Python

Module 1: Fundamental of Tech

Video: Foundations of technology

A comprehensive guide for it.

Beginners, I am Ravi, your course instructor.

In this course, I'm going to help you understand the need

of foundations, which are required for anyone who wants

to make their career in it.

Say you're enrolling for any of the IT courses, like

development related courses for front end, which includes

H-T-M-L-C-S-S, JavaScript

and backend related courses like Python, Java,

or platform related courses like AWS, Azure

and DevOps, and also latest internet courses

like artificial intelligence,

machine learning, and deep learning.

To learn these courses, you need to know few

basic software essentials, which are must

for any IT professional and will.

That's what we are going to cover as a part of this course.

Video: What is Application

Well, in simple terms,

an application is a software program, which is designed

to perform a specific task.

For instance, if you take Uber application,

it'll help you book a ride, isn't it?

And similarly, if I take few more examples

of popular applications like Google Maps,

Netflix, and Amazon.

Now if you talk about any of these applications,

they perform some task, isn't it?

For instance, if you take Google Maps, it provides

real time navigation, location services,

and helps you find destinations.

Similarly, if you take instance of Netflix application, it

helps you stream movies, TV shows,

and original content for entertainment purpose.

If you go with Amazon application, it helps you with

online marketplace buying the products

and selling the products.

So with these examples, we can confirm

any application we use, like Google Maps, for instance.

Netflix and Amazon,

all these applications are doing some task.

Video: Types of Applications

Applications.

We can broadly divide the applications into

the following kinds.

One is desktop applications.

Another one is web applications followed

by mobile applications and enterprise applications.

Now, what are these different types,

how they will change from one another?

Let's start briefly exploring them.

Let's start with desktop applications.

Applications that are installed

on your computer operating system, which are optimized

for local processing and used offline.

If I take few examples, Microsoft PowerPoint,

VLC, media Player, and Chrome browser.

All these applications we will install in our laptops are

personal computers, and we are going to start using them.

Let's talk about web applications now.

Applications which are installed on a server,

and these applications are accessed

with a web browser generally,

and these applications require internet connectivity.

Few examples include Netflix for watching

and streaming the movies, Gmail and Amazon.

Similarly, if I talk about mobile applications,

applications designed for smartphones

installed from app stores,

these applications often use device features

such as camera and GPS.

If I take example of WhatsApp,

it might use the camera module

to help you make a video call.

Similarly, if I take Uber application

in this particular scenario,

it'll use the GPS module in my system,

and then it'll help me going with the part

and start working with.

If I take Instagram as a use case, we go ahead

and use it for distributing our images.

If I talk about enterprise application,

these are large scale softwares,

helps in managing vast data

And integrating across different departments.

If I take examples, QuickBooks,

a popular accounting software, Salesforce

for customer management and S-A-P-E-R-P solutions.

And if I want to build any of these applications,

we need to know something called

as application architectures.

Video: Web app technologies

Architecture.

A three tier architecture is a widely used

architectural pattern.

It divides an application into three interconnected

layers called as sprinting,

backend, and database.

We'll go ahead and discuss about individual layers

first, let's start with front end layer.

The front end layer is also called as presentation layer.

The focus is going to be on the client side logic,

and the front end is responsible

for displaying the user interface.

Generally, the visual elements we see in web pages

like buttons, menus, logos, et cetera,

all of these are part of print end.

Moving to backend,

backend layer is also called as application layer.

The focus is on the server side.

Logic backend is responsible

for processing the user inputs.

Finally, we'll move into the database layer.

This is also called as data layer.

It is responsible for storing your customer data

and also retrieving the data, which is going to be used

by the application.

And if I take most popular applications, we have

similar architecture to go with.

I have given you an presentation over here

which shows the most popular applications.

For instance, take Google.

The most widely used search engine in the world

is having a architecture.

You can see frontend client side with the help of JavaScript

and TypeScript and going

with the backend technologies.

From the server side, we are using multiple

languages over here like CC plus plus, Golan, Java,

Python node,

and coming to the databases for storing

and retrieving the data.

And to perform data manipulations.

We are using databases like Big Table

And MariaDB.

Similarly, if you take Facebook the most popular

social networking site,

it's also having a three tier architecture.

And similarly, the front end is also JavaScript

and TypeScript coming to the backend languages.

We have Python, c plus plus, Java, et cetera.

And also when it comes to the database, you got MariaDB,

Maya, skill Cassandra, and other databases.

Same goes with YouTube as well,

which is the most popular video sharing site.

It also has the same three tier architecture

with front end being JavaScript and type Script

and backend being Python, Java, cc plus plus Go,

and database being big table MariaDB and et.

I.

Video: CRM Application Use Case

We do have multiple courses

like H-T-M-L-C-S-S, JavaScript, Python,

Java, DevOps, cloud AI, and Machine Learning part.

Now these courses can be growth together

on a broader term,

and these groups, we generally call them as stacks,

like full stack

platform stack and AI stack.

So these combinations can help you build any sort

of application to go with.

As I said, in full stack,

we do have courses like HT, ml, CSS, JavaScript, Python,

and Java In Platform Stack.

We do have courses like DevOps, data engineering

and cloud computing,

and coming to a stack, we have courses like

artificial intelligence, machine learning,

deep learning, and data science.

Now the whole idea is to connect all the stacks together

using an implementation.

And that particular implementation is going to be collection

of all this stacks with an application

to build a CRM application.

Now, what is CRM application?

A customer relationship management, a software tool

that helps the businesses that managing their customers.

Do you know, CRM is used by many companies

for following features, which can help them organize

and manage their customers in a better way.

Key features include lead management,

pre-sales analysis, opportunity filtering,

customers management, sales order vendors,

management, organization structure,

products, inventory, attendance management,

email templates, email and SMS marketing

and calendars.

These features allow businesses

to understand their customers

and provide the insights to the customer behavior

and act accordingly.

Few of the popular CRM applications in the current market

include Salesforce, CRM,

Zoho, CRM, HubSpot,

and yes, there are so many other softwares each in the same

domain, but these are the widely used

now coming to Full Stack team.

Our full stack team in terms of CRM application,

they're the one who are going to build the user interface,

the look and feel of the application,

and as well as the backend logic

which should implement the business

and as well as logical patch.

Now, why full Stack team needs the basics of other stats?

Let's talk from AI perspective to structure the data

for AI features that is providing better AI predictions

and from platform perspective to integrate

with DevOps pipelines for automated builds and deployments.

In the similar way, if we talk from platform perspective

in CRM application, this team is going to build

the infrastructure required so

that we can host the CRM system for global access.

And why do platform stack needs basics

of other stacks from AI perspective

to optimize the cloud cost

and AI workloads, for example,

GPU instances for machine learning

and from full stack perspective

to ensure the data formats match

what stacks and what teams need like JSON for API

to go ahead with the development patch.

And similarly, from a stack perspective,

we are going to build

and train AI models

for our chat bots.

Why a stack needs the basics of other stacks

from full stack perspective to deploy the models

into full stack ui,

which is integrating a chat bot into React.

And from platform perspective

to use DevOps tools

for containerizing the models using Docker.

If we take a real world CRM workflow,

this is a use case where a salesperson

uses the CRM to check a customer's risk

of backing out.

In this context, because of full Stack team,

the risk score appears on the dashboard,

which is built using React.

Because of platform stack data pipelines

fetch the real time customer data from AWS

and Azure environments.

And because of AI stack, machine learning models

can calculate the risk score

using AI agents.

This makes sense to understand how each team

or each stack is giving their insights.

Now, why cross stack basics matter?

If the full stack team does not understand the data formats

of the application, the AI predictions won't display.

If the AI team does not know cloud basics,

there might be a chance of

having the models overloading the service.

If the platform team ignores

full stack needs, the CRM could be close to slow

and very inconsistent.

So that's the reason why you need to understand the pitfalls

of working in silos.

If the focus is on full stack only,

you might have pretty ui, but no AI insights.

And if the focus is on platform stack only,

you might build robust service,

but there might not be any user friendly features.

And if the focus is on AI part only,

you might be able to build smart models,

but there is no way to deploy them properly.

So with this part, we want to give a final pitch

to all the learners who are about

to go with this particular course.

So stick stats are like instruments.

In an orchestra,

you might specialize in violin, like a full stack developer.

You might specialize in drums like a platform engineer,

or you might specialize in piano like an AI engineer,

but still you need to understand the rhythm

and harmony to create a masterpiece.

Same applies for our CRM application.

The CRM isn't built by only coders nor cloud experts

or data scientists.

It's all built by working all of them together

with each other.

So this approach shows

how foundational knowledge is required

and it is going to bridge the gap between different teams

and prevent project failures and unlock innovation.

I.

Video: Software Development Life Cycle

For developing any type of applications,

and that is called AS SDLC,

software Development Lifecycle.

It's a process followed by development teams.

SDLC consists of multiple stages, such as

planning, requirement gathering,

design, coding, testing,

deploying, and maintaining.

Now let's try to understand these different

stages in detail.

Let's first talk about analysis.

In this phase, the project team identifies

and gathers all the requirements for your application.

Then understanding the need of users,

then prepare the features

and functionalities expected from the application,

and looking into any specific constraints

or limitations present within the project.

Now let's talk about the design.

So once the requirements are clear,

the system design phase begins in this phase,

the architecture

and overall structure of the application

is defined.

It also includes designing the database schema,

user interfaces,

and other various components like load balances,

connectivity from web servers to app servers

and apps, servers to database servers.

This will be part of system design.

Now let's talk about implementation.

The implementation phase actually involves the coding

of the application based on the system design

given earlier in here,

software developers are the ones who write the code for each

specific module, ensuring it

aligns with the design

and meets the specified requirements.

As you can see all the code written.

Now, let's talk about testing.

Testing is a crucial phase

to ensure the quality of the application

and different types of testing are performed such

as unit testing, integration testing, and systems testing.

And they're used to identify

and fix bugs and issues.

Now let's talk about deployment stage.

Once the application passes all the testing phases,

and if it is considered stable

and ready for production, it is deployed

to a live environment where the end users

can actually go ahead and access the application.

Once the application is deployed,

we'll talk about maintenance phase

after the application enters the maintenance phase.

Ongoing support is provided.

Any issues are bugs reported by the end, users

are addressed in this particular phase.

And additionally, updates

and enhancements will also be made

to improve the application performance.

Now these phases are common for any kind of application.

We go ahead and work in the software development part.

Now, to apply all of this,

we need some application development methodologies.

Video: Application Development Methodologies

Structured approach to manage

the software development process.

I'll take two different methodologies.

One is called waterfall methodology,

and another one is agile methodology.

Let's try to understand both of these methodologies

and see how they can help in constructing an

application first.

First, we'll begin with waterfall methodology.

It's a linear and a sequential approach

where each phase must be fully completed

before the next phase begins.

And as you can see in the image

provided, we observe a sequential flow

means first we take the requirements, then go ahead

with a design, then development, testing,

deployment, and maintenance.

Now let's see the key characteristics of

waterfall methodology.

The phases are in order, like requirements,

followed by design, followed by implementation, followed

by testing, followed by deployment.

And finally, maintenance.

Fixed scope meaning requirements are

defined upfront and they're rarely changed.

Documentation driven where we have detailed

documentation at every stage.

Rigid structure, limited flexibility

for changes once the project starts up.

So these are the characteristics of waterfall methodology.

Now let's discuss the pros of waterfall methodology.

Simple to understand

and manage clear

milestones and deliverables.

Ideal for projects with stable

and well-defined requirements

and projects with more predictability.

Now let's discuss the cons

of waterfall methodology.

No room for feedback until the testing phase is done.

High risk of delays if requirements change

testing occurs late,

which increases defect resolution cost.

Now let's take an application

where waterfall methodology is more suited.

We will build a payroll system

with fixed government regulations

where requirements are stable

and not often changed once fixed.

So a payroll system would be a better choice of application

per waterfall methodology.

Video: Agile Methodology

And flexible approach

that emphasizes on collaboration

and customer feedback.

The following are the core principles

of agile methodology, also called as

Agile manifesto.

Here we focus on individuals

and interactions.

Rather than focusing on processes

and tools, the emphasis is on working

software over comprehensive

documentation, more customer collaboration

over contract negotiation,

responding to changes instead

of following a fixed plan.

The key characteristics

of agile methodology,

iterative development, which delivers the work

in small functional increments called as sprints.

Continuous feedback, regular

input from stakeholders,

adaptability, graceful towards changing requirements,

and cross-functional teams where developers,

testers, and business analysts collaborate daily.

Now let's discuss the pros of agile methodology.

Faster delivery of usable features,

frequently going with updates, improved

customer satisfaction,

and reduced risk of project failures.

These are all the good things about agile methodology.

Now let's discuss the cons of agile methodology.

In here, we require active customer involvement.

We have less predictable timelines

and also less predictable budgets,

and also lack of long-term planning.

So these are the cons of agile methodology.

Now, let's take an application where

agile methodology is more suited for

developing a mobile application where

user needs evolve rapidly with often

changing requirements.

So developing a mobile application would require

a agile methodology approach to build a software.

With this, we also need to understand about

agile frameworks.

Agile frameworks are basically structured practices

that operationalize agile principles,

offering specific workflows, roles,

and tools to manage the

projects flexibly.

The popular Agile frameworks are

Scrum and Kanban.

Left one is Scrum.

The right one is Kanban, which we should

discuss in detail.

Video: Scrum

In Scrum, we have sprints.

Sprints are time boxed iterations,

which can be in between one to four weeks.

In here, we deliver a working product increment.

Following our scrum roles, we have

scrum master product owner

and development team.

A scrum master facilitates the process

and removes the blockers within your project,

whereas a product owner manages

product backlogs and he will prioritize the tasks.

Coming to development team,

it's a self-organizing group that delivers the work.

Similarly, we do have scrum artifacts,

which are product backlog, sprint backlog,

and product increment.

A product backlog is a prioritized list of features.

A sprint backlog are the tasks selected

for the current sprint product

increment a shippable product version

after each sprint.

Let's also discuss about scrum events,

where we have sprint planning,

daily standup,

or also called as daily scrum

sprint review and sprint retrospective.

First, let's understand what is sprint planning.

A team defines sprint goals

and select the tasks from the product backlog.

A daily standup is a 15 minute sync up

to discuss the progress and also obstacles present.

And we have sprint review

stakeholder demo of the sprint deliverables

to gather the feedback.

Sprint retrospective teams review

sprint process to identify improvements.

Let's discuss about Kanban.

Now, in Kanban,

we use both, which will have visual

workflows like in to do in progress

that is doing and done to track your tasks.

Work in progress limits

restricts multitasking to improve focus,

and we have continuous delivery.

No fixed hydrations tasks

flow continuously.

Now let's summarize

the scrum framework by taking all the roles

like artifacts even

and also scrum roads with respect

to A CRM project

and login module functionality.

In here we see different things we discussed in agile

with respect to this particular framework.

We got scrum master, product owner

development team, product backlog, sprint backlog,

daily scrum sprint review, and sprint retrospective.

And finally, the output.

Let's see about each one in detail.

First, let's talk about scrum Master.

Scrum master ensures the team follows agile

practices and removes blockers.

Next, let's discuss about product owner.

Product owner manages the product backlog

and prioritizes tasks.

Defining a login requirement.

For example, we want to design a login page,

which must support two factor authentication.

In that regard. When we talk about product backlog,

a prioritized list of features.

For example, add a secure user login

to CRM is going

To to be a prioritized user story.

Now let's talk about sprint planning.

Team defines sprint goals

and selects the tasks from the product backlog.

Coming to sprint backlog, the tasks

selected for the current sprint, which are going

to be picked up by development team,

development team, commits to coding the login feature

in a two week sprint.

That is building the login user interface

backend authentication and testing.

And for this, a daily scrum meet will happen,

which is a 15 minute sync up to discuss our

progress and obstacles.

Example, login UI is completed.

Now working on authentication.

Next we got sprint review, a stakeholder demo
of sprint deliverables to gather feedback.
Here it is nothing
but giving a demo
of the working login page to your stakeholders.
Then we have sprint retrospective.
The team reviews sprint process
to identify improvements.
Example, speeding up API integration
in the next sprint, and then final output,
a shippable login feature with email,
password authentication, error handling,
and security checks.
Now say we have
completed the entire CRM application in this manner
so that customers can start accessing the
application features.
Now.

Video: Data Fundamentals

CRM application say, we have completed
the development of CRM application
and we want to host the system so that
customers can start accessing the application.
Now, in this context, I'm going to help you understand
how data and compute come into the picture.
Now, let's connect data fundamentals
to our CRM application.
Let's discuss what is data First.
Data is collection of facts.
Figures are symbols that can be processed
and analyzed to extract useful information.

Now let's see how to categorize the data.

We have structured data,

semi-structured data,

and unstructured data.

These images will help you understand

how there is some sort of connectivity

between the data in the first and coming to the second.

We are missing something and coming to the third.

It is completely different.

We'll go ahead and understand about each type in detail.

Let's talk about structured data first.

Data is organized in a predefined format

in row and columns.

Examples include databases, spreadsheets, and CSV files.

In the context of CRM application,

our customer details will be stored in

SQL database

and sales transactions can be stored in

spreadsheets and CSVs.

Now let's talk about

semi-structured data here.

There is some organization in the data,

but it is not as rigid as structured data.

Examples include J-S-O-N-X-M-L log

files, et cetera.

In the context of CRM

application, we can see

The logs of user activity

are vendor contracts.

Now, let's talk about unstructured data.

There is no predefined format.

There is no structuring in the data.

If I take some examples, like text documents, images,

videos, audio files,

and in the context of CRM application,

email templates can be stored in the PDF format.

Your customer call recordings,

which are audio files in the MP three format,

and your customer images may be in GPG format.

And with this we can say that data can be in the form

of structured, semi-structured and unstructured.

Now, let's try to understand

where this data is stored.

It can be in databases

or it can be a historical data,

which is there in warehouses,

and you might opt for cloud storage.

Now let's talk about each individual of them

and we'll begin with databases first.

Databases are used to store, organize

and re review the data

of your applications efficiently.

Databases can be categorized into SQL

and no SQL databases.

SQL databases can store leads

and customer information in terms of CRM application.

In a relational database like MySQL,

Postgres, et cetera,

no SQL databases can store the vendor's data

in a non-relational database like

no SQL databases, which includes MongoDB,

dynamo db, et cetera.

Coming to data warehouses,

data warehouses are used for storing

large amounts of historical data

for doing analysis and getting

Business insights.

In here, we take data sources from multiple entities

and we go through ETL process

and move the data into warehouses

for reporting analytics and data mining.

Coming to cloud storage,

it is an on demand storage used

to store files, backup, et cetera in the cloud,

which provides high availability and scalability.

Cloud storage options include S3 Blob

and Google Clouds.

This can give you cheap cost storage

and providing the benefits

of high availability and scalability.

Now moving ahead, we have different

techniques for going and managing data.

First, we'll begin with data analysis.

It is used for extracting insights,

identifying trends, and making predictions.

The techniques we use here are basic statistics

and data visualization tools.

Used for performing data analysis are

Excel and business intelligence tools

like Power bi,

and we also have one more stream for managing the data

and advancement called as data engineering.

This is used for building

and maintaining the infrastructure for collecting,

storing, and processing the data.

The key concepts of data engineering are ETL,

extracting, transforming, and loading the data.

We use tools like Spark, Databricks,

and cloud data services from Amazon and Azure.

Video: Cloud Computing

With our CRM application,

as per the earlier discussion we had on data.

Now the question is where this data is stored

and how do we process this data to store

and process the data?

We need storage and compute par.

So in simple terms, it is nothing

but the CPU ram, which is memory

and disc storage for our applications.

Here are the key computing technologies.

You should be aware of CPUA

central processing unit,

which is a general purpose processor

for a wide variety of tasks.

For example, to run CRMs, core features

and user authentication

and transaction processing.

We need CPUs

and coming to GPUs, which is graphics processing unit.

These are specialized for parallel processing,

which is ideal for AI training

and graphic intensive tasks

to train our AI models.

