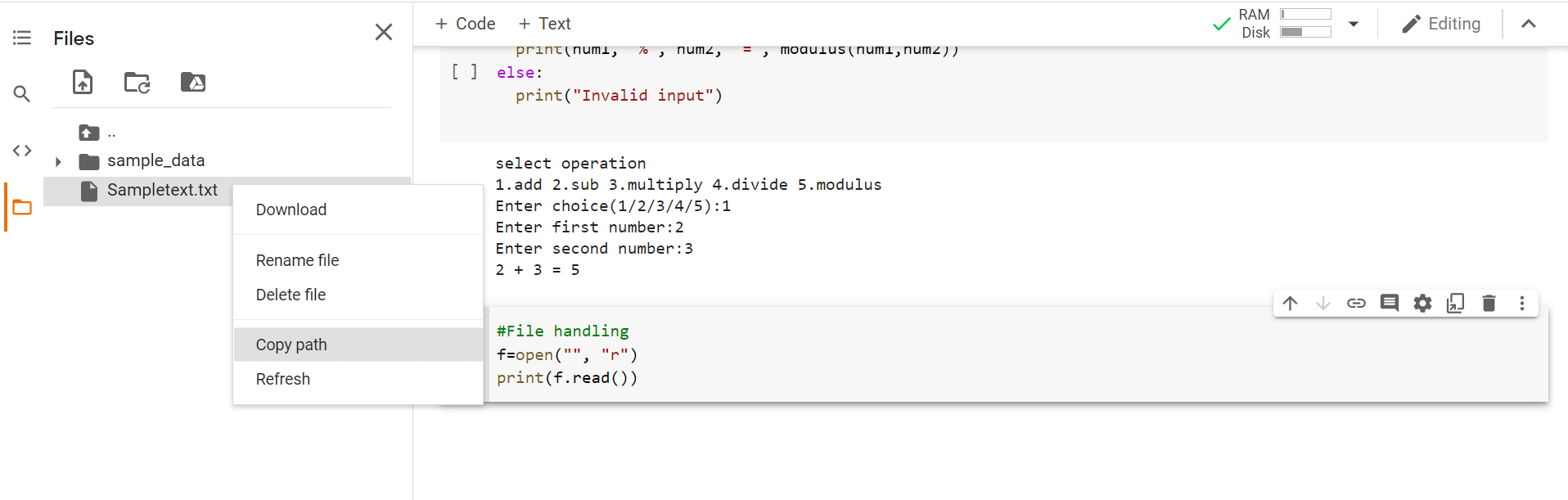
Now upload the file where you have saved that sample text file.



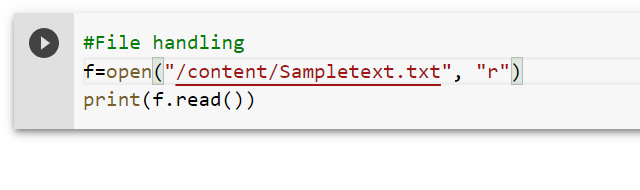
After uploading the uploaded file will be appear as shown in the below.



Now we need to read the data so while writing the python code we need to copy the path as shown below.



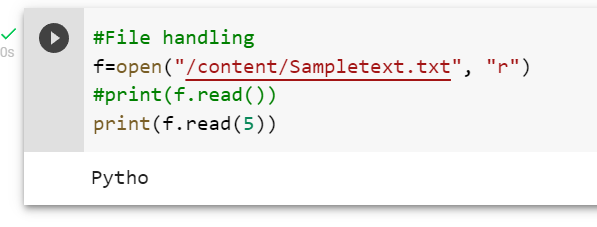
Now paste the path over in the code.



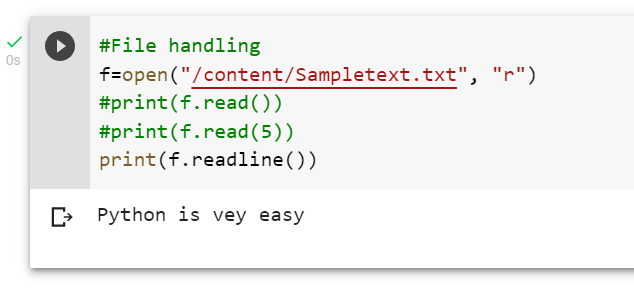
Now by running the code we will be getting the output as shown in the below snap is the example.