For example, like in the customer relationship management

for lead scoring

and email templates, suggestions, these components are going

to be used and also we have something

called edge computing,

which will be processing the data

closer to the source to reduce latency.

We use Edge Service to

cache the application data in here.

The CRM data will be available

for faster access in regions like Asia

and Europe, where our edge servers are going to be kept.

We can traditionally manage the storage

and compute on

traditional, on-premise data centers,

but the effort time

and capital expenditure is quite high in working

With on-premise data centers.

So we generally opt

for cloud computing nowadays in the modern approach.

Now the question is what is cloud?

Cloud computing is on demand availability

of compute resources.

We discussed earlier like CPU,

ram, hard disk, GPUs, machine learning models,

all these particular entities are available over the

internet first so that you don't need to put lot

of effort, capital,

and time in managing these particular resources.

Now let's understand how cloud offers their

services, and we call them as cloud models.

One is infrastructure.

As a service model IAAS model,

we got platform as a service PAAS model,

and we have software as a service SAS model.

Let's talk about each of these models.

We'll start with infrastructure as service first.

We offer basic computing resources like servers,

storage networks,

and infrastructure as a service is generally used

by infrastructure teams where we host

server storage

and network, a very low level entity.

Now, let's talk about pass model platform

as a service offers operating systems

and middlewares for developing, running

and managing applications.

Past services are generally used by development teams

where we utilize the platforms

to build our own custom applications in a easier way.

Now let's talk about SaaS model.

In software as a service,

we directly provide a software,

which is basically a package software,

which is ready to use.

And these packages can be directly accessed over the

internet by our end users.

So in terms of SaaS directly, we go ahead

and consume the applications rather than building the

platforms are hosting the infrastructure.

Now let's talk about popular cloud platforms,

which are available in the market.

AWS Amazon Web Services from Amazon,

and we got Azure from Microsoft

and we got GCP Google Cloud platform, which is from Google.

And these three are currently the most

in trend in the market.

Now let's talk about cloud computing benefits.

We have reduced costs, we have better security,

we have scalability and easy disaster recovery.

These benefits help the customers

to move from traditional data centers

to cloud data centers, and that's the future.

Module 2: Python Introduction

Video: Python Introduction

Hello, good afternoon.

My name is AI here.

I'm China for the Python

and Data Science and data Engineering.

So here are the main agenda.

What you are going to learn as part

of the Python program is programming.

First the basics of Python programming fundamentals,

then advanced python like uh, object oriented programming

and some special libraries like, uh, and

and we have frameworks within the Python that is

for application developments, zago and fast, PI

and flask, such kind of things.

Okay, so let me start about the Python.

So among all languages nowadays,

Python is becoming more and more popular.

So what might be the reasons for the Python like growth

and python popularity?

Basically two reasons. We can say.

The first reason is ease of learning.

You can say it as a easy to use also.

So this is a right option, a way to, so like, uh,

for the beginner to start learning a programming language.

And second thing, Python is having a rich set of libraries

for different purposes.

Python is having predefined libraries.

So each library consists of some predefined functions.

Depends on your use case.

You can input that library and you can call that library.

So without writing any lengthy of the code.

So because of the first feature, like a code, uh,

like a development time

and uh, length of the code is reduced

that way if you use libraries like this, okay,

still deliver dollar process will be very simple.

So these are like uh, the first high level benefits,

but internally, technically it supports some other features.

So here are the Python key features.

Python is a high level language,

also interpreter based language.

In case of interpreter based languages,

you type set up statements in the program

statement by statement.

Execution will happen. For example,

if any is a statement is a like encountered error.

So what I have written enough for four,

our file lines sub statements.

Third line is having error.

So first statement one will be executed

after that statement to when our control teach two part

statement, the program execution will be stop.

So whenever it got stopped, then you'll identify that client

and you will rectify that line.

Again, you'll read on the program again,

program execution will be started from

beginning to after year.

So this is the side of interpretive based languages.

That two, we said one word here, high level language.

So high level language means actually

that there are different language like uh,

low level languages and uh,

middle level languages

and high level languages, especially if you go

through the high level languages,

this high level language similar, uh, similar to

how you are speaking your regular English

with a simple English, a small example.

For this I want to print some message.

A statement in Python is like print out.

Okay? So simply you are asking to print some message.

The message is here that is called

that much simple language.

The Python is, so here one of the key feature pass.

Let me install the key features.

After that I will explain each and every one.

The first feature, high level language

and also it is interpreter based language.

PowerPoint is ease of learning.

Of course it's not a technical

benefit, it is a user benefit.

And one of the important key feature here, dynamic typing

is dynamic typing.

You can technically saying yes, dynamic data typing

match benefit, extensive libraries.

In the beginning we discussed like which set of libraries

can call it as extensive libraries

for different purposes.

And also it is supporting cross platforms

and latest programming ing.

That is object oriented programming,

supporting group feature

oriented programming.

Finally, this is edge.

So these are the major features of the Python.

All these features will be helpful to a developer

and to the industry step by step.

We'll be discussing already I

explained to you about high level language

and what is interpreter based language.

So easy to use means when compared

with other languages it's more easy.

For example, you are achieving some specific task in Java

by writing hundred lines of code,

same functionality you can achieve just a

for pilot in the file.

But the statements are very simple, okay?

And the technical discussion is starting from the dynamic

typing point of onwards.

Okay, I'm starting. That is uh,

discussing about a dynamic typing

in any language.

We are sorting values into variables.

So variables are like the value holding points.

For example, I'm as assigning a value.

I'm declaring a variable here

for this, I'm assigning a value.

This is a Python programming study.

We assign any value to a variable

in this when you compare with other languages.

Whereas in other language what we have to do faster data,

type out that variable name

first we have to declare the variable

also directly can assign a value license after that value.

Example in C

or c plus plus of Java, JavaScript, such kind

of languages you are as sending variable in this type.

So was the NTA equal to.

So definitely how to specify data type

after that the variable net and then finally the value.

Whereas in Python, look into the Python.

We are not declaring any data type based on assigned values.

Automatically data types will be created here.

So here the data type here will be in digital.

For example, if I say something like a B question

10.25, automatically data type of B will be

float suppose C equal to LOSL

or D equals to ISL.

You can use single course

or double course in the Python program in both the cases,

the C and D are here.

String data types. So this is called simply the

dynamic, the feature called dynamic data type.

Okay? But to the next benefit

we said like uh, which set up libraries

I will open fresh, right?

For different purposes

and different use cases of the industry.

We have different predefined libraries.

So here library is a collection of

Python classes.

That means python predefined classes

in object oriented programming.

We design the blueprints by using classes later.

Once we step into object oriented programming,

we'll be discussing about this one

and functions and some

So actually the word libraries a non-technical.

In fact, technically there is a concept called package

set models.

By using these packages models, we create libraries.

The layman term is library.

Technically in Python we call them as packages.

Under packages module, under modules,

we keep classes, functions, variables.

But I'm saying that with set up libraries means all these

libraries are pretty redefined for different purposes.

So what are the redefined libraries that we have

for different purposes?

For example, I have a scale up.

This is machine learning.

Artificial intelligence people and data science people are

using machine learning.

But to implement machine learning algorithms,

writing the code from scratch is very tough.

Very lengthy programs you need to write.

If you use the psyche learn, that means a scale

and library you can implement,

you can try your machine learning models very easily.

Okay? So similarly, if any numerical calculations like uh,

MPA, metrics related competitions.

You wanted

if any metrics matrix specific competitions you want.

There is a library called mpa.

MPA means numerical python.

Example. If you treat like a yes one,

matrix B is one matrix.

I want to multiple, I want multiplication of the matrix.

Simply I can say it B simply one single line

of expression, okay?

The matrix multiplication kind of complex computations.

Also you can do very easy for example,

very more complex computation.

The ES metrics in version.

Suppose if you treat that as a matrix,

I want matrix in version simply INV function.

Is there INV of yay.

If you say like this matrix computation will be in this way.

I have library for machine learning.

That is a scale library for numerical computations, not by,

I want to pre-process some data like a data cleaner.

I want to extract data. I want

to perform some data cleaning activity.

I want to perform some transformations on the data.

Finally, I want

to find out some grouping aggregations for all these things.

I have library, I have a library here on us.

In this way, different purposes,

different libraries are available.

So depends on your role into the organization.

You can use them. Suppose if you are a data analyst

and number, you'll be using, for example,

you're a data scientist, psychic line and number.

You will be on the, so the,

this is a another key feature of the Python.

And next one more thing, it is supporting cross platforms

here, screen study

cross platforms.

What are the meaning of cross platforms?

For example, I one application or one program?

It's a program one in Python generally

for the Python programs figure.

So this program developed in my Windows system,

windows operating system,

but IT test desktop sits here.

The test operating system might be different.

So suppose he's using some Mac,

same pro Python program without doing any changes,

you can apply into this plan and you can execute them.

So finally our production is supposed in Linex.

So from the production environment's only like

with application we'll be live in today end users,

all end users will be interacting

with this production informance.

So once this test is passed, same Python program,

without doing any code changes,

that is applied into production.

Actually I developed my program in Windows tester,

has tested it in is Mac.

And then in the production I have linex operating system.

The linex also my program will be

that will without doing any.

This is what the feature of cross platform in Java.

Same feature is called platform independent.

I want more last feature

versatility in the program.

That means Python use cases are different

but different use cases.

You can use Python.

That means different types of applications.

You can use Python for example. You are a gaming developer.

You can use py. You

wanted to do data visualization just like a power BI tablet.

Such kind of visualization tool.

Similarly, you wanted to do data visualizations,

but the visualizations,

you have different libraries like Plotly, CO,

such kind of libraries

that the interactive dashboards you can order

by writing a simple port interactive

D powerful interactive dashboards.

You can write, you can convert them into HT ML files ladder.

These HTML files will be migrated into web applications

or mobile applications.

And suppose I wanted all of web frame web have application.

So all of the web applications have different framework.

Plak Z,

actually spelling is tgo, but ation is just Z.

Okay? Not as most popular web framework

that is post A, right?

So these are the key features.

And also one more thing,

one final feature I'm highlighting when compared

with other languages, this feature is available only

with the Python.

The Python supporting multi programming paradigms,

multi programming

paradigms.

That means a different ways of programming.

The word programming.

Paradise means different ways of programming

or different styles of program

where our styles,

so in the market, different programming,

paradises like imperative programming.

And one more category is functional programming,

other programming style declarative programming.

And most widely popular programming style is two.

That means object oriented.

Let's some market popular languages,

programming, of course all these features, all these kind

of paradigms are supported by the Python.

In any style you can write your applications,

you can in any style you can write over withs.

So for the programming, the whole is like a C, C plus plus.

C is the best example. C global, such kind

of things you can say as a integrated program instance.

After that c plus plus

and Java, these things comes under C plus plus

or Java or IT dotnet C,

innovative and other languages.

These things comes under object work.

And if you talk about the S scale.

S scale comes under declarative program

and functional programming especially,

there is a big data framework for the data processing

that is spark within that most wide used languages scale,

actually stellar language is developed by Java,

but programming style is functional programming style.

And also before Sklar,

there was another language called hasken.

It is so many world and new languages are,

but each language is following their own

programming paradise.

But luckily here Python is supporting all types

of programming.

Paradises a simple demonstration,

a simple code representation for you.

For all these,

let's look into programming,

how Python is supporting this.

Just look into this, forget about the statements

and syn access just to try to concentrate

what is happening at each line.

I'm taking a collection. So in Python this kind

of collection is released.

This is a collection of elements. There are five values.

I want to find out this sum, I'm writing my own logic here.

Initially, total value something zero.

I'm taking of that sum follow I'm taking for I in R

for V in X.

How many elements are there in the X? 1, 2, 3, 4, 5.

So that automatically is low.

It rates for five at each iteration.

The value is coming into V.

After that we are instructing what would do with that V.

I'm going to add this. V two V, previous variable product

four plus C equal two V.

That means to the exit value of the total.

I'm adding this B now just look into this

small logic explanation.

Before loop, total value zero.

In the first generation, what happens?

Total plus equals to we.

You said then zero plus 10.

This 10 will be stored total in the second ation.

Total value already 10. Now for the 10 20 is added.

Now latest value of the total 30

in the third iteration, total value 30.

But 30 ation value is what? 30. 30 plus 30.

That is 60 in the fourth.

Ation already have 16.

The total, the 40 is added.

Now the value hundred in the fifth ation

total is hundred.

Now 50 is added. Total one 50.

Now high ation ised, but loop got complicated.

Finally you are total,

the sum of the list is total.

You'll get one 50.

Now here observ, in the programming we have written

five, four steps.

1, 2, 3, 4.

This is a 3.1, which is a SubT statement of this part.

This is the power style of I program.

Instead, Python fundamentally follows

programming Paradise

of course as a beginner

fast write program programs in.

So according to this explanation, what exactly the meaning

of em, step by step instruction at each, at each step, what

to do, what should be done.

We are giving instructions just

so when you submit this program

or when you execute this program, as for our instructions,

that the program line by line way basically, okay,

of course this kind of bism is available

with all the languages, with all word languages.

And there is uh, one more uh, programming style here.

What is that last we discussed about

the second programming style

is functional programming style.

For example, same list I'm taking here.

Previous values, 10, 20, 30, 40, 50.

This list, this is a collection. I have five elements. Okay?

For each element I want to, I want to add a hundred.

I want for each element, I want to add a hundred.

For example, for the 10,

if I add a hundred, what is expected?

One 10 is expected. Same for the 2000.

I wanna admins that will pick up one 20.

So generally to do this in I

style, you have to write a funnel.

Okay? In the functional programming style,

what is a way just observe.

That's a wise quality map

of lambda.

XEX plus input is thanks.

Already here, the name A is here just to check,

give a different name.

Automatically each value of A is coming into a for the A.

What will happen? The

ation a value.

10, 10 plus this will become one.

This kind of programming style is

called functional programming.

So based on this, Python is supporting reto programming

and also functional program.

Got it. Nowadays, uh, for some typical logics

and complex, uh, logics,

dysfunctional programming is most widely used, okay?

And there is another style of programming.

Also declare I do program.

Uh, this is the third programming paradigm. Declare.

I do program generally as all your skills statements,

all this kind of model.

So about in the case of square databases,

we keep our data into tables.

Suppose our table is employee. We have some of employees.

Suppose if I want all employees

who sell this greater than one line.

Generally in sql, what is your style?

You are not going to write any formula in sql.

The programming style is like select

start from what is the column name?

Yeah, where, what is the

what employees you want who sell the greater than Oracle?

So here we did not write any far loop.

Understood. So suppose data file plus you need to, you need

to read the data from the file.

All the records criteria to write Far loop.

I teach it recently how to scan one record

and how to check whether that is greater article

that a thousand are one laptop.

But here he did not write any follow simply from EMP

where S is greater than article two one.

This kind of programming style is called declarative

programming style in Python.

Also, this kind of style is available. In what way?

Just absorb. Suppose my data is in on file,

which is employee T.

I'm creating on data frame here.

3D for

read, CSV.

I'm going to read data from that file thing

that I have thousand regards here,

thousand employees listings here.

What I want, I want two s greater than Oracle two.

One like simply year of year

of salary

greater than Article two.

One, like I said, I got

only the implies two sal greater than Article two,

one lack, okay?

Here also, I did not write in one

this kind of programming style.

Also supported by, by that you program

one more four, programming paradise, which is more popular,

more widely used up widely used in

all application developments.

That is object to oriented program

about the class and about object.

Once we step into the object work,

the programming more deeply we'll discuss.

But now just a rate. The class is a blueprint.

Object is a instance of that blueprint. Okay?

So here the programming style is like this.

Some I want to create some employee plus class

EMP, okay?

And I'm creating one constructive here.

So forget about this construction and all these things

because if I start explaining this again,

it'll take another one hour time.

So that's why like I don't want

to go into the paper about this object oriented programming.

Now just observe that programming style.

Here I'm constructive method for this.

I'm going to pass some values, some arguments.

Idea of the employee name. Of the employee.

Salary of the employee.

So here I'm going to ascend.

Values equal

to some Id sell.

Do name equal to name

sell is equal to sell.

And one more. If any other method, like I want like a define

suppose add tax, the tax related functionality.

I will write and define add HRA,

the HR related functionality.

I write this kind of programming

style at Paradise called object oriented program

based on all these explanations, totally,

we have seen four paradises here.

One is

second one is

declarative, of course.

Second, we discussed as a functional in the thought press,

I'm writing as functional

and finally a single programming language.

Python is supporting all these features. Okay?

Being a developer, sometimes I code in the integrated style,

sometimes declarative style, sometimes functional style.

In most cases I use object rated profile. Okay?

So it doesn't matter whether you are a new one to the Python

or already you are, you have been into this python.

It doesn't matter from basics to hero level.

We'll be teaching you everything with simple examples.

That means with the simple, basic examples

of two master level examples.

Okay? Let's have a journey

once we started the course.

Okay, we'll get the invitation. Let's meet in the class.

Bye-bye. Thank you all.

Thank you very much. Bye-bye.

Video: Coding Rules

Start our journey with very basic.

I'm going to start off with uh, Python coding rules.

First, a simple execution.

I'm going to demonstrate I'm opening Google collab.

So Google collab provides you python runtime, okay?

Without installing any stuff, we can practice directly.

Okay? So in the form of notebooks,

we'll be writing Python coding.

This time starting a new notebooks.

So I'm going to connect with runtime after here.

After here, one statement.

I'm new, this is my first step

and one more statement.

If I execute this, the first line is perfect,

but second line I got at,

okay, that means I did not follow coding ruling

here after here.

I'm going to modify this. Now it is wrong.

So what is here? This is rule number one.

You can say it as a indentation.

The rule one is each python statement

should start at

column position one,

ELL or code editor.

This is rule number. So if I give

like this, the first one is value

hash is the comment line here.

Second is inval.

I keep this into our notes.

Look into rule number two.

Aha. Please observe this. So this

is the variable declaration style in the python.

What is the variable name here? Y the value tag.

You are as assigning 10 to the yin.

No need to say you need data.

Type in Python based on assigned values.

Automatically data types will be created.

Okay, so let me modify the first line.

First sale A value

10 B value 20.

I'm going to write some condition A greater than

B condition.

True or false is true.

If the given condition is true, I want to print something.

Suppose given condition false

Print B is big.

If I execute this,

I got error totally.

Here in the code there are six slides

at which line you got error.

Third line. So here the rule is

if it's a condition, whatever the condition you give

condition can be true.

Condition can be false. If true,

I'm expecting some sub statement.

If false. Also I'm expecting some subst statement.

If any statement line is expecting a SubT statement,

the line should end with the column understood.

Now I said Colin here at a risk clear again, running this,

I got one more error at the L spot

because LC is also expecting a substrate.

That's why a l also, I have to say,

okay, now error got cleared.

Now what is the biggest value here?

So validate this condition. 10 greater than 20.

Rule or false? False.

Because of false you came to the control, came to yes.

So this sub statement is executed.

For example, I'm giving 30.

Here I'm giving 30

a value, 30 B value 20.

What about this condition to our false?

Because of truth, this statement will be executed. Right?

So based on this, what is the rule number two?

The rule two, if any,

he is expecting sub statement.

The line should end with

Code.

So example for this, this is the quote, right?

Okay. Now I'm coming to the rule number three.

Once again, I, I'm repeating the same

B 200 if you greater than B.

20 is B. Yes.

Again, never see it. What is the title error? Fourth line.

In the fourth line is a

main statement or sub statement.

Sub statement should be in a forward position to parent

at least one space, two space, three space.

Your wish. Suppose the parent statement

is starting at column one.

Sub statement should start at minimum two, two

or three or four your wish.

Understand. Come on, do the same thing here.

Y is also expecting a sub statement so

that the sub statement should start at in a forward position

to parent clear.

Got it. Please comment. The rule three,

sub statement

of a parent line.

Should we fail

forwarded Which question?

Two parent lines

start, which

I'll have you clear about this.

Let's go with rule number four.

Now, once again, I repeat all these things for you.

We value a hundred be value 200

later will modify the values my condition,

yay, greater than B.

Say that print yay is big.