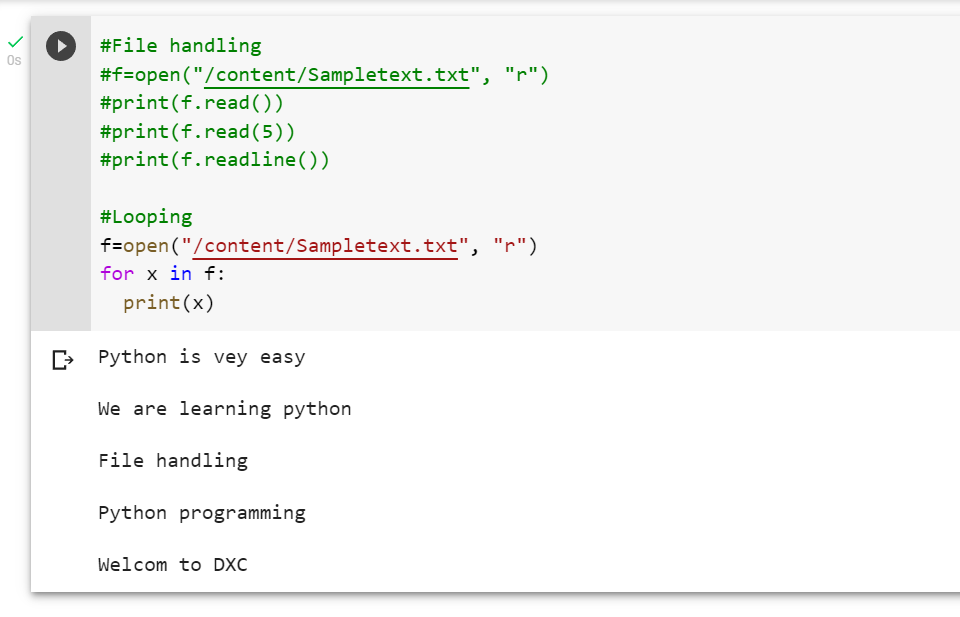
Now in order to print the number of characters whichever we want we need to write the code as shown as follows.



Now to print the first starting lines of the data which is present in the text file we had to follow the code as shown as follows.

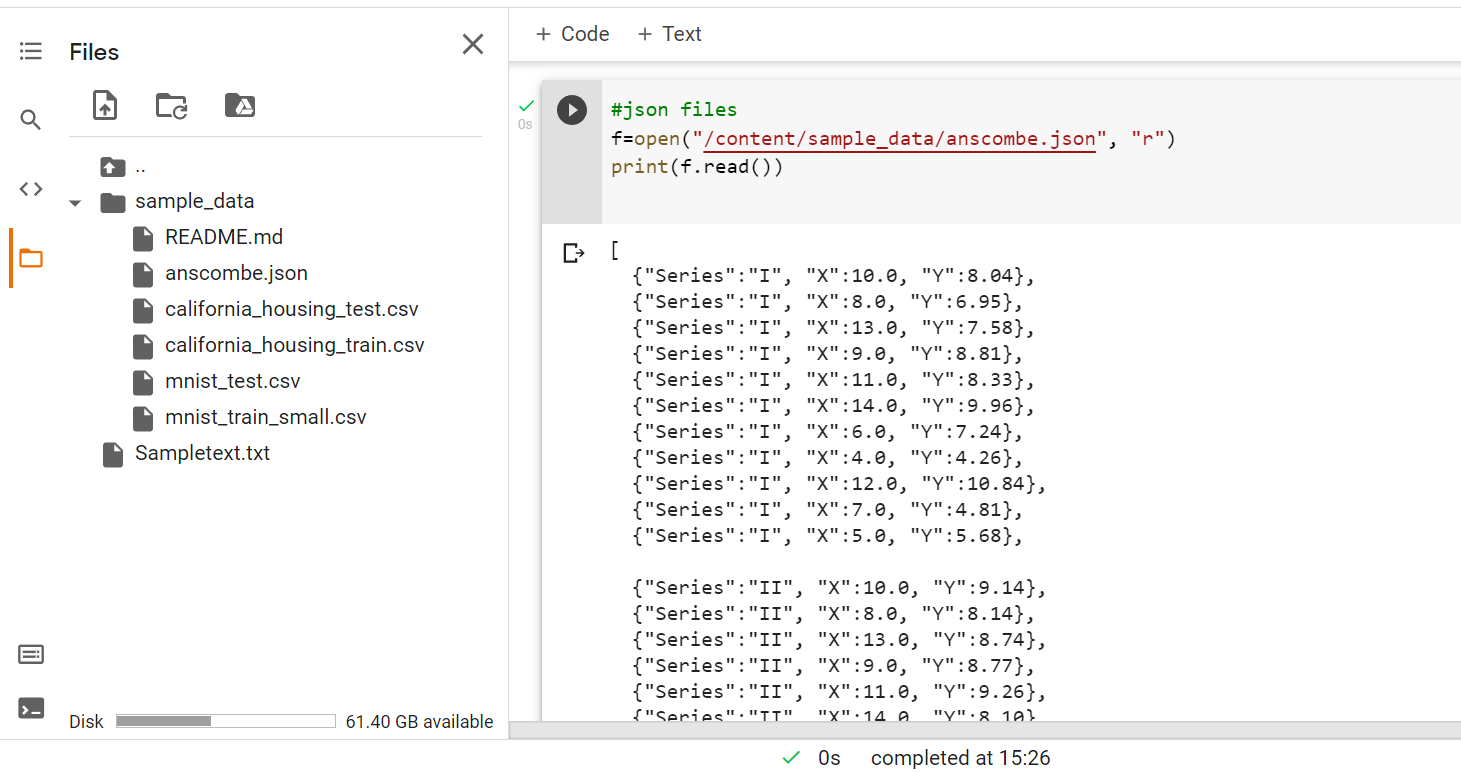


Now for looping we follows the below code as follows.

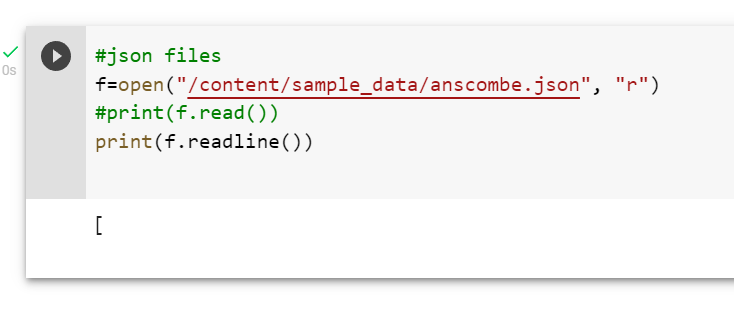


Now for json files:

Now for json files copy the path from json path and paste it over the code as follows.



Now in order to display only the first lines of the json files the code is as follows:

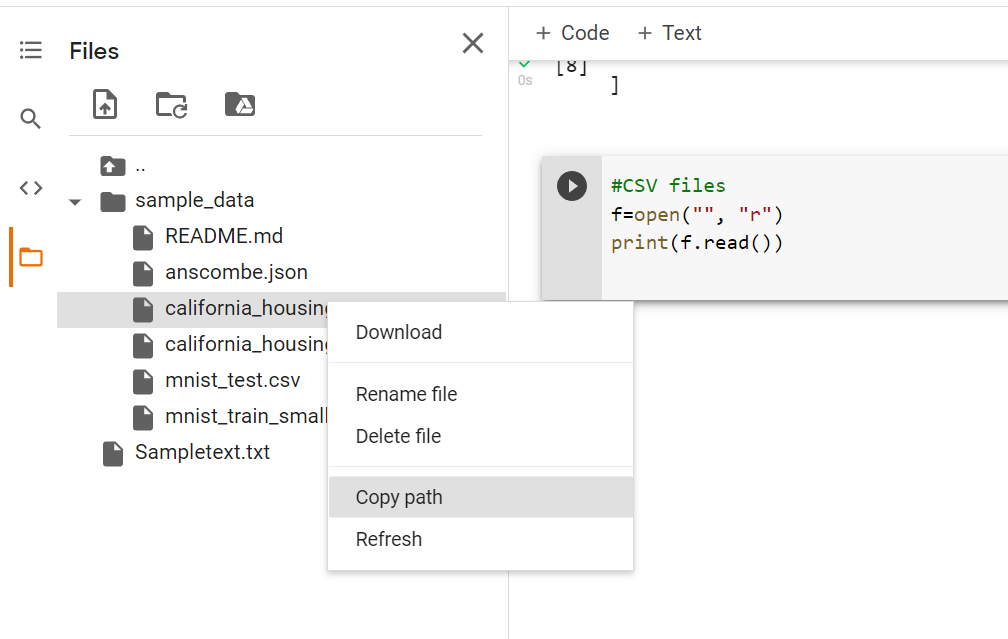


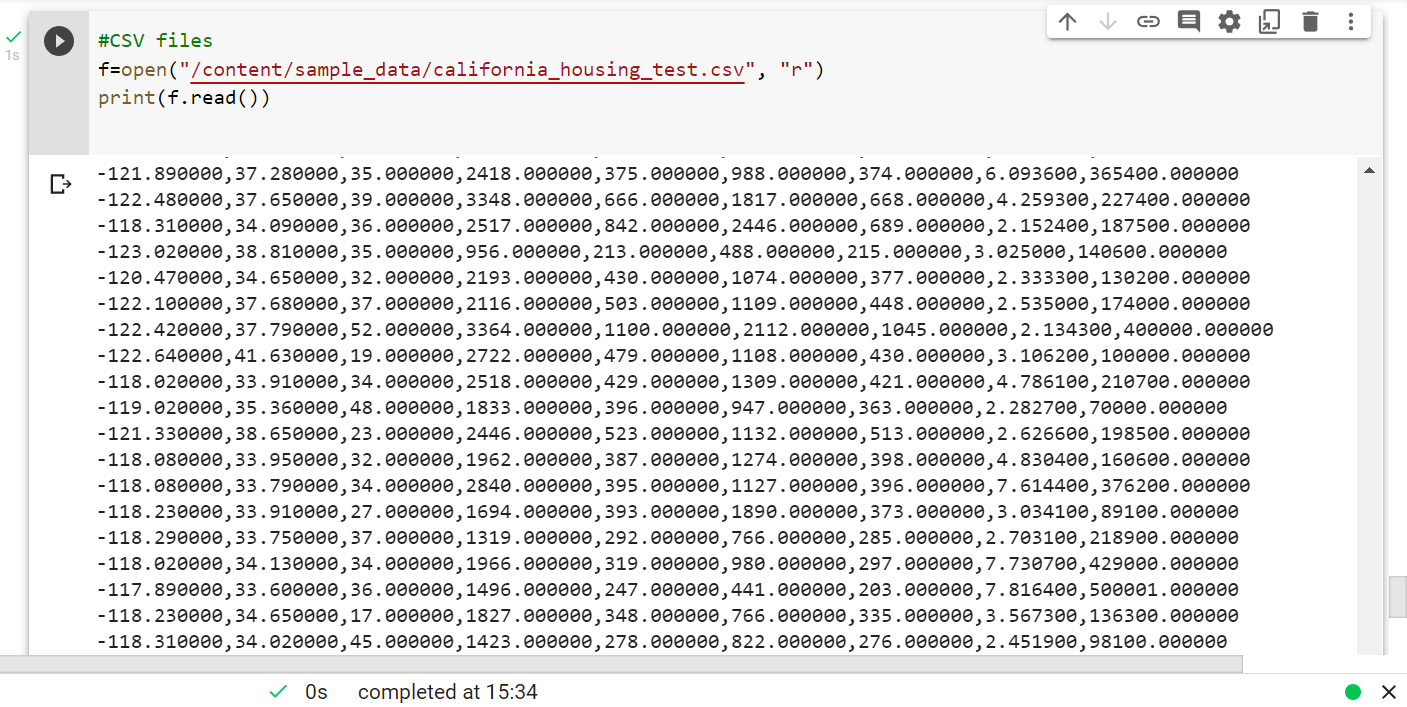
Now for looping in json files is as follows:



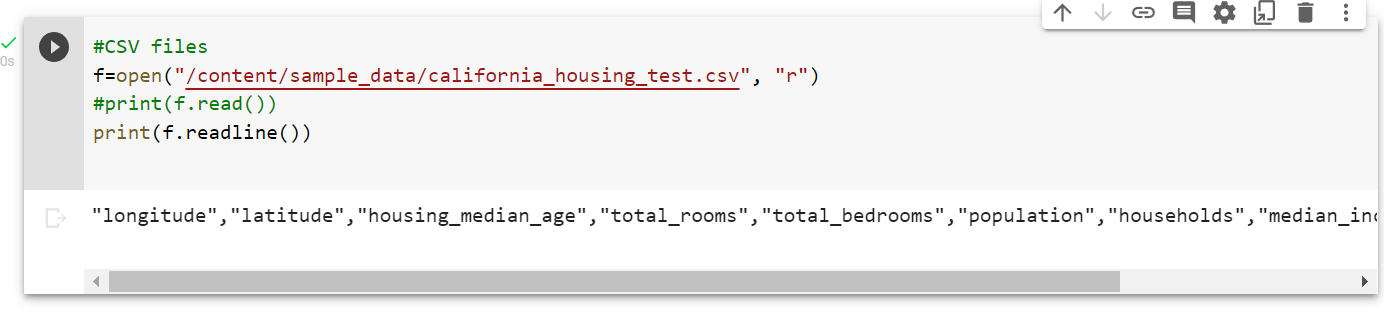
Now for CSV files:

Copy the path related to CSV path and paste over the python code.