If yay big means who is small, which is small.

B. B is small.

Upper here, wanted lamb keeping like this?

Yes else if the condition

enforcements B is big,

yes.

Small.

Again, I got, well I have only one

sub statement, condition two or false?

I have only one sub statement here.

Multiple subs statements. That means more than one.

What is the error here? Here?

Subs, statements started at which equation?

1, 2, 3, 4.

At column version four, the fastest sub statement started.

Second sub also should be started in same column.

Understand, come to the Ls here.

L fastest sub. We started here.

Second also should be in same column position.

Understand.

Now check this. Yes, perfectly.

Got that please. As for this, what is the rule number four?

All sub statements of a parent line

should be in same

column position.

Got it. That is all.

Suppose here these two are sub statements of the

these two are the sub statements, sir?

Y Okay. Yeah.

Started at column position four.

No need to start Y spot in the column four.

You can start with column three Ls,

but all sub statements

of the deal should be in same understood.

These four rules we need to follow.

Module 3: Python Installation and Setup

Video: Installation and Setup of Python

Hello everyone, SMI here.

Welcome to the Python session today.

I will guide you through the Python

installation and the setup.

Okay, so let's move to the installation.

So in your pro, just type Python.

Okay, so Python dollar we are getting right,

let's click that one.

You are getting a official website on top only This one,

ww python org G, we have to click this one.

So here you can select, select

what are the version you need.

Otherwise you can select the latest version by Defaultly.

We are getting the latest version.

You can download that latest version.

So now we have a latest version

of the Python is three point 13.3.

Okay, just click on this.

Uh, download Python three point 13.3.

If you're seeing, it'll download here. Okay, it's completed.

Let me see in the folder.

Okay, so just double click on these things.

This will be available 11, file Explorer. Okay,

you have to select these two things.

Use admin and add Python point EX, C to path.

After this, you have to sell this custom installation here.

Make clear all these check boxes as to build up. Okay, next.

And if you want you can sell it any one times.

If you want any, anything you can uh, choose in a path.

Otherwise you can use this by default pattern.

So it'll be stored in C drive in a users,

you have to click S.

So it's starting installation,

it'll be take little bit of time.

So it's uh, successfully completed that green bar.

So set up also successfully completed, just close.

Now we have to check whether the Python is installed

successfully or not in our system.

So how can we check that one?

Ms, just type in your search bar command prompt.

Here you can see the command prompt, right? Open this one.

So to check which version of Python you installed is

that Python is installed

or successfully in laptop, just type Python

space hyphen version.

Yeah, so we got Python three point 13.3, right?

So this is the latest version.

We downloaded the same version. It's fine.

Now how to run a Python code

or how to execute a Python code in never command prompt.

I will show you now just to type fee Y.

Okay, so I'm writing a basic line of code.

Hello? So it has to print D. Hello.

So here you can see, let me show you another basic one.

A equal to 10 B, equal to 10

B equal to A plus B.

Okay, so I'm printing C Here we got answer 20

A is a 10, B is a 10, 10 plus 10 20.

That will be goal two C, right?

So like this way we can uh,

execute our Python code directly from the command prompt.

If you have a Python in a local system

and the next we can have this id ld also,

if you want some specific night, uh, white notebook here,

you can run your Python code.

Yeah, ideally shell, so here you can see

what is the version we installed.

So same thing how we did in a command prompt.

Like that way we can do here also.

Okay? So like this way equal to can, I'm giving

print a,

Okay, fine.

So let, let me,

so like this way you can uh, you can do all these things.

Okay?

So like this way you cannot do same thing

what we did in a command room.

Same thing in it ideally also. Okay fine.

So we completed our Python installation and setup.

Now we'll move to the Anaconda installation.

So Anaconda also plays a very important role when we are

dealing with the Python things further.

I'm again going to Chrome. I'm typing Anaconda download.

So here we are getting the official website of the Anaconda

for download purpose ww anaconda.com.

So here you have to do skip registration, okay?

If you're giving a mail id, it's not a problem.

Better go with this Skip registration.

Then you have to sell it Mac window or Linux. Okay?

So I'm selecting the window.

If Mac means you can sell it here.

So let's see. So it'll take a little bit time.

We will wait for that. So Anaconda is a platform.

In this platform. We will get all this uh, spider R Studio

and the Jupyter Notebook.

All these things related to the IDs

and notebooks of the Python.

We will get all these things in it. Anaconda platform.

So if you install the Anaconda platform,

we will get all these things at a time in our system.

We don't need to install anything separately,

just we need to install Anaconda.

Okay? So successfully we are done with this installation.

Let's wait for a few minutes.

Has to show the folder icon, then it'll be done.

Okay, so now it's completely done.

I'm clicking again, this will showing a folder here.

You can see we got Anaconda. So double click on this.

So click on this next.

Then you double click on the Anaconda and your file manager.

You will get the popup like this here.

You have to click next here.

You can read this Terms and condition. Just click. I agree.

This is a your recommendation.

If you want to install for just you, you can use all users.

Means you can use all users.

I'm selecting for me, just me, okay?

So as like, uh, in a Python,

I already told you if you want to select any specific path

to store the Anaconda system, you can browse

and you can select that path.

Otherwise, it's a by default please stored in the C drive

and the users next.

Okay, so, okay,

so I'm selecting here normal C Drive.

Let me create one uh, folder in this C drive.

Yeah, make new folder

moving.

Okay.

Okay, so now you can select this first and second one.

So these two, uh, sorry, first

and third one, these two check boxes is a compulsory

remaining thing is based on your computer

and based on your requirements you

can select out that check boxes.

So let's wait for some time, it'll take a little bit time.

So we completed the Anaconda installation also just click on

this next,

next finish.

Just wait a

second problem.

Close this one. Now open error

search bar type and Aconda.

So click this navigator,

it'll open a little bit time.

They can process this one.

If you're seeing like this green image, then it's uh,

successfully installed

and uh, it'll be open in a few minutes.

So successfully completed.

Now, now you can

launch or you can install a Jupyter Notebook

and uh, all these things.

Okay, so this is about Anaconda.

You cannot try the Jupyter Notebook.

All these things from this Anaconda, Jupyter Lab,

all these things, uh, you can, uh,

see in the Anaconda platform.

Now you will see the how to give a extension of Python

to our VS code.

So if you don't have a VS code, just go

to Chrome five vs.

Code download.

So if you're seeing a this thing, right, download vs code.

If you click on this, it'll sell it

like you can select select Windows, Mac, or Amex.

After this, it'll be installed in your

local system successfully.

So I already installed that one. Let me show you.

Okay, so this is a web vs code.

Let me, let me go to first this extensions

pick one on the here you have to select it Python.

So I'm searching for the Python.

So now we don't, uh, we not give access

to our visuals to go with Python.

If you want to write a Python code, if you want

to execute a Python code

or Python files in this VAs code, you need

to add an extension first.

So this is the extension for the Python.

Just to click on this Python first one you will see here

Python, Python extension for Visual Studio code.

Just click on this install

click python debugger also automatically installed

successfully, uh, successfully extension completed.

Python extension.

How to check whether it is completed or not.

Miss, just go

to this one

You can select go to file

file here you can select Python.

If your installable Python enable local system,

then only you can get this option like this.

Okay? After the installation, also you have

to give extension of Python in code.

Then only you can get this Python file or Jupyter Notebook.

We already know Jupyter Notebook, right?

In previous Anaconda we already saw the Jupyter Notebook.

So if you want to uh, do anything in the Jupyter Notebook,

just simply you can click on this launch button.

It'll redirected to the Jupyter Notebook.

Otherwise go to the search bar.

You can type Jupyter Notebook

and you can open this Jupyter Notebook.

Okay, fine. So let's move

to the via code file.

New file. I'm selecting Python file.

Let me write a code.

Okay, so now, and run this one.

It'll ask to save.

Okay, so here you can see it's extension saving

with the dot p.

If you're seeing any extension with the file, having p,

that will be under the Python file.

So let me run this one. Yeah.

Hello. So we are getting the output

here in this term, right?

So this is all about the Python installation,

Anaconda installation in that Anaconda we saw this.

So many things like an conduct on the no two

Jupiter Labs Spider.

And if you want Anaconda prompt also you can search here

Anaconda prompt.

Okay, so you can uh, take this Anaconda prompt also,

you can get that Anaconda prompt.

Next we saw the V code

and uh, how to give extension

of the Python in this V code also.

So that's it guys for this session.

So we successfully completed the Python installation

and installation and vs code installation

and this setups also.

Okay, so in upcoming session we will discuss about the

data types in Python.

Thank you.

Module 4: Python Datatypes

Video: Data types: Int, Float, string ,Bool.

Discussing about like uh,

data types.

You can classify these data types into basic electric

categories for these simple types.

Second one is collection types.

The basic difference between these two, these variables

hold a single value.

So here it holds

multiple values.

This is the fundamental difference.

Examples for simple types,

float good.

Such in depends

in, yes.

R means string. The is here. Yes. R.

Okay. And the float is float.

That means to allow the dis marks.

So these are the base types,

but all these data types are holding single value.

Okay?

And before proceeding,

if you talk about the collection types

list, apple,

DICT stands for dictionary.

And one more set.

All these are collection map.

Simple example in Python monitor S

and a data type not

to declare the data type based on assigned values.

Automatically data types will be created.

These are the data types.

For example, I'm saying equal to a hundred here.

What is type of here?

Just it is connecting with the,

that is,

I'm saying 10.25.

You can guess what is the type of it?

It's a float. Let's check with all the things.

What is the value?

My question is, are we married?

Your answer can be or not? No.

Suppose if you are un married, the value is true.

If you are unmarried, false.

Suppose we are a single fox,

but the true false, the first letter is upper case.

Those are literals in the pipe. Understood.

Qualified. Are you qualified? That is my question.

Again, as certain you can say it as a, okay,

I'm keeping something with square.

Square brackets. Some values

I'm keeping here in the x, I'm keeping multiple values.

That is a list here for the list in python, square brackets

of the symbol,

what is it is here?

Symbol change S. Open brackets.

So whenever you see object like that, that is it. Okay.

The first one is product id. Product id.

P one means the descriptions.

TV P two, let's sell laptop

and P three

something mobile.

So what symbol is applied here?

That ly brackets that. Two. How many elements are here?

Three elements. Each element is a key and value.

Understand, if I pass product id, I will get value.

The value is t. Suppose if I pass P one, I will get tb.

If I pass P two, I will get laptop.

And one more type. I'm taking let's say,

So

Here, also same symbol,

but each element is a single entity, not

so here it is a list.

Multiple values, couple multiple values,

info, multiple values.

U is a set, multiple values, info, dictionary.

Again, multiple values. Check all these values.

Print a type of print.

V, b, the type of way I want to print

that value along with the data type.

After that, we said married,

type of merit

qualified.

Again, type of qualified

print.

X,

x comm, type of x,

print Y and type

of one after that.

Printing for that is a previous one.

Is that info

Type of info.

Come on. Check value.

And it's

what about 120.35.

Hello? High. Both are strings for the string.

For the first I said double pull.

For the second faster single pony code, you cannot apply.

Okay, what about false bullier?

Suppose for married we are send the value. False.

There is a bull. Bull means bullier,

which holds only two values, either true or false.

Second, there is also bullier.

Now the multiple values and of the symbol

and its data type square bracket is here.

List next to open brackets,

next to currently brackets with the key and value base.

DI CT stands for dictionary.

The last same bracket with single entities.

That is set. These are the.

Module 5: Operations and Expressions

Video: Arithmetic Operators

Hello, welcome back to the Python session.

In this session we are going to discuss about

automatic operators, different operator.

First we see different operators in Python.

So basically these operators are different types like uh,

automatic operators,

relational operators,

large scale operators,

and next category thematic assignment operators

and one more bitwise operators.

In this session, mainly we ate on automatic operators.

I'm ing notes for you letter and download this notes.

Let's come into the automatic operators.

So plus this is for addition.

So minus depends Here the symbol is hyen.

This is for subtraction.

There is a hashtag symbol. This is for multiplication

slash this is for division

And double slash, this is for floor division.

What exactly each one I'll explain

there is a percentage symbol.

This is a modular operator.

This is the remainder calculation

and still we have other operators

double star.

That is double symbol. This is to find power values.

Simply, you can say it as a power operator.

So these are the automatic operators using Python.

So for each one, example

plus for addition,

suppose your statement is your expression is 10 plus 20.

It returns. But

suppose you assigned value 10 to the A and 22 B.

If your statement is C equal to A plus B.

If you said print something like sum of

A and B

is C, what is the expected output

for this sum off?

10 and 20.

Is

that okay?

Similarly, theen symbol which is minus,

this is for software action,

10 minus 20.

We get minus 10, a hundred minus 70,

we get 30 programmatically if we want to say equal

to 70 something B equal to 20.

Now C equal to A minus B.

Now if you print it C,

what is 70 minus 20 here?

That 50, the value of C is 50.

Similarly, you want to multi, you want

to perform multiplication operation.

Then opera operators has take symbol

simply you can say it as a star.

The star mark.

This is for multiplication.

For example, 10 has 3 23.

It returns 200.

I'm taking your salary by example.

Your salary is one line.

I want to compute tax value.

Suppose 10% of tax. I want to apply salary.

Two point 10%.

Point 10 by a hundred means that point.

Now what is the tax value here as

for this 10,000?

Okay, similarly, house intelligence, HRA want to compute

salary into point to something 20%.

I'm applying so that the value

of HRA will be here 20,000.

Now I want to compute by using all expressions.

I want to compute net salary where it is salary

plus HRA minus tax.

What is the value here for the salary?

10,000 plus 2000 of HRA minus th 10,000

of 20,000 HRA 10,000 tax.

The total will be one lack

11,000, one lack 10,000.

Okay? This is the reason

that way you can use automatic operations

to perform different competitions.

Still more operators, we have step

by step wheel

slash This is where

division, some

of I said 10 slash five.

That will stand by five. What is the result? Two.

Okay, suppose five by two,

you get two five.

Similarly, double slashes.

This is called a flow division

10 double slash two by seven.

What is the meaning of it?

Divide 10 with the two by two.

Two fives tab. What is the remainder values? Zero.

Now this is low division value.

So what it exactly,

for example, if I say

15 by six, what is the result we get?

6, 2 12 Still please. There.

Point six.

In software using the slash, I'm going to use

double slash six.

What we get? 15 six.

Six two. Remainder value three.

This two is loaded simply against say quotient.

Okay, this

is got it.

So 15 slash two is two point something

where it returns only two.

That is a difference between slash

and w slash

in the next expressions spot we see how

to use them in into the programming

and one more operator is spending,

other two operators are spending.

We need to discuss person.

Then this is where, what is this? Operator's score.

This operator is score mod below operator.

This is to find remainder value.

This same previous example, I'm taking

16 slash three.

What are the result? Five something.

If I say 16 double slash three, it gives you

five 16.

That model operator. What are the remainder?

I divide 16 with

by

3, 3, 5, 15 for the remainder.

This is the remainder

what I got here, not this expression.

So this is the difference between slash double slash

and this model operator, the passengers.

The last operator here,

double star.

This is to find power values

for example two.

The of five I one. What is the meaning of two?

The power of five

two.

So how to get this two double star.

That's it.

Okay, so what is the value?

Two. Two. Two is four. Four. Eight. 2 2 8 18.

Two 16. 16 eight. Two.

That it

that it not 30.

That it? Q

eight again.

Two. Two for 16. 16 to do 30. Correct.

So similarly, one more thing you want to find out.

Square, by using the same technique,

for example square root five, how

to give the python expression for this

without using any predefined functions.

Of course we have predefined functions in pythons letter.

We discuss about those functions actually fight

to five square root of five minutes.

What is the meaning? It's a fight

to the in python.

How do to get the expression for this five double star

one by two

or you can say file double star in the place

of one by two.

I'm giving point. This is the meaning of square off.

So this is about simply

automatic operators used in the python.

In the next session, we are going

to discuss about a relational operators.

Thank you very much. See you in the next session.

Video: Relational Operators

Welcome back to the Python session.

In this session we are going

to discuss about relational operators.

So generally these relational operators are used

to compare two variables or two values.

These are used to

compare two variables

or values.

So for

comparison, we have different operators here.

These operators are called relational operators

and also we can call them as conditional operators.

And one more name, but these operators,

comparison operators,

comparison operators.

So below the list of different operators,

one is w equal to

not equals to

greater than.

Greater than or equals to less than

less than our equals.

So these are the six operators used

for the comparing two variables.

So the first one W equal to this is

for equality.

Check

if the, if the values are equal, it reads true,

otherwise false,

it rids billion value, which is true.

For example, I'm saying 10 equal to

20, the condition is false.

Here it for example,

10 W equals to 20, sorry, 20 is done.

Let's

next operator not equals.

For example, 10 not equals to 20.

This time what will happen? The 10 is the not equals to 20.

That's why it for example,

10 not equals to 10.

Now in this case it

and whether value are bigger than the current value

or not, you want to check

What they see.

How greater than,

so suppose age, I'm taking a variable.

Age is 25, condition is greater.

20. What is the value

of H 25?

25 greater than 20, but this is true.

It returns

similarly is greater than 30.

Amp.

A is greater than 30. It returns false

because value is 25.

25 greater than 30. It is false.

This is about not equals to any greater than

and one more thing

greater than or equals indicates

that greater or equals

greater than or equals.

Example, I'm taking your salary,

I'm assigning value one like to the cell.

The tradition is like this salary greater than

Oracles two 50,000.

True or false is one like actually one like is greater than

Oracles two 50, which is true.

One more example on the same salary

greater than Oracles two,

one like Actually here.

S is that is greater

than other equal to one likeness.

Greater or equals anything is okay.

But yeah, Sally is not greater than one

like but equals to one.

Like still it is

the loss condition.

Sally, greater than our equals two likes.

Now here the value one, like one leg greater than another

to two likes, condition, fault returns, false.

Excuse me. The next operator here,

less than,

let's talk about it every age.

Again, the age is 25.

My given conditions like this is less than 30.

Just substitute here. The 25 is less than 30.

The condition is, and one more

statement is less

than 25.

There is value 25, 25, less than 25.

That is false because both are equal.

Similarly, less than are equal to operators,

less than are equals.

Same age. I'm saying less than Oracle equals to 25.

Now in this case it,

okay, so the value can be less be equals.

Anything is okay. That is what less than are request.

So these are all the Alaska operators.

Generally user conditions,

whenever we want to check, sign any condition logical that,

that the relational operators are used.

Okay, so another name

for these relational operators in three different names you

can use the other name relational operators.

What is another name?

Comparison operators,

another name conditional operators.

So use it to check conditions.

But in all these examples, we have used only one condition.

Whenever we want to use more than one condition, we have

to go for large scale operators.

So basically large scale operators are used to

check multiple conditions.

In our next session completely, we discuss about

large scale operators.

Thank you very much. Let's meet in the next session.

Video: Logical Operators

Welcome to Python session.

In this session we are going

to discuss about logical operators.

Logical operators, basically three

and or not.

So what exactly These things we'll discuss now

and if all

conditions are true, final result is

this is correct

and final result is true

in case of our, if at least one condition is true,

that means any one condition,

if at least one condition is true.

So final result is

not simply isn't a negation of the,

that means not of true is false.

Similarly opposite, not a false is true.

Let's say

let's discuss more on this topic

for easy understanding.

I'm writing on end table.

Let's say condition two conditions I'm taking

for easy understanding

condition one,

condition two and final result.

So here I'm using symbols for true capital.

Ty for false capital, capital at array of five minutes

for example, false condition is true.

Second condition also true.

Then final result is look good to the second rule.

True and false. But what is the rule for that?

All conditions must be true,

but here only one condition is two.

Second one is false. So the result is here. False.

Similarly, false and false.

False and false. Definitely the false

look at tro all the given conditions are true.

Then only that big end but this is adopted.

I'm taking about or same kind of table.

I will consult for our operator.

So it is condition one,

condition two, and final result.

So let's look into this. True, true, true.

Guess how far true false one true enough total thing will

become to false.

True, false.

False. This is false.

Based on this you can understand one thing.

If all conditions are false then only final result is false.

This is the case for our operator. Okay,

next one is not, not a simply negation

the condition if any condition is true,

but that if you apply the negation like not the threatens

false simply negation.

For example, salary value 10,000

are one lack my condition.

Is salary greater than one lack?

Actually the value is one lack.

One lack greater than one lack, definitely that is false.

But the same thing, if you apply the negation not

of salary greater than one lack,

now it returns true.

Okay, so not of false is true not

of tool is false like that.

These are the three logical operator operators.

So generally these logical operators are used to apply

for more than one condition.

For example, condition one

and or all condition.

This is a fault after this one more hopper,

one more function I have to use means one more condition.

You want to apply again end or all then condition.

So here of the release example,

there are three conditions between each condition.

One operator again go

between condition two and condition three.

Another last operator. If I have young conditions,

we have to give young minus one

to last kilometers

either end or on.

So well all conditions must be true.

This is whatever expectation. Then go for end.

Anyone condition is true.

Can you go for or right

on That way you can decide these things

look into different examples based on this.

So suppose I want

only males data

then what is the condition for this?

Think that there is a color variable called gender.

The condition is gender equals to

let's say the gender values.

Gender values are like BF, like that gender equals

to um, that is a condition.

So here no need to use any logical operator

because single condition,

it's a single condition.

No need to use logical operators.

One more example, just look into this.

So from the given data, what I want is I want males from

from high travel

for example, this value variable is gender

and city value is location.

Now condition for the gender based gender male screens the

condition is gender equals to

double equals to because you are comparing.

Yeah. And one more condition for the location,

location equals pattern.

Now condition one condition I'm

keeping And operator here.

Okay, so this is condition one

and this is conditioned both should be true.

That's why. And operator for example,

the details are like this.

His name is Ravi. He is a male from Pune.

What happened First condition through

second condition,

location equal toba actually location value Pune Pune equal

toba is false.

True and false. What will happen? False.

So that this record will be that resected

like this.

One more test we do for the sale.

Previously we said gender is equal to male

and location equal to

the operator.

I'm giving you same previous condition I'm take

check for under person.

Suppose her name is Lata, female

from Hydrovac whatever

first condition, true second condition to

final, that means

accepted similarly.

Gary, one more record. I'm testing mail.

Let's say Gary J this letter,

this in this case both conditions.

First condition falls and secondary condition is also false.

Finally, what false So that it ize

in this whenever we are giving multiple conditions,

then only how to use last scale operator

of course in this example and is are right.

Similarly, one example I'm taking for our operator,

I was after this example.

I want, okay,

I want from,

I want all records.

I mean I want all the employees from

and Dell.

So remaining city people I don't want

now observe here how to give the condition,

tell me the condition for hydro.

The condition will be like this location equal to

hydro

Location equal to Pune.

That is second condition

location equal to de

this is now here it is condition one,

condition two condition.

Again more than one condition between them operator circuit,

but so here right operators, okay?

Because cities work for example, if you are from,

definitely are not from the remaining cities.

So I'm from Delhi, definitely not from first.

How is the chance here? Any one

condition can be true at a time.

All conditions at a time cannot be true over here.

That's why R is the right operator.

Look at this, Mr.

From whatever the condition, first condition,

true or second.

False or false.

One true enough, final result is true.

For example, she is from Delhi.

First one is false or second is false

or thought condition is true.

Final letters and one more Mr.

John

from chairman.

What happens here? First is false second, false

third also false.

So the final is that false

because no one, no condition is true here.

So this is the example for the our operator.

So I tell you one situation for not operator

for example if the requirement is like this,

I want all employees

except from

Aaba tell you

that they have employees from hundred different cities.

Accept these three remaining you want then

apply the negation condition.

Three ways you can do two ways.

Wave one already have a condition.

Suppose only hba want, only

people want already have planned a condition.

Location equals to aaba

or location equals to

Pune

or location equals to.

So now you'll get only the people from Pune and Delhi.

I'm going to apply. I don't want these people then they have

to say not okay, this is

so finally tell this

Ravi from Puni, what about this?

This condition false, this condition true,

this condition false between them

what is operator final result.

True. So here we are applying, we are applying not not

of truth is false from Pune.

Actually don't want Pune because except from Pune.

Except from Pune. Delay one remaining people.

So that we are in this case Ravi, that is not

of true is false, not a false is true.

Okay? So in another style we can apply this

it the not symbol

look into the second way for the same question.

So look into the question again, the task.

I want all employees except from Aaba daily. Now, right?

Independent condition, I don't want from Aaba.

What is the condition? Location not equals to

aaba condition.

I don't want two. Then condition, location

not equals to

location not equals to.

So here I don't want this,

I don't want this, I don't want this.

At time all the three conditions should be trooper.

That's why here operator?

Yeah, yeah,

this see wait, got

Uh, retest that from Pune.

What about the first condition?

Pune articles through une, articles to Pune false

location articles to delete through who And false

and true in case of and all things should be true, right?

But here this is false. So that final result is

false because of false.

This record is rejected. Got it.

For example, your name is running,

we are from China for this what happens,

this condition, true Shana article, true

articles to delete true, true and true and true.

Finally, final result is true.

So that ran from Chennai is accepted like

that you can apply the different things.

Now we have seen separate examples per end,

separate example per r and also separate examples per not.

I want to use the combination of end, end or operators.

Look into this

we can use the and or combinations.

The task here,

I want male employees

from Hyderabad, Delhi, Pune, Hyderabad,

Delhi, same thing.

Females from

Pune, Mumba

and all always which includes both males and female.

All from,

so this is my requirement, think

that there is a location column

or C uh, there is other columns like uh,

variables like a city and gender.

Now I want to apply this.

Let we go to the next screen for this.

Look back to the tasks.

Males from Del Postal condition only for the males

gender equal to male

and you want, you want males from haba and deities.

So that condition city equals to Haba.

That is one condition.

City equals two in Delhi.

This is second condition again here.

Condition one, look at only the city

condition one, condition two.

Your city cannot be two, your city can be either

or Delhi or not of anyone.

The chances, anyone. That's why here operate

risk or operate.

Now you see the center thing is condition one, the center

part is conditioned again between these two things.

Alaska operator here I'm keeping the operator and

because here gender should be male at the same time.

City can be either higher above or so

and or commerce.

This is condition number one. What about the females?

Look into the task. I want females from Mumbai.

Then condition should be gender is equal to female

and city

equals to une

or city equals to double equals to.

Whenever you are checking equality,

that should be double equal to operate.

Okay, double equal

number

Now this is condition one conditioned

and one more thing you need to concentrate for from China.

I want all people. That means from Sinai,

I want males, females, both.

That's why no need to give you want,

you are expecting all genders.

That's why no need to give any gender specific

condition directly.

The thought condition is city equals

in this case.

Condition one, condition two. This is condition three.

Again, multiple conditions between these conditions.

Again, laws per, for example,

you are a male from definitely will not be in into

next two categories.

For example, you are from haii,

but definitely you are not into foster two categories,

so your chances you can be at any one place.

That's why here one more operator or one more operator

or, so the center thing is condition one,

this is condition two.

This is condition between these three things.

Again, large scale. Now observe many conditions are normally

the two we have used to mix of and, and our

and our operators complex.

Okay? This is simply about logical operators.

In the next session, we learn about how

to use all these operators in the form

of expressions, practical use.

Okay, thank you very much. Let's meet in the next session.

Our topic is writing expressions with Python practically.

Okay, let's do the lab exercise. Lab exercise on the back.

Thank you very much.

Video: Arithmetic Expressions Lab

Sessions we have discussed about the different types

of operators, like automatic operators,

relational operators, and logical operators.

Today in this, the practical part of uh, how

to give the expressions we see, okay,

so we have different types of expressions.

So these expressions can be classified as

I read the same thing on the notes.

Then we see the practical part of it.

What type of expressions are automatic expressions?

The second category, relational expressions,

thought logical expressions,

the both category, membership expressions

and fifth category bitwise

identity Before that, identity expressions

and six five BITWISE expressions.

So let's see all these operations step by step.

Now I'm presenting this lab demo to the Google collab,

okay, so which is organized by the Google.

Google. So it provides a runtime computing mission

to run your code even

for machine learning, such kind of operations.

Also it provides G missions

and TP missions so that without having to any infrastructure

and without having to install anything, okay,

you can execute all the python port.

So already we discussed the different

automatic operators faster.

Let me list off those operators. One is plus minus.

The iPhone symbol is by the subtract.

Next asterisk symbol simply says star.

This is for multiplication slash for division.

What is this? Score module, low operator

and a double slash module L is for to find the remain.

A double slash is for floor.

The patient takes quotient.

When you divide some number with the other number,

the ENT part will take it.

And to find out the power value you have double stars.

So that is to find out the power values.

So now let's see the expressions here.

For example, I'm assigning some value to the variable

B, equal to something A three.

Now A plus B. Let's say C equal to A plus B.

I got here the expression is here, A plus B.

Now C, you're assigning the result

of the A plus B into the variable C.

Now I'm going to print something somehow.

Yeah, and B

is six.

Now what is the expected output here?

Value 10, B, value three

and C value 10 plus three depends 30.

We'll be getting that

just what it is.

Connecting with the runtime. Just after what is happening.

So now execution is completed. Just looking into this.

Some of 10 and three is starting similarly, let's play

with all of the remaining automatic operators.

I'm saying C, D equal A minus,

no a value 10, B, value three.

So that what will get the D, which is seven

independently, we'll check each automatic operator first.

Let's say equal to gain to three Y to B,

eight 10, B three, 10 to three.

It should get 30, 20 E, 10 to three 30.

This is for multiplication. Same thing.

I want remainder value when I divide

that return with the three.

Okay, I want to find out remainder value

before we are going for the remainder.

Let's say division value.

Let's say your fig, yay by B, that means a slash bill.

So it returns actually yeah, value is integer.

B value integer

and divided by three will get,

definitely will get a float value.

Just look into this. The value is float

3.333 something.

Okay. Similarly, I want to know like

what is the remainder value when I divide that?

Uh, 10 with three.

So a percentage,

that means this operator is called a modello operator, A

person symbol with the B.

So definitely the remainder B three three is

nine 10 minus nine one.

The remain is one here.

And let's check with the double slash.

What is that double slash shape? The is part floor division.

The is quotient valley.

So what exactly the ENT already I explain you.

So here for example, if I divide 10

with three, what is the question?

Three threes nine, I valley one

ion it's three.

There's three is floor division,

technically this is called floor division.

The operator to find out this one double slash

just look into the output.

10 double slash three. I got three here.

Got it the same time. I want to find out some power values.

Yeah, double stop.

For example, let's look into with directly with values.

So 10 is the value I want 10 square.

10 square means 10 double star.

So 10 double star. Two means 10 to the two. That is just 10.

Square means certain I got 10.

So even uh, like for example, if it is simple value

and titan, it can easily compute it.

Suppose there is a big value like a hold to the power

of something I want, or two double star eight.

That is do power of eight, which is 256.

So just let's look into very biggest value, like two

to the four of 32.

So in this way, similarly want to find out square

for example, to find out the square root.

Actually here we have something in import math.

Math sq opt like 25.

What is the meaning of it? Square root off 25.

So this is what your expertise square root off 25 is

how much that is Five.

But without using this predefined function, also

by using this double star operator,

that means power operator.

You can find out this square root.

So what is the meaning of uh uh, square root?

Old did you pour off one bite?

Suppose 25, the double star. I'm saying one by two.

What is expected result? Expected result is what?

Expected result is five, but you don't get five.

Just check what happens. 12.5, I got it.

So this is a mistake like uh, faster double star is computed

after that, that result is divided by the two.

To overcome this, what I have to say, keep this one

by two in the brackets.

Now one by 2, 2 1 by two is separat allocated.

Now 25 to the four. That one by two means 0.5.

That will be upright. Now square root, 5.0. Exactly. Got it.

Other easy way, you can say the same expression 25.

That will start off 0.5.

One by two means in the place of one by two.

Here I have apps 0.5.

So here I'm getting square. Similarly Y through.

Do you want to find out my like uh, fourth route

or QIC route or fifth route?

You want to find out, for example, you want to know

fourth route of hundred.

Okay, so simply say a hundred double star.

Fourth route means that is three four of one by four.

Now we get both in this way.

We have different operators in different states.

Let's say some real time example.

For example, your salary is one lack.

I want to find out tax.

Suppose your tax is 10% on the salary.

Our salary into 10%. 10% means standby by a hundred.

Instead of saying this has 10 by a hundred, you can say 10

by a hundred is value is 0.1.

You get 0.1.

Similarly, HRA HR is equal to salary into point.

That means that 20% you are sending us.

Similarly net salary won become

for the net salary inputs are salary tax, hr, I'm using

that word salary minus tax minus plus hr.

You'll get net salary.

Now let's print all the things.

Print salary,

the salary value hand printing here.

Similarly, the tax value I'm printing

HRA

net

chase net.

Now look into this one.

So here to find out the tax.

HR, a net for all these things,

automatic expressions are used.

Got it. Similarly, want to find out any number,

whether it is a even number or odd number.

We want to find out, for example,

let's go talk about hundred is 1 27

even number or number.

I want two find. Okay

then what is the even number?

If I divide, I divide any number with the two

and the remainder should be it zero.

That's why a person K mod operator, the person operator with

two equals two is zero.

Now if it is even number, it returns true.

If it is not even number, it returns false.

Just look into this, this conditions false means the 1 27 is

not even number.

Let's say a equal to 1 28,

which is even number now a personel.

Two double equals two zero.

Now it returns has 1 28 is

a even number.

So conditionally you can check.

Later we discuss about the if conditional statements.

Just simply I'm writing on condition.

If yay or two equal to zero,

then print.

I'm printing A is given number.

Yes, the condition falls miss, definitely

that is odd number.

Really? Yay is odd number.

Understand. So

here something comm, Mr.

Dr. Just check this execution.

So it's all the practical part

for the operators.

In the next session we see some expressions based

on operators.

Okay? That means we are going to work

with the relational expressions in the next session.

Okay, thank you very much. Let's meet in the next session.

Video: Relational Expression Lab

In the previous session we have discussed about

automatic expressions.

Okay? Now we see relational expressions.

Before we proceed into the practical part, I just try

to recall what all the operators available in the,

as part of the relational.

So there is another name for the relational operators,

comparison operators.

And one more time you can use conditional operators.

All these relational operators are used

to check the conditions, okay?

So like you are comparing always two

values to variable values.

That's why these operators also called us operators.

So for equal check here we have double equals to

the double equals to is for equal check.

You can say equals to greater than

and the symbol is for less than

and one more greater than or equals.

Two

can be greater or even it can be equal also.

Then it returns true

similarly, less than a equals.

And I suppose if I want to say not equals true, the negation

this exponential and equals.

This is for inequality Check.

Inequality check simply can say not

equals,

let's say the practical part of it.

So I'm coming to Google color to execute all these things.

Relational expressions,

the support, these examples,

I'm taking your age for example.

Your age is 25. The value 25 year

as assigning into variable page

and age greater than 18 M.

Look at this as value actually here, 25.

But you are comparing with 80 but it is greater than 18

Or not. We are checking.

So definitely the 25 is greater than 18.

That's why condition. Let me give one more condition which

can be the false for the same thing

is greater than I'm giving.

30 now is well 25

but I'm checking with whether it is greater than 30.

Now it returns false for example, to join my course.

The as limit is 80.

If it is greater than 18, then only he's eligible.

Otherwise he is not eligible.

Think that this is the credit here conditionally.

I can write a statement like if he is greater than 18

and print eligible

for example, condition falls.

Then I'm giving yes print

the student or the the proper candidate is not

letter about the yip statement statement.

What all the rules for everything like we'll discuss later

already in the coding rules we explain like how

to write these yip statements.

But more detail is specifically about if statement I create

a few more sessions per you.

Now check this actually as value is what? 25.

25 greater than 80.

Now P sales for example, I'm changing the value

one more candidate raise.

Now what about this ship? 15 greater than 18 condition.

False Because of false. What? It'll say what?

It'll print not eligible. Got it.

For example, if the value 18 look into this

18 greater than 18 again

for example minimum age is 18 at the time you can say

greater than r equals to 80.

So this is another upgrade of a greater than R equals

it can be greater equals little

bit in that is also that will be,

got it.

So this is greater than R equals for example, uh, the two,

the years limit, 40 years to become my student.

Think that. So then you can check the condition

if yes, greater than R equals two.

Now I'm checking with the max size limit

less than R equals two.

Let's say 50. Think that 50 is the maxis limit.

Okay? Now then print the candidate cells.

Otherwise not else.

Every is otherwise in python

and programmatically you have to

Say yes and then give your statement, right?

Not e. Suppose

what is value you have kept here previously?

19 Is there, let me keep some age

is uh, 55.

Think that this is age of one student

is applying for the course.

Now what did is say 55 55

less than article equal to 50 condition falls.

That's why it is saying not eligible. Same thing.

Test with 35.

35 is eligible. That's say exactly equal to 50.

So 50 also I'm keeping 51

not least understood.

That can be less or it can be equal to that 50.

And both the cases, the condition is true

then it returns true.

If it is true, it is printing. Got it.

This is less than less than our equals still not equals.

If you want to check, suppose equal check you want to,

you want to see for example, name, I'm taking my name bar

and his gender, but it's male.

Mm. If the person is male

or not, you want to check then the condition

for equality gender equals too.

Yeah. Now in this

case that is true.

Got it. That gender value is Yeah.

You are comparing with yeah for example gender value.

I'm keeping something like this gender value upper case.

Yeah. Now check the same thing gender equals

to Yeah.

Now here upper case letter is different

and a lawyer case letter step

that it is ing false.

Got it. But in both the cases I am expecting that is

to be true whether user has a type a small LA

or upper cm, anything to be true here then we have

to use a lawyer function, gender, talk to lawyer.

Even though upper case value is there

that is converted into lawyer, the coed value are checking,

comparing with smaller.

Now for small and or capital

in both the cases you'll get it wrong.

But suppose general values like this, yeah.

Now generally is equal to

you say definitely the condition is false.

Definitely we will get false.

Now based on this conditionally

We to operate this, then you need

to write the statement like this.

If gender lawyer

equals to yes, if yes print

that's applicant is female.

If condition falls, then

print applicant.

Yes, of course I have only two options that are male

or female conditions true.

I'm printing as female. Condition response are pretty

as male here.

Same thing. Inequality. Uh, inequality. Exactly.

Wanted for example gender

not equals to Yes,

actually currently the gender is what

female look at the value of the gender.

So current value is female, right?

I'm checking gender not equals to. Yeah.

Gender value of

and are not equals to comparing

with the you have not equals to F condition.

False actually f equal of threat.

So here you will get false here

for example, gender value is the,

I'm comparing gender not equals to.

Yeah. So what is the value of gender?

Yeah, you are not equal inequality check

you're doing with yf.

Yum not equals to yf condition is true like this.

Okay, so here we have seen the execution of equal to

for quality check not equals to

for inequality check rather than rather than

or equals to less than, less than are equals to

all examples she has.

So this is about the relational expressions, okay?

All these relational operators use in expressions

independently cannot do anything.

Just why using rather than what you can do.

I'm saying only greater than I'm running it.

It is giving you all these operators to be placed

in expressions only.

So how to give the expressions.

This is, but in all these expressions

you are checking, you are giving only single expressions

left side of the operator, one variable or value

or right side of the operator.

One variable or valid.

What if, if I have multiple conditions to be checked

at the time we had to use logical operators

in the logical expressions in our next session

we completely did the practical session, practical part

of the logical expressions.

Okay, thank you very much. Let's meet in the next session.

Video: Logical Expression Lab

Relational expressions.

In this session we are going with the logical expressions

probably we discussed what are the

different logical operators.

So I just recall those things.

The logical operators

and are not.

So it means all conditions must be true.

If all conditions are true, then on the final condition,

that means the final result is true.

So in case of far at least one condition,

at least one condition will be true.

Suppose I have high conditions, four conditions falls,

but one condition is true but still that result is true.

If all conditions falls, then on the final result is false.