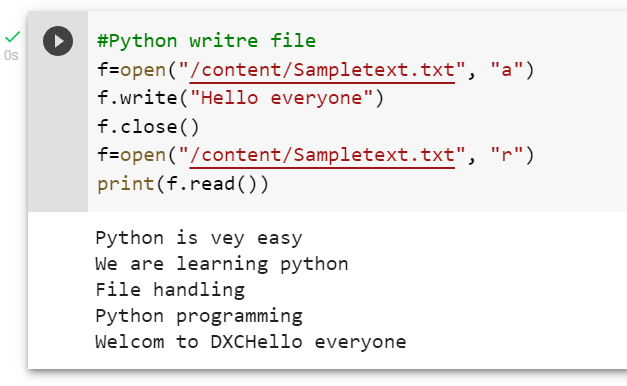


Now to print only the starting lines of code is as follows



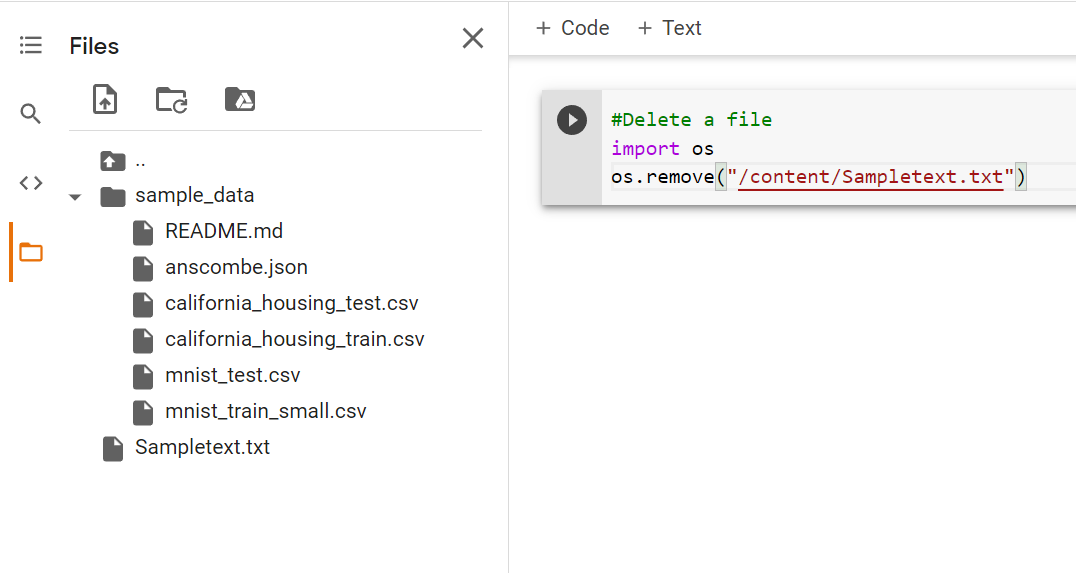
Now for writing a file:

Following the code below same and copy the path which we want to write.

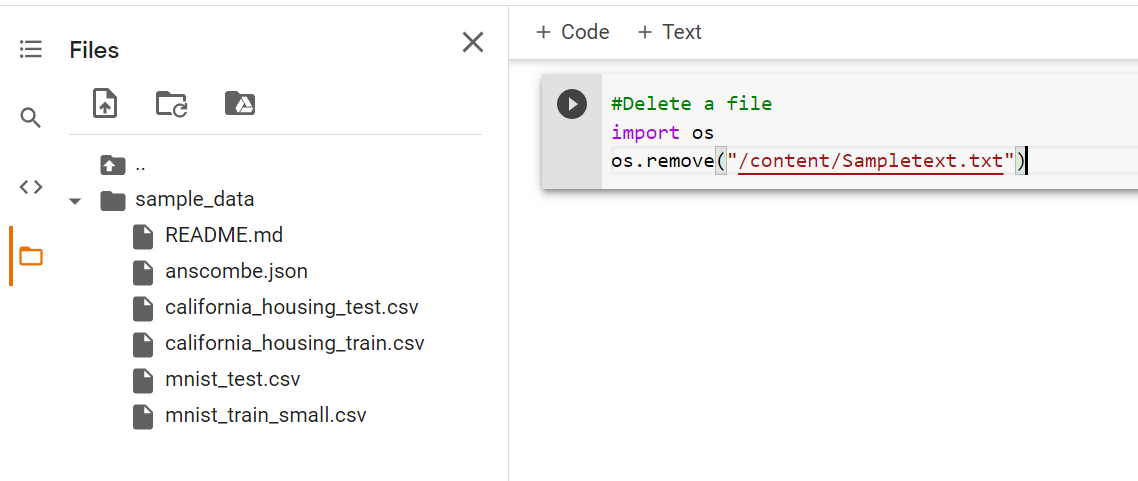


Now for deleting the file:

The code is as shows as follows:



Now we can see that there is no sample file in the folder so that we had removed it.