So not means you know negation.

So in the previous sessions we discussed about reus,

now we see the practical part of it.

I'm starting notebook. Let's go with the fresh notebook.

Look at this. I'm taking your salary

for example, a salary 70,000

and name that salary.

So I'm just validating whether salary

greater than 50 and less than one mark.

One like one like

or mark salary greater that

50 K and less than

or equals to one black.

Let's say here also greater than oracles.

What this, how to write the condition.

So look at the condition,

the variable itself greater than Oracle two, two 50,000.

That is first condition

and salary less than Oracle equal two one like

so here, two conditions are,

so I'm expecting both conditions to be true.

That's why that means whether salary is between in between

50 and one like you are checking other here.

What is the salary value you have again? 70,000.

Now what about first condition?

First condition is true and what about second condition?

70,000 is less than four to one. That is also true.

So that what is the final result?

Okay, so this is a condition check for if salary is

between in between 50,009

one, not check this.

So here what is the final result? Check it.

So finally it is true. Same thing.

One more salary.

Not between 50 K

and a hundred K.

A hundred K means smaller this time

what I want less than 50,000

and greater than one lack I So

however, the condition salary less than 50,000

and salary greater than one lack here.

What is operator and or, or the right operator is

or now check it here.

Actually we a salary. 70,000. What about first condition?

First condition, false.

70,000 is not less than 50,000

or what about the second condition?

70,000 greater than one life. This is also false.

What is the final result? False. Got it.

For example, same thing. Do check for the 40,000.

Suppose your salary is 40,000.

What about first condition here? 40,000, less than 50.

That is true or salary Greater

than one lack.

This is false. The 40,000 greater than one lack is false.

Not true or false In case

of far one true enough final condition will be

that means 40,000 is eligible for this.

Got it. Similarly, one more value.

I'm taking one lack 10,000.

What about first condition?

One lack 10,000 less than 50,000. This is false.

Second condition one lack 10,000

is greater than one lack.

One lack. 10,000 is greater than one lack. That is true.

False are true. This is true

because one proof is okay.

Now let's validate and check it here.

My condition is this one.

What is the value? Salary value here? 70,000. Hmm.

Let me keep one second. Let's test all the phases.

In the first phase I'm taking salary as 30,000

condition two or false.

Your condition is true because

of the first one is true but second one is false.

One true enough. So final result is,

okay, let's check with another here.

I'm taking salary one lack 20,000.

So fast condition falls here. One lack 20 less than 50.

That is false but one lack 20 greater than one lack.

That is true. So false are true is true because of why.

Okay and check in between value, we'll take it.

Suppose your sally now is 90,000.

So first condition 90,000 less than 50, false

90,000 greater than one lack false.

Both are false. So

that if all conditions false final result in all false

and this got it.

So here a quick summary about two, about two operations.

So suppose if the task is like this.

If you want to check Sally between 50,000

and onelan, that means a hundred K.

Then the condition should be Sally greater than our equal

to 50,000 and salary

less than article equal two one.

Same thing. Not between salary, not between

50 K and on that K in this case in case of positive,

I got the end operator definitely,

definitely needs negative.

I will get all operator. This is positive statement,

this is negative statement in the positive statement end up

here means in its negative statement or

or will come into the picture so

that what is the condition here?

Salary less than 50,000

or salary greater than one black like this.

This is uh our pre previous example

what we have executed similarly.

One more task we just observed

to get more ly have it.

City, city column. Suppose your name again.

One second I'm taking your name.

Your name is Ravi

suppose city value.

Hyderabad. Okay, so the task is like this.

I want to check for whether the person

is from Hyderabad, Delhi Uni.

Okay these three cities are valid for me, okay?

I want to check whether he is from one city of Hyderabad,

Delhi Uni, okay,

so like the condition for listing

deliver.

Now what is the condition? If the column name is city,

the variable name is city.

City equals to that is one condition

city equals to this is under the condition

city equals to this is next condition.

Now I have more than one condition here.

Definitely lost operator required.

So between each condition you need one operator

Here I'm keeping or

or suppose if I give you end, first of all,

if I give you end, suppose if you are from Hydrovac,

definitely you are not from remaining cities,

definitely a condition will become false.

For example, another person is from Pune.

Definitely first two conditions will be false in case

of an all must be true, right?

That's why here, how is the chance anyone condition

to be true If you're from Hyderabad,

definitely you're not from Delhi and Pune.

If you are from Pune definitely are not

from Hyderabad and Delhi.

That's why what is a operator?

R is operator because your situation is any one condition

can be two here, not all at a end.

That's why R is right also.

Okay, just check it whether is from

the wanted list stop cities or not to check it.

So this time it returns true. Got it.

For example, let's take another profile here.

Let say Ronnie from Shana.

So here I got false

because post condition falls that condition falls

cloud condition is also false.

All our false means in case of our final result is once.

Got it. So here Ronnie running profile

is not eligible for this.

One more last profile I'm testing.

Mr. Wayne is from uni.

So Pune is uh, one of the one city are not here.

Whatever first condition, false second condition,

false third condition, true one condition

to total result will be

right in this multiple conditions.

So suppose if the task is like this,

what I want is

I want accept

aaba remaining city people are eligible,

accept and aaba.

Now that means here city should not be,

city should not be Pune.

What is the condition? Now city equals to city

not equals to negotiation equal to means double, equal to

not equals to that explanatory mark.

Mark symbol and then equal not equals to not equals

to one more condition.

City not equals to P.

So how many conditions are here? There are two conditions.

Two conditions means here fastest should not be so.

Uh, like conditions it should not be, it should not be Pune.

Both should not be understand.

So that that means both conditions to be true.

That's why other here

whenever I'm asking Pune daily, people

here are here definitely needs negative.

We end up here. There is a small, okay, just validate this.

For example here the city valley something Pune,

what about this post condition une bad condition.

True. What about second condition? Pune not equals to.

Pune is false. Pune equals to Pune is true

but Pune not equals to Pune is false.

Okay? One condition true here

but second condition is false.

That's why total result is false.

For example, I'm moving one more value here.

City equals to Delhi.

What about false condition here?

City Delhi nautical also had a bad roof

and Delhi nauticals to une.

That is also true. Both are true so that he is a true.

So here final observation here if you want positive here,

if r appeared, you know in positive case R appeared

definitely need negative case and will be upgrade.

Okay? Now let's take this example

our condition une.

Why is false? Totally false because false condition true

but second condition is false.

One false enough total thing will become false yet because

and is expecting all conditions to be true.

Okay? I'm replacing this

with the Delhi first condition.

True. Second also true.

That's why for Delhi the given condition is

true like this.

But in all these examples may maybe in the salary examples

or maybe in the city examples there are or

and anyone operator may have used, I want

to use the combination of both R

and n For example, if the task is like this,

if my task is what I want is

iaba daily people and their salary should be

greater than or equals

to one.

Like this is what expectation.

So person taba Delhi

at the same time the salary should be greater than Oracle

equals two one.

Like now tell me the condition for the city only for city.

City should be hba Delhi.

Anyone is okay what the condition for hba

city equals two aaba.

I'm typing the city name of short names for the city

and city equals to Delhi.

So between them Sal operator are so the center thing treated

as one condition and one more condition is required.

That is for the salary. What

is the condition for salary here?

The salary greater than Oracle equal to one lack.

So this entire left part condition one,

this is second condition.

Now between them again operator city to be true salary

to be true here strictly that's why.

And observe in city within the city

or operator between city and salary

and operator for example, your name ra,

the name of the profile is Ra is Sally is Lac, 20,000

and he's from

now whatever.

Just validate this first.

Internally validate this first condition,

two second condition.

False here,

greater the article two one like

then to our false, what will happen to our false

true and

what about S condition?

True again. True and true. Final result. True.

Understood. That's why Mr.

Zel will kind it for this criteria. Got it.

Let's check it past

current, but I'm changing this

check in the next example.

That's a Chris in the

name solid and thousand.

What about city? City condition is true

but s conditions now.

So Mr. Chris is not eligible for this

like Sally.

So city condition is true finally because he is from,

but Sally is not met one false enough in the case of end,

total thing will become false.

That's why he's not Kris' not eligible.

Let's check with one more thing Sally.

One like 20 but city is Chennai.

Now what happened here? City condition got fails

because he's not from Hyderabad at the same time he is not

from de, he is from Chennai.

That city condition got fail but Sally is still there.

Is result is false because city criteria met Sally.

Sorry Sally criteria met city criteria not met.

So Karen is not eligible here. And one more drive.

I say Karen, my

from that

to Sally is 20,000.

Hmm for the current I here please check.

What about city criteria, false salary criteria? False.

Both are false. That's why Karen,

my is not also is also not

eligible in this way.

You can use combinations of your end or it.

Got it. And one more thing.

What are the thing that we have for this criteria?

I'm applying negation. Just check what happens

and keeping in one bracket I want to apply its negation.

For the negation, the not symbol that I have used.

What is not? Not of false.

Okay, but finally here it is not accepting like this.

So apply with the knot function.

Is there any solution with this? Not off.

And what knot is the, so you can use this symbol

for the knot when

whenever you are checking inequality, knot equals to like

that not for entire condition.

Got it. Actually here, what is the criteria?

Accept to deli that is a meaning.

At the same time, Sally should not be greater than Oracle.

Equal two. One lack.

Actually this is positive case per the positive.

If you apply the negation, which is not, finally it turns

to false.

Not of true is false. Not a false is.

Okay? So this is about logical expressions

in our real programming part we see more complications

with the deep statements and statements like that.

Okay, so in the next session we are going to work

with the topic called programming.

Using these expressions up to now like uh, okay,

before going for that one.

Also like we see some other type of, uh,

expressions after that.

Finally, we include all these things into programming.

Now independently after you have checked up

to now independently have checked up

to now automatic operations, relational operations,

relational expressions, and logical expressions.

Okay? In the next class there are special operators

which are called as a membership operators, like in

or not in such kind of operators.

So in the next session we'll be discussing about that. Okay?

Thank you very much. Let's meet in the next session.

Video: Membership Expressions Lab

Operators and collects expressions

with some practical stuff.

Now let's enter into the topic of membership operators.

There are two membership operators here in

the number one yin and not in

in operator.

Second one is not in operator.

These two are not statements.

Simply can call them as operators

first.

Uh, let me explain some example letter functionality.

I will explain. I'm taking a string. Let's say hello.

I'm checking the word Y,

the character Y, yes or no.

This is a condition E in.

Yes, the letter E available

or not available if available.

It returns. And one more.

I'm keeping E, L, L in.

Yes. What happens?

L-E-L-L-L Available. Yes or not? Yes.

So it returns and one more condition I'm

giving you that says Z zin.

Yes. In the hello, the as well is hello.

Within the hello there is no character concept.

Now it returns box

On that way in operator you can apply on the strings.

Same thing. It's opposite.

For example, that's a fruit

equal apple.

I'm checking. P not

in fruit.

P not in, but the word P,

the character P available in apple.

Okay, but you said not in red, not in.

So it returns false.

Okay, one more example.

Z not in fruits

here you said he applied some

actually Z is not available on the apple.

So basically that is a false

but not in he applied not a falses.

You get true. So based on this

you can guess what is the functionality.

You can guess what is the functionality of the scene

and not in operator.

That will keep something into the notes.

Let me explain the functionality

in if given item

available in the collection

that says string is also acting as a collection.

As a collection of characters like given item

available in the collection.

Then returns

if not available returns.

False.

So this is functionality of again,

and if you come to the mountain,

it given item not available, it true

opposite if available.

Its false.

Okay? This is the functionality opinion not in operators.

Let's see the practical part of it.

I'm opening Google collab notebook,

come to the first notebook.

Let's take our beginning example, which is, hello

I'm checking for HRE in S.

The letter E available in the hello.

That's why it has to written.

That's why it is uh, connected. The run tap.

Now it disconnected. Same thing.

Bunch of characters you can give suppose LL

in that means in, yes.

So E, L, L, L.

Instead of characters available in

that given thing.

That's why it returns true. Let's say

Z, the character in yes,

but in the hello, there is no Z character.

That's why it returns false. Got it.

Same thing I'm applying with negation Z not in.

Yes, that basic J is not available in the yes.

Then it returns

true by example.

The letter ye in fact available in, yes

we are saying ye not in.

Yes, if you're available it returns false power

not actually E available.

That's why it returns false. So this is the functional.

After this there is a line

of comment is on Python.

Okay? Now the word class available in the line or not.

I want to check. I want

to check whether the class word available in the given line

class in

or line available.

That's why it returns true. Okay? Same thing.

Whether this line is related with python,

Python line or not.

That means Python related comment or not.

You want to check then check like this python in line.

Still it true, it doesn't matter

what is the position of that line.

Maybe Python can be at lost or in the middle

or in the beginning it doesn't matter.

It is checking word by word

if available returns true otherwise false.

Okay, same thing. I'm checking Java.

Java in line.

Java actually not available. That's why false.

Let's go with its negation.

The same Java not in line.

Actually Java not available. Then it returns true

because of noting

similarly, let's say Python.

Python noting line,

actually Python available in case

of noting if available it returns false.

Okay? This is in and knotted.

So this in can be applied not only on the strings,

you can apply on different types of collections.

What are the basic collections in Python list type

dictionary set.

Let's try all these things.

In all these both types of collections.

I'm taking one list subject already in data type spot we

have discussed about list double

dictionary, set all these things.

Suppose I'm giving value something like this.

10, 30, 50, 80 and 90.

Now I want to check whether the value 50 available

in the given list or not.

15, yes, 50 available.

That's why it has to return true check hundred

not in X, actually hundred not available in the list.

If not available then only returns true.

For example, I'm checking hundred yin X

in fact hundred not available.

So that falls, that is a difference

between in end not in in case of in if available,

true in case of not in if not available then true.

Okay. For example,

I want to check for the multiple values.

30, 50, 30, 50 is the list

in I'm saying x check.

What happens? Actually 30 50 is available,

but still it is showing error.

The principal because it is taking each element

is the first element is a list of 30 50,

not is the second element is a list,

not is the third element is a 30 50 list.

Not fourth knot, fifth knot.

That's why that has return. False.

Same thing apply with negation.

x clear supply.

Not now it returns

because not of true is false, not

of false is true previously got false.

That's why when you say nothing,

not of false will be applied.

That's why I got it. So in this way, listed, uh, that uh,

in not in these operators are called membership operators.

This can be applied on list.

Collectionals, same thing you can apply on Apple.

Let's say I'm taking a top, top off

the employee id 1 0, 1 name Ravi

is salary 90,000 and mail

and designation software engineer

and he is from the department.

Let's I depart. So think that here,

there is a structure here for the info.

What is the structure? Id name salary next bill.

The fourth one is gender. The next field is designation.

Next field is or department.

So department is it.

Now you want to check whether it department,

whether the word maybe in the first, it doesn't matter

what is the position of it, whether the word it available in

this info not you want to check it in info

check this output, it is available in the last position,

which is as a department.

So that's why this is true. Got it. Same thing.

The word marketing available in this or not.

We just confirmed it. I want to check for the department.

Marketing, marketing in

info that is not available.

So that false for the same thing I'm going to apply.

It's negation marketing Martin,

Martin info marketing.

Actually not available. If not available

then only it returns, right?

Got it. Similar style these membership operators seen

and not in can be applied on dictionary apps.

What is a dictionary? Dictionary is a collection

of key and value page.

I'm taking a sample dictionary. Let's say the name D info.

I'm taking your ID as key.

So is a key value pay always plus one is key.

Second one is value. Already, you know the key rule,

key rules key should be unique.

Value can be duplicated.

For example, 1 0 1 is uh,

details like 90,000.

Think that the employee ID

and salary our bank account number balance

and for the 1 0 2 90,000

because value can be duplicated

and 1 0 3 40,000 1 0 9 between, uh,

after 1 0 3 and uh, before 1 0 9.

There are no account numbers here.

The next account hold is 1 0 9 is balance amount 80.

Next one 20. Balance amount one.

Think that in this way there are some five

ways just come from this.

Come from this. Learn off the info.

There are five element I want to check.

Suppose treat that key. The first one is key.

That is account number, bank account number.

Second one is balance amount.

I want to check whether this is available,

whether a particular record number is available

in the dictionary or not.

I'm checking with 1 0 1 1 0 1 in DO.

Check this now 1 0 1 available.

That by for example,

one level triple uh, triple one.

In fact 111 is not available

in a given dictionary.

Now you'll get in case of in operator, you'll get false.

Got it. Let's supply its negation.

The P is taking only key part.

It is not checking the value part.

Triple one n

dean let's supply this time.

Not in opera, not in actually triple one not available.

So that if not available then only that is not in opera.

Once again, look at the din info.

There are five page, the account numbers are key

and balance amount as well.

But I want to check in the values.

But if you apply in operator

or not in operator on a dictionary, it is checking only

for the, it is checking only for the dictionary.

So that means key spot only.

It is checking, but I want to check within the values

then the technique, we need to separate it.

Let's look into this. The info do values only values

spot will be separated.

Once values got separated within the

on the value spot you can apply in, not in operators.

Now let's do, I'm going to check for 90,000.

The value 90,000 in DFO

dot values directly.

I dfo it is second only.

Keys only in the keys.

90,000 keys not available,

but I want to check within the values dot values.

Now it returns true because 90,000

value appeared in two types.

It doesn't matter how many times it appeared.

Least one time it's available, it returns similarly.

I want to check with the tool la I want to check

with the tool la ax in

din info dot values.

Now ax not available returns, false

apply the negation, same two LA not

the input values.

So actually two lax is not available if not available,

then only not in returns.

True. I got so up to now.

We applied this up

to now we have applied this in operator

or not in operator on strings, on list on top and on al.

Now one more collection spending for us. That is set. Okay?

Set is a collection of unique items. So you can apply this.

Uh, membership operators.

These membership operators are set

also, let me take a sample.

Your sql, I'm keeping it as a set.

Hundred, 200, 300. Let me take one more. 300, 300.

It doesn't matter how many three hundreds you are giving,

how many times the values is repeated.

Set is not allowing duplicated duplicate values.

It takes only unique values. Let's give 400 and 500.

So actually here I have five to five

and, uh, three totally eight values.

Okay? But in fact in the yes, how many values will be there?

Only three values, five values

because 304 times we have one,

but it has taken only one time.

Now, this, uh, this a what is the type of object?

It's a set, no type of s just say and confirm it.

This is a set. Now I want to, within this set,

whether a particular item available or not.

I want to check. Suppose value four 50 in Yes, I'm set

where CA set, but four 50 not available.

So that falls, lets check for the 400,

400 in years, but 400 is available so that it returns

one more four 50,

which value?

Four 50. Four 50.

I'm applying not, not in. Yes.

In fact, four 50 not available. So that it has to return.

If not available, then only it is written. Got it.

So this is the thing about membership operator

without membership operators also,

you can play all these things, but you have to use

multiple conditions between each condition for the, in

our operator have to use for the replacement of not in

the end operator have to use.

Okay? Instead of doing like a inin that the end or,

or such kind of things directly can apply

in, not in operators.

Got it. So that the exhibition will be faster.

At the same time, sometimes it will be confused whether in

this situation where are to be used

or had to be used in the beginning at the time

of a beginner level programmer.

Okay? So simply you can avoid the

confusions by using this one.

Okay? So it's a discussion for the, for this session.

In the next session, we're going to discuss on

something objects called

identity objects.

Okay? Up to now we have discussed in all our previous

sessions about the expressions, automatic expressions.

We have seen relational expressions,

and in the last class we have discussed about logical

expressions in this session, membership operators

and expressions, both theory and practical warehouse.

Okay? In the next class we work

with identity objects and identity.

I, I like checking the identity expressions.

Okay, let's meet in next session. Thank you very much.

Video: Identity Expression Lab

In the last class we have seen about membership operators,

like now one more interesting operator

that is identity operator

to in this session we'll discuss about it.

So mainly here there are two operators.

Number one is the

negation for it not is

so where exactly this thing is useless.

If memory location of two variables is say

if memory location of two variables

is equal, then it returns true.

If not

returns false.

Whereas same thing like opposite the negation action

if memory not equals, if memory location not equals,

then it returns true

not equals then returns.

So if equals

it returns false.

Okay, that is similar to in

and noting here is not is so here what exactly the concept

of memory location, we try to understand it before that.

I copy this notes into the your document.

A small example I'm creating only stop it.

Let's say X is a list

keeping some values from 2034.

If I say Y equal to X.

Now let's print both the things print X

and print what for the both we get

the same output, which is done

20, 30, 40 here.

When you say Y equal to X, what happens internally is

that means in ram,

in the ram in some location one object is created.

The list object. The list object is here.

10, 20, 34.

For this, the variable name assign that is X.

When you say Y equal to x for the same memory location

one more alliya will be set.

Now in fact there are two variables,

but memory location is set.

Okay?

There are two variables

and that memory location is same here.

Okay? Now in this moment,

if you check a condition like this,

Yex is one.

Both memory locations same or not? Yes.

That's why it has to red done same thing.

Opposite action. XE, not

or not is what it returns opposite

false, okay,

so how to make sure that the memory,

memory location is equal or same or not.

Let's say the practical part of it.

Lemme open that Google collab for this execution.

Okay, let's create one list first, let me run the runtime.

Now it is connecting the run python runtime in the cloud.

I'm creating one list, same previous value I'm taking

10, 20, 30.

But after that, why the exam

before this some operation will play later.

We think about this allocation.

Just try to compare both the values.

Now print x print value,

same values in both.

That's why I'm changing some value in the X.

For example, the fastest cell, which is index number zero

X of zero, previous value was 10.

First value was done. I'm going to make it as hundred.

Now print both

X and Y.

Now here actually I did a change in X,

but this change reflected into Y.

In fact where I did change,

I did change at XI did not do anything for the Y.

But in Y also, same change happen.

Let's do some change in the Y for example, last element,

I want to make it as 400 Y off minus one

way off minus one equals two.

That's of 400. Now print both values.

Print x, print. What?

Now check the values. I made a change at Y,

but in fact the change reflected into x excels

why this is happening because the memory location of X

and Y in both are same.

How to make it true simply say X is Y.

That is true.

That means both memory, location is same.

That's why if you do something change in the X,

the same change will happen in one.

Okay, it seems like that,

but in fact that memory location is same.

You are indirectly you are updating that memory location,

values, understand,

look into that idea of x.

Run it. What is id Id for X?

This is a memory location id. Four x is this one ID four Y.

What is that?

Understood, same Id generated for both, right?

Got it. For example, X equal

to one more example.

1, 2, 3, 4, 1, 1, 2, 4, 5

Y equal to XIZ

and look into the jet Z is equal to

the comprehensive style.

I'm applying expression follow loop, left side expression,

right set follow loop in the place of where I'm being

for V in X.

How many values are there in X? There are four values.

So that that means loop transfer loopit rates

for four values at each iteration

the value is coming into V.

I'm going to take that V.

I'm not doing any modification for that value.

Now gender is also created.

Just check what are the types of each inhibitor.

Print X type of X, sorry,

and a print Y type of Y,

print Z, the type of check.

Now everything is a list come from this

by checking the

output server list.

But I'm saying that XC is Y, the condition is true

because memory, location of X

and Y not check

the Xs Z.

False because memory location is different.

That means if I do some changes in the X, the changes won't,

won't be reflected into J.

Same thing. If I operate on jet,

the changes won't be reflected in X,

but if I do some change in X,

the change will be reflected in Y.

If I do some change in Y, same thing.

Reflection will happen at X. Got it.

Uh, now opposite X is not one

actually XY memory location sale in the case of is not

if location sale, it returns false.

For example, X is not on set. Now it returns.

Got it. Now I'm doing some change in X. Let's say X of zero.

First element time. I'm dating previous value one.

Now I'm making it as hundred. Confirm this.

Print X, print y, print it.

What is your expectation in X and Y?

Same values, but inject the bold values

because it did not do any change in inject.

Now from this output, XY has same values

that is having different values

because the changes are not reflected into jet

because memory location is different.

Similar style. I do some objection in jet

it says J off minus one.

I'm going to modify the last value in the jet 500.

Now print all the things, print X,

print Y, and print.

I said check this.

Now actually I did change it J,

but we changes are not reflected in XY

because their memory locations are different.

Let's compare their memory locations, how

to get its memory location id idea of X

idea of Y

and ID of side for X.

Y. Same values you'll see,

but that is will see a different value.

Confirm this ideal ID. Same for X and Y, right?

Okay. And for ax also same id

but for J different, right?

Remember here E is different, double equal is different.

I'm saying that X equals two y, what happens?

Two X is y true,

but there is a difference between W equal two and X is Y.

So what is W equal two is

checking check C.

Quality

that means contains same values or not,

but it'll check it.

But in case of it is checking what even those same values,

even those same values, the condition may be true or false.

It is taking what? Memory, location checks.

Memory location is same or

that is a difference between list.

Got it. Same x not equals to YX is not Y on the,

this is about identity operators and their expressions.

Okay, so in the next session we are going

to discuss about Bitwise operators.

Thank you very much. Let's meet in the next session.

Video: Bitwise Operators and Expressions

Discussed about identity operators and their expressions.

Okay, previous to previous class,

we discussed about membership operators, like in now ease,

not easy, the identity operators.

So in Python, one more special operators,

that is bwe operators.

In this session we discussed about that

Bwe operators.

So especially into electronic devices,

especially in the digital electronic space,

these bit device operators are most useful.

So generally in regular application programming you don't

use, but whenever you are trying

to embed your code into the chips, okay, so at that moment,

this bit five to increase the speediness within the chip

level, okay?

These bit fives operators will be helpful for,

of course there are many, but for a four, just

for understanding purpose here I listed about two,

like there are two symbols here, two operators here.

But is ampers percent singular amp percent.

This screen is end, that means bit twice. And operator,

similarly, there is a pipe, this is R

bit nor normal.

R bitwise operator.

This two operators how behave

and how are behaves in just second.

So when now we cannot A give a by

or any is any number into the by bits.

So that will to the boundary format.

Uh, for example, I want to convert a five.

What is a way just observe.

I want to convert this into a boundary array,

which is a four size.

Okay? Two twos. What is the remainder?

Two, twos four. Remain value. One, two ones.

Remainder value zero two zeros.

What is the remainder value one.

Now what is the binary format of this?

Five zero coming disorder. Coming disorder.

Okay. Zero one. Zero one.

Okay, this is a binary format of similarly.

Let's talk about three.

I want to convert this three into

binary two. One

Two, remain value.

One, two zeros. One.

What is the format? So come in this sort

0 1 1 of course here, uh,

I'm comparing with four size.

That's why I keep one more zero extra

because in the left side, zero there is no value.

Got it. So now remember this

double zero one, double zero, double one.

This is binary format of three

and 0, 1, 0 1.

This is binary format of five. Now let's play this bit.

Device operators on this one. What happens? Just check it.

Sorry,

I will keep it to the dock part.

For example, our expression is uh, three end of five.

How it works. You'll see my statement is

three and five.

This is expression. Hmm.

What is the value of three bin format of three?

Zero. Zero and one. One. What is the value of five?

The binary format of five, what we got previously.

0, 1, 0 1.

Other here if all our ones,

the end behavior is like this,

it compares each bit,

each corresponding bit.

If both are one

returns, one

other here, zero, zero.

Both are not one. So that 0 0 1, both are not ones.

0, 1, 0, 0.

Now here in the last, both are one

that means one.

Now the result is in binary

that into like a byte.

That means number numerical format. If you cannot, this

Other here, this is two deliver of zero.

Two, deliver of one because all values are zero. One, two.

This two of two, that means two square. This is two q.

Now format is like this.

Multiply this zero and a two Q zero into two square,

zero into two of one, one in two of zero.

What is the result of zero into two Q? Definitely zero.

Anything is zero. Zero into

two square, zero into three.

Four of one plus one in 2, 3, 4 of zero.

What do you get? Zero plus.

Zero plus zero plus one and two.

One. One and one is one. All left side are zero.

Final result is one.

That means when you say three

and five, your reply is your result is one.

Okay? So similarly for our operator, what happens

there is a symbol called pipe symbol.

Let's try to understand what what will happen.

In the case of our for same three

and five values, we'll see

three pipe symbol, a pipe symbol four or our operator

and five amp say what is the result of the three when format

for the 3 0 0 1 1.

Is it true? Confirm this.

0, 0 1 1 we have taken

for

3, 5, 0, 1, 0, 1, 0, 1, 0 1.

This is four, five. What are operator doing?

Pipes.

If anyone is one.

So it is also like comparing each bit,

compare each bit.

If anyone is one.

One,

okay, if both are zero, zero.

If both are zero, zero

based on this T tell me what I can see here.

Zero. Zero. What is expected?

If both are zero, but it has to return zero.

If anyone is one again. One zero.

Anyone is one. This is one.

At least one should be there one.

Now both are ones understood.

So in the boundary format, the result is zero,

triple 1, 1, 1, 1.

Now this is a binary format

that means bit format cannot into bytes.

That means numeric format. Hmm? What is your style?

Start from two deeper of zero. This is of one here.

Two. That means two square here. It doesn't matter.

Anything into zero is zero. Okay, lemme write it.

Two Q, command summit zero into,

zero into one in two

and one in two and one in.

Come on. Do the same thing for each. One.

0 2 2 Q plus

one in two, two square plus

one in two of one plus

one in two of zero.

Next zero to zero anything zero one in two, two square

four, one in two.

Two of one of zero is one,

one into one, one.

So final result is what? Seven.

Got it now three.

The five five, the result is seven.

Same thing when you applied three

and which was that with five?

I got one whether our calculation correct

or not, we'll test it

practically in this way.

The bitwise operators will be working

across notebook for this,

but we cannot with the run time disconnecting now

I'm directly applying 3, 2, 3 and five.

We had previously what is our expectation?

Three and five. How much

One is the result?

Now confirm. Well

that means our calculation is correct and the same thing.

Let's apply for the R

3 5 5.

What is our expected value? Three or five. How much?

Seven do it. That is seven.

That is our previous explanation of the,

the computational explanation is correct for the

bitwise operators.

So finally get clarity.

What is the functionality of end bitwise and

and uh, functionality of bitwise are in the case of end,

it compares each value.

Both are once then only rating one. That's why zero.

Zero, zero. So compare this zero.

Zero is 0, 0 1 is 0, 0 1 is zero zero.

Everything is zero because it is

expecting all should be ones.

But in the last of the right side, one,

one, then only it is written.

Understand. And next,

bitwise our operator here.

The functionality of this bitwise operators

0, 0, 0, 0.

One, one. 1, 0 1. That is what? One is.

One is okay. If any at least one. One is there. It one.

Understand. So zero. Zero no ones.

That's why 0 0 1, at least one. One is there? 1 0 1.

One one. Both are ones. At least one. It is one. One.

It is expected. Understand.

Okay, so this is about the between operators.

In the next session we are going

to discuss about try

to use all these expressions

programmatically in one program.

Okay? Thank you. Let's meet in the next session.

Video: Arithmetic Assignment Operators

Hello, welcome to Python Sessions.

In the last session we discussed

and uh, did some practical work with BITWISE operators.

So today we are going to discuss about automatic

assignment operators.

Let's start it directly, practically

our initial examples.

I will insert, demonstrate with Google color.

Max will use ID like vs.

Code pie chart, such kind of ideas.

So I'm connecting with the runtime.

For example, there is a variable, yay.

The value is now for the

variable, I want to add 10.

And traditionally you can say equal to a plus 10.

Now already here the value a hundred of

that, you're adding 10th.

Now the latest value of a

one, so the same thing.

This is here equal to a assignment operator,

automatic assignment operator.

I want to, I want to show you,

let's say yay latest value.

Now one 10 yay plus equals to to the existing value.

I want to add 150. So what is our previous value of it?

Current value one 10 for the 1 10 50 is added.

So instead of the meaning

of the center expression is a equal to a plus 50.

So here a value one 10 and plus 51 60. Now one 60 stored.

Yeah, the equivalent expression is simply in short,

a plus equals to 50.

Now, latest value of a one 10 plus 50, which is one six.

You can apply the same kind of model, same kind

of assignment for decrements,

like a subtraction multiplication for everything.

Let's say yay minus equals to I'm keeping 40.

So the meaning of the statement year to,

yeah, minus 40, what the value now?

1 60, 1 60 minus 40, 1 20.

That will be, that is the latest value of that is one 20.

Same multiplication. Let's a C equal to five.

I want to make this valid.

Double C, that asterisk symbol, which is

for the multiplication equals to

what is the meaning of the statement.

C, equal to C and two, two.

Now C, value five, five into two.

Now latest value, the C is 10.

Now C latest is 10.

I'm trying, I'm applying same kind of operation

with the divided by the slash two.

The meaning of this statement C equal two C.

So C by two, which is the latest value of C is five.

So same thing with the, some realtime data will, like

for example, the person name is,

is 50,000 or one like I'm taking

now I want to give

20% increment.

What is the expression of salary plus equals to

Salary into point 20% means

20 by a hundred, 20 by a hundred means point.

So previous value one,

like one like into 0.2 20,000, that 20,000 is added

to salary.

Now what is the meaning of full meaning

of this entire expression?

Salary equal to salary plus

salary into point

instead of equal to, again, same variable plus.

And then this expression directly can say salary

plus equals to salary.

Two point. Now check the latest salary

of previously one lack.

Now it should be one lack 20,000.

Got it. Similarly, I want to apply the tax.

So this tax that is the independent tax expression we are

giving because tax variable previously not available,

it's about 10% tax.

I want to apply. I can say Sally into point.

Now tax is 10%, one lack, 10% means 10,000.

This is not automatic assignment operator.

This is simply assignment operator understand.

For example, he see previous one lack 20,000.

Now I want to give decrement of 30%.

I want to give decrement of 30%, then

give decrement

or 30% to salary.

No salary because of decrement.

I'm saying minus equal, equal salary into 0.3.

The full meaning of the center expression salary, equal

salary minus

salary into 0.3.

So what are the value? One like 20,000 on the one like

20,000, the 30% is calculated.

So that amount will be directed from this one.

Now the latest salary here

after decrement,

which is 84,000.

Got it. That means 36,000 is uh, up,

not directed from here.

Got it. So in this way you can use

automatic operator and incremental

as automatic operator.

Decremental automatic operator.

If it is incremental plus equals to your saying incremental

minus equals to.

Okay. So in the next examples we'll discuss about

automatic operators are operators on

all types of data types.

Maybe on the independent data types in the,

like a single value data types

and uh, collection data types,

everything on automatic operators we'll apply

on all data types.

We see that about, we see about

that part in the next session.

Thank you very much, but.

Video: Arithmetic Assignment Datatypes

Hello, welcome to Python sessions.

So in the last session, what we discussed

in the last session, we discussed about automatic assignment

operators, right?

So in this session we are going to do same kind of automatic

operators on different data types,

how it works, just check it.

I'm starting a new notebook.

So connecting with python runda, I'm going to take a,

let's say equal hundred, our data type of cept

other here, VA plus

inte then is also anti in plus inte.

You get incept and one more thing

ya inte, but 10.25 is a float.

Now integer plus float always it returns float.

For example, basic equal

30.5, which is a float web

for the B I'm adding one integer,

another value will be floats.

Okay? All these are the ab,

what are the variables have taken?

And these variables are holding simulated.

That means single value. Now you can apply these operators

on string cells, but all operators are not allowed.

Depends on the data. Different operators are allowed.

For example, I'm taking string,

let's say a SQL to follow

data hypothesis string.

So for this, yes I'm adding a number

that will be invalid expression

because string plus in is invalid.

Same thing, yes.

Plus, let's say why now string buys a string.

Now here what happens is string

of concord nation will happen.

No, just check it. Yes. Says hello buy has.

So the next year adding bye string plus string

string Concord nation.

Suppose between them I need space. Yes.

Plus give a space plus and then one.

Now after this output including spaces

that was con coordinated.

Hello. After that space,

after that one, that means plus operators

allowed between the strings.

Similarly, another operator is allowed,

which is multiplication operator as strict symbol.

Okay? If it is applied,

suppose string, uh, I'm taking a string value,

which is iPhone.

I'm multiplying with the 10.

That means a 10 times the iPhone,

the string iPhone will be repeated.

Now check the output. Same iPhone is for 10 tap.

Okay, one not thing. Let's, let's say A, B, C, A, B.

I'm saying in two three, the same A, B, C is repeated for

three types like A, B, C, A, B, C, A, B, C.

So that means on stringing what are the,

what operators are allowed on string

plus multiplication operators are allowed.

So what I try with the yes minus three

inval operation, yes, by three inval

is only allowed operations on the string is plus 10 stop.

Okay? Similarly, let's take another different data type.

Let's take a list

here.

X equal, I'M writing list of some values.

10, 20, 30, okay,

yeah, x plus hundred.

So please tell me the data types here.

X list plus in, actually this is invalid application,

okay, this is invalidation.

So list two plus inte is invalid.

But a list, two plus list, try to do that.

The access list of this more list M adding.

Now access five elements, upper plus,

there are two elements in the list.

Now this is valid operation between list objects.

If you apply the plus list con coordination will have that.

I'll just check it. Five plus two total seven elements will

be con coordinated as a single string.

Check the output. Okay, list

to plus list is a valid application.

Same thing. Let's take Y equals stream

here I'm taking 102 hundred.

There are two values Y in two three

similar to what happened when I applied the multiplication.

Operat must a string is repeated for 10 times.

Okay, so now Y into three.

I said why has 102 hundred values a hundred,

200 are repeated for three times.

One more list will be created. That has total six values.

Just check it under 200, a hundred, 200,

a hundred, 200 likes.

So that means on list what automatic operators

one is plus second one

is multiplication.

Operators are so remaining operators,

not only these two are allowed.

Similarly, I'm testing with the tap.

I'm going to do the same trial on the tap.

So this is a symbol for the tap.

Let's say hundred, 200,

301 more thing.

I'm taking a new double

506 sample.

Now both are doubles here. Info plus new.

So info has three elements

and then new has two elements, totally five elements.

You should get it. So plus solves.

So plus is allowed between double solves.

Okay, look into this. X or x, X is a list.

Try to say info plus x. What up? And just check it.

Info is a apple here, X is a list.

Invalid top plus

list,

invalid other.

This info plus x is actually list.

I'm converting that into double now both are doubles.

Now check this is valid.

Optic info has three elements.

The X has five elements.

All the five elements have a con as a single double.

Okay, one more trial we do

with the multiplication operator.

Let's look into info again, I have three values

now info into, there are three values.

All the three values are repeated.

Again, two times that will be repeated. Now check it.

So a hundred, 200, 300, 1, time hundred, 200 per time.

Again second time.

So on top what operators are allowed

again plus multiplication operators are allowed.

Now I'm going to take the same operators trials on

the dictionary.

Object. D,

C, DA.

C means stands for dictionary is a

collection oft

and value based.

Let's take some the info for our bank account.

I'm taking your account number as key.

Your balance amount as well.

1 0 3 50,000 1 0 3.

That's a 90 plus thing that I have total.

Three account calls in my bank.

So account number has key that balance amount as well.

Okay, now 1 0 2 is depositing

10,000.

So 1 0 2 previous balances thousand.

Now if you are, if fees deposit ticket it 10,000 means

that the 10,000 should be added.

So for this bank of 1 0 2 plus equals

incremental assessment.

Okay? Plus equal equals to 10,000.

Now check latest values of it. Latest values of the 1 0 2.

So previously 50,000. Now his balance is 60,000.

Similarly, 1 0 3 is withdrawing 30,000

1 0 3 is with withdrawing.

30,000 previous balance 90.

If 30,000 is withdrawn, 90 minus 30, the balance,

latest balance should be 60,000.

Now bank of 1 0 3

here, decremental.

Assignment minus equal, equal minus equal, equal to 30.

Check the latest values of the bank.

Now this one,

so on dictionary, water operators are allowed,

all operators are allowed on

dictionary base.