2. Explain water fall model. why water fall model failure?

Ans: The Waterfall Model was the first Process Model to be introduced. It is also referred to as a **linear-sequential life cycle model**. It is very simple to understand and use. In a waterfall model, each phase must be completed before the next phase can begin and there is no overlapping in the phases.

The Waterfall model is the earliest SDLC approach that was used for software development.

The waterfall Model illustrates the software development process in a linear sequential flow. This means that any phase in the development process begins only if the previous phase is complete. In this waterfall model, the phases do not overlap.

**Reasons for failure waterfall model:**  
The traditional waterfall model suffers from various shortcomings, basically we can’t use it in real projects, but we use other software development lifecycle models which are based on the classical waterfall model.

There are some reasons which are given below due to this waterfall model fails.

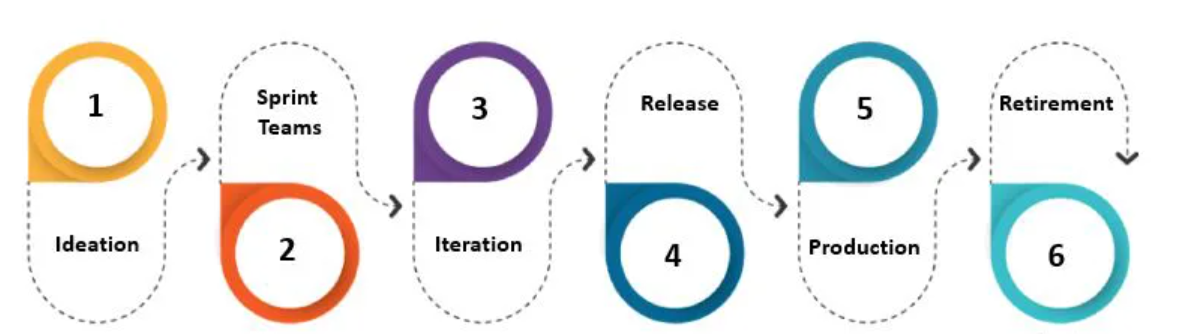
* **One way street:** This model is just like the one-way street. Once phase X is completed and next phase Y has started then there is no way to going back on the previous phase. This is one of the issues to the failure of the waterfall model.
* **Overlapping:** The waterfall model has lacked an overlapping among phase.The waterfall model recommends that new phase can start only after the completion of the previous phase. But in real projects, this can’t be maintained. To increase the efficiency and reduce the cost, phases may overlap.
* **Interaction:** The waterfall model has lacked interaction among phase. Users have little interaction with project them. This feedback is not taken during development. After a development process starts, changes can not accommodate easily.
* **Support delivery of system:** The waterfall model does not support delivery of system in pieces. After a development process starts, changes cannot accommodate easily.
* **Feedback path:** The waterfall model has no feedback path. In the traditional waterfall model evolution of software from one phase to another phase is like a waterfall. The waterfall model assumes that no error is ever committed by developers during any phases. Hence, it does not incorporate any mechanism for error correction.
* **Not Flexible:** Difficult to accommodate change requests. The waterfall model assumes that all the customer requirements can be completely and correctly defined at the beginning of the project, but actually customers’ requirements keep on changing with time. After the requirements specification phase is completed difficult to accommodate any change requests.

3. Explain Workflow of Agile Model?

Ans: Agile workflow is an iterative method of delivering a project. In Agile, multiple individual teams work on particular tasks for a certain duration of time termed as ‘Sprints’. “Agile Workflow can be defined as the set of stages involved in developing an application, from ideation to sprints completion”.

Azile workflow/ Lifecycle:

Let’s understand the Agile Workflow and its stages:



The Agile application development lifecycle consists of:

* **Ideation:** The very first stage of the Agile workflow is about defining the business scope for each of the project ideas and determining the time and set of tasks to execute such projects. This judgment helps in identifying financial and resource availability so as to conclude which ideas are worth executing!
* **Creation of Sprint Teams:**After identifying the project and discussing it with the stakeholders, the agile team members are identified and resources are allocated so as to execute the project. These team members are assigned work responsibilities and the timeframe to complete the same for each of sprint teams.
* **Iteration:**When sprint teams are identified upon discussion and feedback from the stakeholders, the project work is resumed. The team with the objective of launching a workable product at the end of a sprint, start working on the first iteration. As the product goes through many rounds of improvement or iteration, the first iteration usually includes minimum functions. The team can put up more sprints to develop the entire product.
* **Release:**At this stage, the product is ready to pass into the production phase. The quality team tests the product functionality and rectifies defects or errors if any.
* **Production:**During the production stage, the team must ensure that the product is being launched successfully and guide the users on its usage. Thus, at this phase, the team offers continual support for the product release.
* **Retirement:**This is the last stage of the azile software development cycle. As the name indicates, the product is removed or replaced with a new release. The new product is introduced when the previous product becomes redundant or does not fit the ongoing business model.