All operations are cloud

like uh, plus minus,

multiplication, division,

Floor division, power values,

everything is a lot on the dictionary

but on the value part, okay, so previously you are adding,

you are adding some value to the 1 0 2.

You are directing some value from 1 0 3 like that.

But what if like uh, between the dictionary,

if I apply the operations, what happens,

let's say look into the bank.

I have three accounts. One more branch.

I'm taking, let's say bank two.

Think that this is of a different branch.

I'm taking 2 0 1 10,002 zero,

two 30,000.

Think that in the branch two bank is branch one,

bank two is branch two.

Between banks. Between these dictionaries, I'm going

to apply optimatic operation.

What happens? Just check it. It is not allowable.

Okay, so between dictionaries here, the comment

within a dictionary you can apply all these operators.

But between the dictionaries, across the dictionaries,

you cannot apply other medical operations.

Okay? So between dictionaries

we cannot apply

automatic operations.

This point you just remember

now I'm going to take one more collection, which is set.

What is set? Set is a collection of unique items.

I'm taking set one.

Let's say a hundred, 200, 300

and 400, 500 total of five values I have taken.

Even if I type, if I give duplicate values sheet one set,

I'm taking set to hundred

200, 700, 900, 300.

So here five values in the five values in the S.

Okay, other here, here the special operators.

If I say yes, one plus S two, what happens?

Invalid, same minus.

Okay. Yes. One minus S two here minuses alone. What happens?

You just check it. The values which are available only in

the S one but not available in the S two.

Okay. Yes. One minus S two.

What it is giving the values

which are available only in as well,

but not in

so here minuses.

So minus is not allowed of at least not allowed for couple.

Not uh, not allowed of between the dictionaries

but within the dictionary.

Minuses allowed because we not arrive with the draw.

The amount to be directed.

We have seen that, uh, in the previous example. Got it.

Similarly, I'll apply something. Just one.

The law operates just two. Yes.

One, just look at what happens.

Just by seeing this output, you just guess it.

I have given the pipe operator.

Actually it is a bit twice Operator. Okay, between the sets?

Yes. One is a set, yes. Two is a set between the sets.

I have given pipe operator. That means our operator.

What happens to the union? Office one and has three happen.

But duplicate simulator a hundred once union is

done in the S one.

Five elements. Yes. Two, five elements total. 10 elements.

But a hundred is taken only one time.

200 taken one time 300 taken 100. Okay.

400, 500, 700, 900. Like this.

That means here what is happening? What functionality?

Union. Got it.

Similarly, end operator.

I'm bit that S one a percent,

yes, you'll get common.

So in both sets, a hundred, 200, 300 a day. Common elements.

Understand? So this is what like uh,

a simple, I'm taking my friends.

My friends are Ravi,

Gary Siri and man,

and these are my friends.

Think that these are my Facebook friends

and I'm taking your friends.

Your friends are Ravi.

Ravi are also my friend and also and sir,

so between you and me, what are the common friends?

I want common friends now.

So you can say in Facebook,

technology mutual friends, look at this.

The common friends equal to my friends the intersection

of the intersection and percent.

And your friends.

Now just look into this common.

What is your understanding? Common friends from both.

Who? Only my friends. Those friends should not be with you.

Who are those GI money When

that means that these are friend of friends to you.

I'm your friend and gi money.

When are my friends but not with you.

Understand those are friend of friends party.

Tomorrow when I will log into the Facebook,

these three will be recommendation to you.

You may know this person. Got it.

That means the recommendation to you.

Your recommendation,

the recommendations for you is my friends,

minus your friends.

Now look into the final output. Your recommendations,

check the output, give money.

Previously you don't know them.

Now these three are recommendation.

Who are these three friend of friends to you?

Similarly, I read my recommendations, my recommendation.

I'm just reversing the previous acquisition.

Your friends minus my friends.

Now just look into this panel output of it,

my recommendation.

Got it. Jun,

were not only exclusively your friends, I don't know them.

So, but you are already my friend

and these three are friend of friends to me.

That's why these three are recommendation to me tomorrow.

If I sign into Facebook, these three will be recommendation

to understand.

So I'm not way here.

The allowed operators

on set one is minus

five and percent.

And what is one more thing?

So only these three operators we have taken. Got it.

These are the allowed operations.

We can do this for minus only available

with you, but not with me.

So in the case of my friends, manage

or friends means only my friends, but not your friends.

Got it. I want common friends.

I want common friends. And percent intersection

and percent is for intersection.

Here. Understand, let's apply the pipe.

What happens? Your friend, my friends

pipe, your friends.

Just check this. I got union.

Understand. So that means totally

your friends and my friends.

Okay, this is union is happening.

Finally, what is your understanding?

The single pipe is

for union.

Not single amp percent is for intersection.

I mean common to say common friends minuses were

for what?

Only available with set one,

but not with set.

So all these sessions we discussed about

in all our previous sessions we discussed about different

operators, different types of operators,

and how to apply them, apply them, and different data types.

All these things we have seen in one single session.

In the next session onwards, we are going

with the Stan statements.

Okay, thank you very much. Let's meet in the next session.

Bye-bye.

Module 6: Conditional Statements

Video: Conditional Statements Intro

Welcome to Python sessions.

In the last session we have discussed about

operations with all data types.

So in this session we are going to enter into

conditional statements.

Exactly. The purpose of conditional statements is

based on given criteria.

The creditor means condition.

Some block ops statements to be executed

if condition falls.

Some other block ops statements to be executed

here, a single block, uh, may single statement

or multiple statements.

If condition falls. Some other block op statements

to be executed.

I keep it for you in the notes.

So you can classify these e statements as below

a simple statement

and if and else statement.

The third category here, nest strips.

Nest strips and the,

the regular format of these three statements I will explain

it is simple.

If the format is like this,

if condition,

you know python coding rules.

If any line is expecting a sub statement,

the line should end with column then

the block of the statements.

This block my end single or multiple statements.

Example.

Suppose I'm taking a value as 10 B value 20

and I'm giving a condition if E greater than P.

If this condition is true,

I'm printing A is big.

Then one more statement. I'm printing in this block.

I'm giving two statements,

B small.

So this block will be executed

if the given condition is true.

Okay? Suppose if we given condition falls,

nothing will be exhibited here.

Okay? So try

to evaluate this one condition.

Evaluate and be valued. That means 10 greater than

20 is the condition.

So in this example, the given condition is false,

so nothing is executed.

For example, the value a hundred B value is 50.

Now what about this condition?

A hundred greater than 50.

Now in this case the condition, given condition is true.

Then it prints like a hundred is big, 50 is small.

So if condition is false, nothing is executed.

If condition is true, then only this statement is executed.

This is simply format.

Same thing.

Defend statement.

The second format,

defend else.

That means if the given condition is true, some statements,

condition falls, some other statements,

which format is like this?

If condition

think that is a block one.

That means all these are true statements.

Yes, some other

block of statements I'm giving treated as blocked.

All these are false statements.

So for these different examples we'll see

same previous values.

I'm taking a value 10 B value funding my condition is here.

If greater than B,

print Y is big,

yes,

print B is big

of course, in this example I have even only one statement.

If condition is true, a big's statement will be printed.

If condition falls, B is big will be printed.

So for this program, what is output?

What about year value then B,

value 2010 greater than 20 falls.

So that control will come to Y spot.

So what is the final output?

B is big B means 20, but 20 is big, is output of this.

Similarly, multiple statements.

I wanted for the same example. B value 10.

That's a B value 20.

So before printing you need to check the condition.

Then what would Small?

Small,

yes, I'm reversing it.

Print B is big.

Then Y is small.

So for the given data, that means given values of A

and B, what is output here?

Condition falls. That's why you're coming into yel spot

and the L spot, there are two statements.

So that uh, output will be like this, 20 is big

and 10 is

small.

In this way, conditional statement will be working

and there is one more format of it that is nas.

So what exactly the meaning of is if

within that other, if

one more.

If you can say if some condition,

if condition is true, you are checking

with one more condition.

Again, this is true. One more condition, you are checking

again else.

This is spo. This.

If this is IL four, the second if,

and this is I lpo the first.

If again I'm checking with other conditions.

For example it all the things based on the,

if the condition is true, you

are checking with other condition.

If condition is true, we

are checking with other condition.

Same thing with condition falls.

Also, you can check with our condition.

So that format in the next screen we'll see.

Look into this. I'm giving a condition here.

Think that it is true.

Some statement is executed if condition falls yells

and then if you are writing in stuff of that in Python,

the AIF combination,

the ALIF combination is AIF

Again, this condition also falls.

Again, one more LF than condition three.

In this one. This is

what simply NAS and alis.

So based on this nas some condition, some example will.

Once this explanation is done in the next session, we go

to the practical part of it.

This time I'm going to take 10, three values, value

and B, value 20, C value 30.

Uh, finally I want to find out what is the biggest value.

Also, I'm keeping EAB into the

competition, which is biggest.

I'm checking if you greater than me.

This is my condition.

If this condition is true, definitely A is big, B is small.

Then A needs to be checked with C.

Then if the condition is true, one more if condition, I'm

if you get it, then C.

Okay. Now if the second condition also true means definitely

A already bigger than B.

Now A is big, bigger than C also.

Now finally A is biggest rent.

A is big.

For example, if the second condition got false,

second condition falls means what?

A already bigger than B. Now C is bigger than A.

Then

print C is big.

So it's all that robot. Your first condition is true.

For example, first condition itself falls,

then come to this one.

Yes, I will do in the next line.

Look into this. The S were

is primary first.

After that, immediately I need to check

with one more condition.

In Python, the S and combination is yellow.

That's why here I'm giving yellow leaf.

The no need to say this one, right? Yeah. Okay.

First condition. False means what? B is bigger than you.

Then B needs to be checked with the competition with C.

Then bigger than C,

if this condition is true, print B is big.

Every is false. Print,

definitely C, S, P.

Now for the given values of 10, 20, 30, try

to evaluate this condition.

What about 10 greater than this value is 10

and B, value is 20.

So condition false because of falsity is coming

to it is reaching LC.

Okay. Now, B, value 20.

C value 30, condition two

or false 20, greater than 30.

That is false. That's why again, it is reaching L spot.

Now what will be printed funnel output is

30 is big.

Let's test with the other different values so

that easily can understand.

Now let's give the values in a place of uh, yeah,

I'm keeping a hundred in the place of B 250.

In a place of C I'm giving it 240.

I'll try to evaluate this

with without given values, not latest values.

Yeah, a hundred B, two 50,

a hundred greater than two 50 condition falls.

Again, it is reaching yes spot.

Now, B, value two 50 C value 2 40, 2 50.

Greater than two 40 condition is true.

That's why what will be printed two 50 is

okay, let's make it thousand this time.

This time it is thousand B value. Thousand B value two 50.

I think that it's two nine. What is the biggest 1000 value?

This yay thousand B two 50,000 greater than two

50 condition through it reaches this one.

Understand now competition with the C

thousand C value two 90,000 greater than two 90.

Again, condition through so that what we printed B,

that means panel output thousand days.

Big focus, especially if you look into this

example, nest.

If within that, if that is covered.

Again, both the things have been covered in this one.

Okay, simply these are the three formats of the regular.

So what are the three for points I try to recall.

The first one is simple statement.

Second one is if and else. Third one is nest, dips and L.

Okay, so in the next session we see practical part

of all these things step by step.

Okay, thank you much.

Video: Simple IF & IF ELSE Lab

And the types of uh, conditional statements,

like a simple if statements and

before that, uh, if and L statements.

So in this session we see practical, simple, if

and else

good plan.

So currently with the runtime time, five

possibly demonstrate.

We see demonstration about simpler.

I'm taking a value this hundred.

My question is like this, pay greater than

print

bigger value.

If condition falls, I'm not giving anything.

Just check this. A hundred, a hundred.

Greater than 50 condition True

because of two bigger value it has printed.

For example, I'm changing the value, okay,

in the next cell I will change it

and value.

So 40 greater than 50, the condition is false.

Here, condition is false, means nothing is executed.

We don't get any output of yet.

Okay, but condition falls also I want to execute something.

Then you have to give if and I statement.

If L statement, if condition

through some statement may be singular, multiple.

If condition falls, some other

statements keeping the same.

For example, a value a hundred D,

value one 20.

If you greater than B, print A is three.

If condition falls, then print

B is now check for this values a

hundred greater than one 20.

Actually this condition is false because of

False it came to the L part.

The print BS begins one 20 big.

Okay, let's do a trial

with other different values in the next Excel.

I repeat the same

code, it's 900.

We value 900 B values one point.

Now condition is through this time 900 greater than one 20.

The condition is true. Then it prints 900 is big.

For example, we both are equal values. I have hundred.

Now what is the output? What? What are the first condition?

A hundred. Greater than a hundred.

The condition is false here

because of false again, it reaches,

suppose you want to display three categories

of output like uh, a big B big or both are equal.

Okay? Such kind of things you want to print how to go

to the nests.

Okay? So in the next examples we discuss about nest labs,

but now let's continue with this one.

Now even the values are equal.

Definitely a hundred greater than a hundred. That is false.

Then it is printing 103.

Of course in these two values,

a hundred hundred is definitely a big, got it.

Similarly, if condition through multiple statements,

you can expect that is a block of the statements.

I'm just modifying that statement.

If this condition is true, print A is big,

then print B is small.

It also, I'm adding one more statement.

If condition is false, A is B is big, then A is small

like

C this at the same time,

in fact both are equal.

To handle that kind of equality, you need

to add one more manage.

So in the nest steps that is possible.

In the next session we discuss about nest steps.

Now with some scenario based example, we'll see

I'm taking a

student marks,

let's say have three subjects,

young one and three and three.

After this a pass mark in each subject

50,

the pass mark is subject is 50.

Now based on the marks whether student passed

or fairly one to five,

let's say student name something soldier

and M1 score is 70 M two score

is 30, M three score is nine.

Now I'm checking the condition if yum one

greater than hundred, but the power is the pause mark.

We said 50, minimum 50, greater equal,

two 50.

Then M two should be, should also be greater than article

equal two yam three greater than article two.

So here more than one condition, more than condition.

You need to give operators here

every condition must be true.

And then if

all these conditions are true, I'm rent

is passed,

which is my true state.

Yes. Print

name is five.

Okay, now check it.

Try to evaluate the condition for the surgeon here.

What about the first condition?

The monster score is 70, 70, greater than equal to 52.

Second condition, M two value 30,

30 greater than R equal to 50.

False one. False enough total will become false.

Got it. Now what is the final output here?

Surgeon is fake. Just check this.

Got it. Let's test with other data.

Let's say name is score is 60.

Now condition rule, second condition, rule,

part condition also all conditions are true.

Final result is now

what will be printed is passed.

Got it. So this is

what simply defend state

and one more example, let's take

your bank example.

You think that you are a bank customer,

your account number is one zero, okay?

Your balance amount is 10,000.

You want to withdraw

11,000.

I'm taking the condition.

Now, if a balance is greater than

with the dry

balance or greater than

or greater than a equal, then

I'm printing some statement.

Account number

has sufficient balance.

The memorial statement, I'm printing multiple statement.

Then immediately balance equal, equal to the two minus SQL

to automatical assignment.

Okay? All we discussed in the operator sessions,

balance minus SQLs, et

and also one more print statement.

I'm giving amount with the draw. Successful

is success.

It's all a sorting if the given condition is, got it.

For example, if the given condition is false, right?

It is printing you in tion.

Once your balance is

just balance and display.

Okay? Now here one account number is 1 0 1.

His balance amount actually 10,000.

He wants to get the draw 11,000. Check this condition.

10,000 greater than Oracle equal to 11,000 condition falls.

So what it'll show you inception for your balances.

10,000. Okay, now check this.

What is output in sub?

Let's do the test with other credit.

Now you are giving 9,000.

You want to withdraw 9,000. What?

What is your balance amount now? 10,000.

Now what is happening?

What will happen here? What is the condition?

10,000 or greater than to 9,000. True or false? True.

Because of rule it is displaying MS

assumption balance

and immediately what is happening from your exit balance,

it is directing with the dwell.

That means that 10,000 is a balance from

that 9,000 will be a deducted.

So that our latest balance now thousand and

after that we are printing confirmation as amount

with the dollars is successful.

And now immediately I'm displaying print the latest balance.

Your latest balance.

Again, we are 20 grand balance.

Okay, now check this output

actually as per given that I have 10,000 balance

and we are trying to withdraw 9,000.

Now here what happened? The first message

you have tion balance the amount with are successful

before that amount is with,

and then your latest balance is what all on

that way in real time you can use if and L statements.

Okay? So more,

more complex scenarios in the coming following sessions will

be seeing about this one.

This is simply a a small demonstration about if event else.

Okay? In the next example, in the next session

we'll discuss about nest statements.

Okay? See you in the next session. Thank you very much.

Video: Nested IF & ELIF Lab

Welcome to Python sessions.

In the last session we discussed about

a simple statement and if

and all statements with some scenarios, bank scenario

and one mark example.

We have seen that. So in this session we are going to work

with nest steps.

Depends if one condition is two

other condition to be checked.

If condition falls, one more condition to be checked.

So these kind of things are called negatives.

Let's say the practical part of it,

starting in a new notebook

is connecting with one time.

Now please wait.

This time I'm taking three variants.

I want to find biggest among the three variables.

So a value a hundred, B, value one 50 C value one point.

So for this I want to find out the biggest

first I'm comparing

and if greater than B is my condition.

If this condition is true, one more among A is big,

but again needs to be still need to be checked with the C

for that one more condition.

If within that one more, this is gone.

If you greater than C, if this condition is also true,

then big equal to yeah.

For example, if the second condition is false, then

greater than C, false means definitely C is big.

Why not with B? B,

B is already defeated in the first condition on That's

what the equal do we'll C, right?

Resolving will happen if the primary condition,

if if the first, if condition is true,

suppose first if condition got false, yeah,

then I'm checking with one more condition in Python Ys.

If that combination simply, you can write it as, okay,

suppose first condition falls means B is big among A, B.

Now B need to be checked with C.

If this condition is true, biggest being

big is C,

because competition between now B and C.

Okay, now finally I'm going to print the biggest

base, big

check the values given values.

Hundred one 50 to hundred, which is the biggest, which 50.

So definitely one 50 is output.

Here the biggest is one 50

and try to evaluate this condition.

What about this? I'm not greater than one 50 condition.

Let me explain. Clear

is what the given condition.

Check it the first step.

What about this condition?

I'm not greater than one 50.

This is false because of false.

It goes to L spot. What is the L spot for it?

They sell now be greater than C. What is the B value?

One 50 C Value one 20. Condition two or false? True.

Because of rule this will be executed,

then it'll come out of the state.

Then final printing is the biggest is one fifth.

For example, if I have given the values like this,

think that its values one certain.

Now check the first condition 100

greater than one 50.

This is false. So control comes to

I live now.

B. Value one 50 C, value

one 70, condition two or false again, false.

So that it goes to yes fault. What is the statement here?

B, equal to C, then it's value is one, seven.

Okay, now let's make uh, as biggest,

let's try with the sun value, which is 200 for the

now what happened to this condition?

200 greater and 1 52.

Because our route is coming to this spot,

now we value one per 200 C.

Value one 40 again, then yeah,

then it'll come out of the state.

Now finally, what is output? The weak is 200.

This is one more trial we do

and taking your name,

supposed name care

taking your salaries, your salary,

one lack 20,000.

So 12,001 lack 20,000 is the salary.

Now I want to find out salary grade.

The condition is like this fine

grade of employee

based on salary.

The rule is like this.

If salary greater than articles to one lack,

then that is grad eight.

If salary greater than articles to one lack

or greater than articles to 75,000

and is less than equals to one, lack less than one lack,

then grade is big.

If salary greater than articles two 50

and less than 75.

70 5K.

Then grade is C.

If salary less than 50, everything

then grade is C.

Now I have four options.

Okay, so based on the salary, I need

to find out what is a grade.

Let's say in the first example, I have one. One lack 20,000.

So for this I'm going to write the condition.

If I have only two options, a simple if

and yes that is okay,

but if I have more than two options, I need

to use master stepss here.