The different types of Agile Workflow models are outlined below:

* Feature Driven Development (FDD): As the name indicates, FDD is an iterative and progressive workflow model for system development, primarily focusing on features. The concept of FDD is to come up with high-end features and scope, so as to develop the entire model and execute the same. The FDD goes through the five stages:
* **Develop the Entire Model:** Firstly, a few models are selected specifically to each domain. And these domain models are successively blended into one entire model based on the concept and scope.
* **Develop the Listed features:** Features are the list of requirements specific to the customers. These features are to be completed within a timeframe of not more than 2 weeks for release.
* **Feature Planning:** The listed features are then assigned to the developers.
* **Feature Designing:** The details of every feature are developed, evaluated and then finalized.
* **Feature Building:** After the design is inspected and improved, the entire feature is included in the main build to be delivered to the customers.

#### Crystal: Crystal is not a set of pre-defined tools or patterns but is a set of various software methodologies. The crystal model is human-centric and considers people as the most important asset in system development, followed by the processes to meet the client requirements. The crystal process does not pre-define the tools or methods at the very start of the project but is determined based on the business and technical requisites of a project. Based on the time duration and project conditions, crystal is further sub-divided and denoted by various colors.

#### Agile Unified Process (AUP): The unified process is an iterative and progressive method consisting of four steps. These are, ‘Inception’, ‘Elaboration’, ‘Construction’ and ‘Transition’.

#### Extreme Programming (XP): XP is one of the popular agile workflow models, which is focused on continual improvement through customer feedback. The workflow of XP for system development is comprised of 4 stages. These are, ‘Coding’, ‘Testing’, ‘Listening’, and ‘Designing’.

#### Scrum: Scrum is an agile workflow process based on a repetitive approach. Scrum emphasises continual improvement for customer satisfaction. The workflow of Scrum consists of ‘Product backlog, ‘Planning Sprint’, ‘Sprint Backlog’, ‘Sprint’, ‘Routine Scrum Meetings’, ‘Sprint Reviewal’ and ‘Internal Scrum Meetings’.

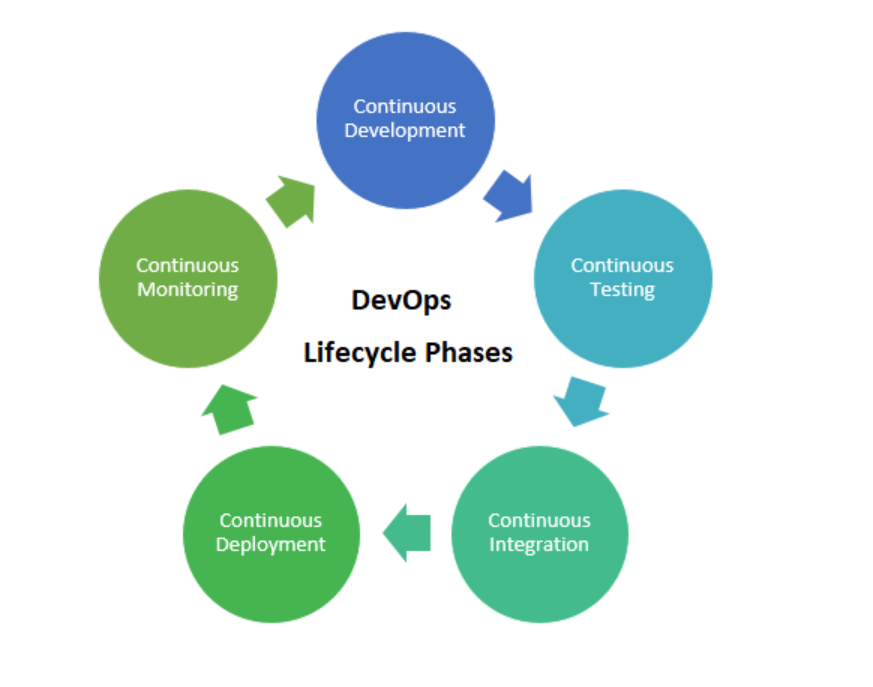
#### Kanban: Kanban is based on the lean development process, which is a popular methodology introduced by Toyota Motors. It is an agile method that is non-iterative and is less structured. It emphasizes on delivering quality on a continual basis by effective team collaboration. The Kanban workflow comprises of ‘Product Backlog’, ‘Requirements’, ‘Design’, ‘Development’, ‘Testing’, ‘Deployment’, and ‘Done’ stages.

#### Altern: One of the agile methods which have been prevailing for the longest time is also used in Non-IT sectors. In this method, the project benefits and requirements are pre-defined clearly before the actual execution. This workflow includes stages of, ‘Pre-project’, ‘Feasibility’, ‘Foundations’, ‘Exploration’, ‘Engineering’, ‘Deployment’ and ‘Post Project’.

4. What is DevOps explain the phases of DevOps?

Ans: DevOps is the combination of cultural philosophies, practices, and tools that increases an organization's ability to deliver applications and services at high velocity: evolving and improving products at a faster pace than organizations using traditional software development and infrastructure management processes.

The DevOps lifecycle comprises of the 5 Cs and these are as follows:



* **Continuous Development:** The first phase of the DevOps lifecycle, Continuous Development involves the phase in which all the planning and coding is done for the software application. This is the phase where the team members sit down and visualize the outcome or, in other words, how the software application will turn out to be. Once the vision is in place, the developers start writing and maintaining the code.
* **Continuous Integration:** After the phase of Continuous Development comes Continuous Integration, and the first phase automatically translates into the second one. This phase involves different steps which include planning of test execution to be carried out in the testing phase, as well as understanding the project requirements provided by the client to incorporate the necessary features in the software product. The stage also involves changes made into the source code frequently, and often on a daily or weekly basis. Many refer to the Continuous Integration stage as the heart of the overall DevOps lifecycle, as this is the phase in which problems in the code can be detected early on.
* **Continuous Testing:** The Continuous Testing phase involves the phase in which the developed code is continuously tested to look for bugs, defects and flaws. This is also the phase where the usability of the software is tested using the set of best practices for QA, and it is determined whether the software meets the specifications defined by the client. Continuous testing is carried out using automation testing tools, which can be open source tools like Selenium or advanced test management tools like QARA Enterprise and TestNG. These tools allow the QA team to execute multiple tests in parallel in order to increase test coverage and increase the efficiency of the testing process. The outcome from the testing is shared with the developers to make changes in the product.
* **Continuous Deployment:** The fourth phase in the DevOps lifecycle is that of Continuous Deployment and it involves deploying the code such that it doesn't affect the performance of the product or website. During this stage, it is important to ensure the code is deployed accurately on all the servers. It is all about ensuring the product is deployed with utmost accuracy, with no scope for errors.
* **Continuous Monitoring:** The fifth phase in the DevOps lifecycle is the Monitoring phase which involves recording and processing of key information about the application to deduce the areas where more efforts are required, as well as the general patterns. This one is an operational phase and the objective is to enhance the software application's operational efficiency, and also continuously monitor the performance of the application. Hence, this one is a crucial phase.

5. What is Jenkins and list out the various advantages of Jenkins?

Ans: Jenkins is used to build and test your product continuously, so developers can continuously integrate changes into the build. Jenkins is the most popular open source CI/CD tool on the market today and is used in support of DevOps, alongside other cloud native tools.

### **Advantages of Jenkins include:**

* It is an open-source tool with great community support.
* It is easy to install.
* It has 1000+ plugins to ease your work. If a plugin does not exist, you can code it and share it with the community.
* It is free of cost.
* It is built with Java and hence, it is portable to all the major platforms.
* Easily Configurable. Jenkins can be easily modified and extended. It deploys code instantly, generates test reports. Jenkins can be configured according to the requirements for continuous integrations and continuous delivery.
* Platform Independent. Jenkins is available for all platforms and different operating systems, whether OS X, Windows or Linux.
* Easily Configurable. Jenkins can be easily modified and extended. It deploys code instantly, generates test reports. Jenkins can be configured according to the requirements for continuous integrations and continuous delivery.
* Developers write the tests to detect the errors of their code as soon as possible. So the developers don’t waste time on large-scale error-ridden integrations.
* Issues are detected and resolved almost right away which keeps the software in a state where it can be released at any time safely.
* Most of the integration work is automated. Hence fewer integration issues. This saves both time and money over the lifespan of a project.
* Easy support. Because it is open source and widely used, there is no shortage of support from large online communities of agile teams.
* Rich Plugin ecosystem. The extensive pool of plugins makes Jenkins flexible and allows building, deploying and automating across various platforms.