First I targeting grade here. Past time targeting grad A.

What is the condition for grad?

A salary greater than oracle Equal two, one lack

if salary greater than Oracle equals two.

One black. So let me write in the plain text letter.

I will keep it to here. If salary greater than oracle equals

two, one lack, then grade equal to,

yeah, suppose condition falls.

Definitely the value is less than one lack.

Then else if you know EL and if is here, yeah, leave.

Then I'm checking for the B.

If first condition falls, I'm checking for B grade.

What is the condition greater than equal to 75,000

and salary less.

Then one like no need to give the second condition.

Why? Because if foster condition is false,

then only you're reaching the second condition.

Foster condition falls means definitely the value is

less than one, like a lot.

That's why save S greater than equal to 75,000.

Then grade equals to B grade

still L save salary greater than Oracle equals to 50,000.

Definitely that is less than 75.

Then grade equals to C grid.

Suppose first of all, second, all third falls.

Then else definitely remaining all values.

Remaining value is definitely below 51.

Then grade equals to D grade

name salaries.

I'm printing that salary and

the grade is the grade.

Hmm? What is the salary for the name? One. Like 20.

What about this first condition? Two?

One like 20 is greater than one, like two.

That's why the grade eight. But now grade contents, yay.

Then it'll come out. What is output? Supposes name is Karen.

Karen Sal is one like 20

and grade is, yeah, for example,

our SAL is 90,000.

Then what about this condition? Condition falls.

Then it is reaching there nest the next condition

salary greater than 75.

Now it is true. Then it'll be executing this.

Then you'll get grade B.

For example, the value is 60,000.

The salary value first condition falls

or reaching the second conditions.

That is also false. Of which the power condition.

Now 60,000 greater than another equal to 50. That is true.

Now grade is C. The C will be printed here.

For example, we have 20001st falls,

second falls, third falls.

Then finally it is reaching yes here.

What is that grade D that is printed here.

Now let's check with this data with all possible values.

We'll test it

for Karen.

Actually we have given one lack 20.

Okay, now let's more for the value.

One more person name.

Let's say salary. 90,000.

From 90,000. What I got great. P.

One more test. Let's

say 60,000.

Thesal is 60,000. Which condition is two?

First, false, second, false third is three.

So that grade will be

C, C grade.

The last test, let's think the SAL is 30,000.

Now first one is false, second one is

false, third one is false.

Then finally, what is option grade D?

Check this. I'm running six.

Got it. And one more example,

I'm taking your details like name

and your department number also.

So those name money, but department number is that.

So I want to transform the department number

into department,

department number into department.

So was here. Rules are like this.

If the department number is 11, that is department.

And if it is 12, HR department.

If it is 13, think that there is a finance department.

So remaining

that 11, 12, 13.

For all the remainings, I'm expecting the department. Ms.

Got it. So this kind of transformation I wanted to perform.

Now plan this condition poster target

is marketing department.

If DNY equals to, this is equal check level.

If it is level, what is the department?

Department is going to market it? Yes.

If the first condition falls,

I'm checking with other department.

DNY equals to 12.

Then dynamic equal to hr.

Still I say D equals to

30 equals question.

Finance not,

not all, not then

Printing and message name is from

is from Department.

We have given 13. So name is money

and department number is 13.

What about foster nation here? False.

Then it reaches second. Again, false. It reaches third.

Now this is true. What is the DM value? What finance?

What is the output? Now money is from

department finance.

So this is all, okay, let's check this.

What is output money is from department finance

because 13 value.

Let's let's test with the different value I'm giving.

14 freeze. 14 past the second.

Third, all the conditions falls. Then finally other.

The output is money is from department other.

Come on, test the department number level.

Now it's value from marketing

testing with 12

output hr.

So this function is working fine.

Okay, so in this session we have learned about how to use,

how to write the code with the initiatives and nalytics.

Okay? So in the next session

what we'll discuss is different ways to write your

conditional statements about the same task in different

how many different ways that you can code it

and which is the best performance way of the coding.

Everything will decide. So how to do that.

In the next session we'll discuss about it.

Okay, thank you very much. See you in the next session.

Video: Different Ways to Write Conditional Statements

We discussed and did, uh, lab work with uh,

NA defender sif, okay?

And multiple examples we did with uh, nas.

So in this session, NA related only,

but what are the different styles of writing the code?

Writing the conditional statement. So we'll see.

So straight away we jump into the practical part of it.

Part in the first notebook.

Same previous task I'm taking here among pre values.

Find out the biggest value using nests.

First suppose A value 10. B, value 15.

The C value for, I want

to find the biggest value among these things.

First we see wave one, which is in

If you greater than because we are comparing.

And if this condition is true among AB A is biggest

now year two B compared with the C.

If you greater than C,

this condition is also true.

Then A is the weakest, then big equals to A

suppose the second condition falls.

If second condition falls means A small B is big, C is big.

Then biggest is big, equal to C.

It's all apart. The first condition is true.

Suppose first is also false. First condition is false.

Then B is big, GA is small.

Then B2B compared with C, Y, C, Y.

In Python elevated big, greater than C.

If it's true the gig was true, B,

still else will do.

Yeah, so see now finally you are printed.

Biggest is B.

Now check the things here. So we have the values stand

15, 14.

Now among all these things, 15 is the weakest.

That is expected answer check. The output biggest is 15.

Same program in different style.

We second way

this time without writing a steps.

We try to do that if he greater than B

and he greater than later within the single condition.

We are checking that one. If this condition is true,

big equals to, yeah, I'm not writing L

and u writing a separative statement.

Now this time I'm targeting B.

If A B2B has a big,

the requirement is B should be greater than A

and also B should be greater than C, then

B equals B.

Similarly, if this time I'm targeting

C, C greater than it

and C, greater than B

equals to six.

We did not write an E, C even else.

Also, we did not write. Now how many statements?

This is separative statement.

This is separative statement

and this is separative statement.

Now finally, I'm printing. The biggest is,

is now check this.

I got same reply,

which is 51 more

modification I do in the group.

Now if this is third way, one condition,

I would like to try reduce here my technique,

I'm reviewing the lost condition before starting.

If only I'm fixing some default values.

A big chase, big equal to see.

Now observe this how this logic will work.

Try to understand.

So this is what the statement, what we have.

So what the values we have again,

B, value 10, B value 20 C value.

Suppose B value 15.

We have again, let's say

what is a C value we have taken previously.

14. We have again, let's skip 14 more.

C has 14. Now initially big value,

big equal to CV set.

That means the big value is 14.

Now check the second condition.

The festive condition, yeah, greater than me means 10.

Greater than 50

and 10 greater than 40.

Both are false here. So entire cost condition is false.

Now come to this one. That means 15 greater than 10

and 15.

Greater than 14. Both are true.

So finally, this is so latest value of the weak,

previous old value of the weak is 14.

Now latest value have the biggest 50.

Now what it means, six 50 on

that where this logic works.

Okay,

and one more way, very simple way.

I will present. We have three option either whether A,

B, C among these three, anyone is the weakest.

Okay, first time fixing anyone as a default value

as I'm fixing C as a big.

Now my condition, if yay, greater than big

then big equal two.

If CB greater than big,

then big equal two.

That's it. Only two conditions, two separative conditions.

I'm giving here. Final.

I'm printing the biggest, is

now still the same answer we got Now try

to understand how it works.

I I will explain here

to this same previous values, the value 10, B,

value 50 C, value 40 in the beginning.

What is a big, big equal to 40?

After what is this condition? A greater than big.

A greater than big means 10. Greater than 40.

Now current value of the big is 40.

Now this condition is false so

that the statement won't be executed.

Next condition. B value

15, big value.

14. Between greater than 14 condition through.

Now this part will be executed.

Now what is the latest value of the week? Big equal to D.

D means 15. So finally 15 will be correct.

Okay, now

let's check with other different values.

Example, this time let's make A as big.

Let's say A is 70, B is

35, and C is 40.

The biggest should be 70 in the beginning.

C value big value 40 because big equal to C.

You said that's why it is 40 now. Now check this condition.

A value is 70, 70.

Greater than big, big is 40 enough condition. True.

Then the statement will be executed.

Now big value big equals big value.

70. Now 70 will be correct,

but this is operative condition.

Once this, if condition is executed again,

the control goes to the next step.

Statement. This is operat.

Separate in vendor statement, directly won't,

it won't reach this statement.

So that's why. But up to now, latest value

of the big is 70, previously 40.

Now big is war in with the 40.

Now check this condition,

B value 35, latest value of the big.

So here it is 70, right?

Let us value of the week 70, this condition false.

Still the big value is 70.

Finally, what will be printed 70 will be printed.

Same thing. This time we try to make a Cs B guest try

to evaluate this condition

that say this time you value 70 B, value

90 C value one,

10, yeah, C value.

Now in the beginning, B, value one 10,

the first condition 70, greater than a value 70,

greater than one 10 box.

This won't happen. I check the second.

Second condition,

B value 90, 91, 10.

Again false. This won't happen.

So that latest, so what is the current value

of a big, which is C?

That is what will be printed. One 10 will be printed.

Understood. This is working very simple ways.

The last one. Okay.

Now among the about three ways,

which is best in this situation, we try to evaluate.

So look into this. The first way is this one.

Second way, I'm keeping here.

The third way I'm keeping here.

So first way, if you greater than B,

if you greater than C, B equals

to DY,

big equal to C.

This is four positive yel, C, L, F,

B, greater than C, then be equal, equal

to B.

Yes, be equal to C. This is fast away.

So how many conditions will we checked here in this,

for example, first condition is true.

First it will check the first condition.

This is condition number one. Check number one.

If it is true, it is checking this.

Then if the condition is true, the statement,

if condition falls, the statement will be executed.

So if the faster condition is true, I have two options.

So here, number of checks, only two checks.

For example, if first condition falls,

then directly will come to this spot.

Then this will, this will be checked

in case of condition two.

It is checking one, two conditions.

If in case of condition falls, it is checking one

and three conditions.

Okay? Always it is checking number of checks here.

Two, in this example, look into the second way.

If you greater than B

and greater than C,

then big equal to Y.

If you greater than, sorry, be greater than Y

and be greater than C,

then big equal to B.

If C, greater than Y

and C, greater than B, then B, equal

to C.

Now how many checks will happen? Condition?

Check one condition, check in this way.

I have three statements here, two here, two here, two.

Finally, six conditions are checked.

Think that in the case of first case, in the first way

in first way, I have young options.

I have three options. That's why number of checks, two,

if I have young options, number

of checks will be and minus one.

If I have 10 options, number

of checks will be 10 minus one nine in the first way.

Hmm. Whereas in second.

Second way, if I have three options, number of checks,

three into two, if I have an options, 10, two and minus one.

If I have 10 options,

10, 2, 10 minus one means nine totally,

90 checks will happen in second way,

which is best if I have more number of options.

First way is best and compared with second way.

But third way is still more simple

because what is a statement we have kept

and the thoughts are third way is equal to B or C.

Then you are targeting, you're comparing with the

if a greater than big then big equal

to if C, greater than big,

sorry, big, greater than big.

Then big equals two big.

So here, condition check one, condition,

check 2 1, 2 options.

And the first way, two options in the second way,

two options, okay?

In the second wave, two options.

So both of us, so, so wave one, wave three, both are best,

but wave three is more simplified, especially

for this kind of situation.

I can use thought because more simplified code.

So this is what like, uh,

today's discussion in this session, discussion

in the next session, we go with the

real time bank scenario with initiatives.

Okay? Let's see. Thank you very much.

We meet in the next session.

Video: Real Time Bank Scenario

Welcome back to the Python sessions.

In the last session we discussed about the different ways

to write conditional statements.

Okay? Now we see a banking example scenario.

So by using these, I'm opening a new notebook,

just connected me the wrong time.

That here, the use cases

I taking a bank accounts, for example, bank customers.

So I'm taking help of a dictionary here.

So initially bank is not having any customers.

That's why this dictionary isn't,

no, I'm giving something I code number.

Let me write in a plain text, lateral sub, substitute

that into the notepad, cell notebook cell.

Then we'll execute it.

So what account number one zero

and the amount something 10,000.

That mode, that means type of operation. He wants to pay.

For example, he wants to deposit.

So account number,

bank account number 1 0 1.

He wants to deposit 10,000. First of all, what is the rule?

The account number should be existed in the bank.

If it does not, uh, if we,

if it doesn't have the account number,

the account number to be created.

Okay, so if

that 1 0 1 account number is not available in the

dictionary, first we check in the beginning,

we don't have any account numbers in the dictionary.

If not available, we need to insert that page.

So for that pay account number as key that the amount

as value allocate.

Okay? If account number

in, in operator is

the bank, bank is a dictionary.

If we is available, we need to add that amount.

Suppose if it is not available,

if account number not in the bank, then bank

of account number equal two.

Up to now what happens? Just second.

This is simply my condition

of this.

Print the bank. Hmm? Compare the values.

1 0 1 available in the dictionary.

Not available if not available.

The given condition is true

because we said account number not in bank,

if not available available.

We are making account number as key amount as value.

Just check it. What happened here?

10001st time inserted. That means his account is so bad.

Got it. But the same thing I'm giving 1 0 2.

Same

still 1 0 1.

Let's say 20,000. Now if I run

this, what happens?

Nothing happens. What is the expected functionality?

1 0 1 has already account his balance amount is 10,000.

Once again, he's the doing deposit.

That means if the account number available in the bank

to the balance amount, the 20,000 should be added.

Now his latest balance should be

10 plus 20, 30,000.

Suppose if not available.

If you're available then else

the given condition is if not available.

If that false means already account number is already

available in the then else bank of account number

to the existing value, we are adding that amount.

What is that amount? That

20,000 is available in the variable amount.

Now check it now.

Latest balance of 1 0 1 is 30,000.

Got it. So all this operation

to be done if the operation is deposit.

Got it. That's why I want to add one more condition for

that after this.

If mode equal to deposit,

if mode is equal to deposit

there only, I'm checking if

that account number available in the bank or not.

If not available. We are inserting if available,

we are adding bank account number plus equals

to Yeah here if within

that one more if necessarily.

For example, post condition falls.

Post condition falls means what is the meaning?

That is not the deposit.

That might be withdrawal,

that might be check balance like the remaining operations.

For example, if fast condition is false, then I'm coming

to the else yell

and then immediately if yell, if you know else

and if you know that is yellow I mode equals too.

This time I'm checking fur with the drop.

If mode is withdraw once again to withdraw money,

what is the first condition?

First of all, you should have account come on, check

the account available or not.

If the account number not in bank

or if account number in bank first you are checking if

account number in bank, then you are allowed to withdraw.

For example, your record number is

1 0 1 1 0 1 already there in the bank.

Got it. Then to withdraw the money,

what is the under requirement?

You have balance of 10,000.

You want to withdraw 11,000 in sufficient funds right

now in that case, I'm checking our balance availability.

If the amount

is less than a equal to the available balance, how to get

that available balance back off.

I contact them. If they given our requested amount is less

than a equal to balance amount,

then only with the drug to be done.

So if fit the drum means previous balance 10,000,

suppose you are withdrawing 9,000, 10,000 minus 9,000.

Now latest balance should be thousand.

The piece of bank of amount, bank of account number

minus equal equals to,

okay, for example, this condition is false.

What is the meaning? If this condition false means

you don't have sufficient funds.

Then I'm printing a confirmation here.

Print

insufficient funds.

Your balance amount is I'm printing that

bank off account

up to now just check it.

The center processes were fund availability.

If fund available you are withdrawing.

So that amount will be directed from your balance

and amount if not available.

It is showing you some error message to the user points

and balance showing current balance and models.

It's all a story. If this condition is true, for example,

if that condition is false, if this condition is false,

I mean this condition, what does the meaning the account

number not available in there then?

Then it is printing invalid account number.

Let's create account some confirmation message

today, right?

So this entire work will be done

if this condition is true.

If mod is with the truck for example,

mod is this condition is also fall false.

That means mod is not deposit, mod is not with the truck.

Then mode might be balance check.

I'm giving a if mode equal to balance.

Okay? Then if more equal to balance also whether

that account number available in bank or not.

I'm checking if account number in bank.

If account number in bank

if available I need to dis display.

I need to print the balance amount, right?

Current balance,

bank of account number passing.

If I pick pass the key, you will get value.

The value is balanced amount. For example, condition falls.

Again definitely account number not available in the bank.

Then the message invalid

account number, which is the A CM.

That's it up to not in the high level.

There are three conditions.

Condition one if false, more equal with the if false,

more equal to balance.

Still, if it is false print

invalid operation.

Think that only allowed operations are three a deposit

with the drug balance set.

So for these three things we are going

to test with our application.

I'm replacing the center part,

the previous value.

Second number one, zero the amount,

let's say say 10,000 more

equal to initially deposit.

So before this sale, I'm keeping bank as empty additional

in the last every time.

Let's bring that back a protectionary. Now run it.

So first time 1 0 1

with 40,000 amount the account is created.

Got it.

Why? It is what?

First we run this sell because already 10,000 is added.

After that 20,000 we added 30,000.

Now this time 10,000 is added. That's why it is showing 40.

Now let's start it from the beginning. Now bank is empty.

Now run this first time 1 0 1 10,000.

Just to validate this condition in this moment.

1 0 1, not available.

First model, equal deposit condition rule.

Okay, then it is checking that account. 1 0 1 not in bank.

Yes condition rule because not available.

Then you are in setting. Okay,

now run it and see it.

1 0 1 10,000 is inserted.

Bring that next time the account number is 1 0 2.

I'm giving one like 20. He is trying to withdraw money.

Hmm. What about this condition? More equal deposit. False.

Then it'll come to this one more equal to

with the drug condition.

True the account number in bank

1 0 2 not available in the bank condition.

False Because of false You'll come to here.

What it'll say invalid account number,

please create account.

Check it. Now check the message.

Invalid account number please create accounts.

Okay, now he decided to create account

making this as a deposit.

Run it. Now check the condition.

First condition through 1 0 2.

Not in bank. True. So that bank

of 1 0 2 equal two, one like 20.

That means account is great.

Run this.

So double time we ran it.

That's why this became one like 21st time

after that one like 20,000 is added.

So two like 40 because we did run two times.

I want to withdraw. This is condition.

This time. 1 0 2 wants to withdraw.

Withdraw one like on now check it.

Condition falls. It'll come to here. Condition through.

Then it is coming to here.

Now account number 1 0 2 available in the bank.

Then it comes to here.

Now whether the requested amount is less than Oracle equal

to balance amount or not your check requested amount,

one lack balance amount or two like 40 we have.

So condition is true. Then from the two lack 40

from the two like 40, that one lock should be detected.

Then it is coming out.

Then it is showing you, it has to show you latest balance.

Just check this. What are the latest balance

now one LA button.

Okay, one more time. 1 0 2. Wants to withdraw dry again.

This time it is trying to withdraw one lack 50,000

but his balance is how much one lack 40,000.

Now check that condition again.

Condition, false condition through condition

through condition.

One. Lack 50,000, less than

or equals to one lack 40.

Condition. False because of false.

What you'll get inception, balance. Balance.

Your balance amount is the running balance.

It'll should now run it and see the result.

Inception funds your balance amount is one like 40

because we requested one like 50.

Okay,

no 1 0 3 wants

to check his balance amount.

Forget about that amount.

1 0 3 wants to check his balance,

but in fact 1 0 3 account number is not there.

So where condition will be failed, this condition falls.

This condition falls. Now this condition true

again this condition 1 0 3 in bank, not available.

So that falls. It'll reach this point.

The message will be invalid. Account number.

Okay, invalid account number. We did not run it. Not run it.

Invalid. Account number 1 0 3.

Got it It now 1 0 3 decided

to create account with the 10,000

is trying to deposit

condition true condition true.

So that account number will be created.

Check again. Now third account holder will see 1 0 3.

1 0 3 amount deposit,

invalid operat.

Check the mode again.

Account number 1 0 3 amount 10,000.

Mode deposit if mode equal to deposit condition through

previously account number 1 0 1 not in the bank.

That is also true so that it has to be displayed. Why?

It is saying that invalid operation

somewhere, the spell mistake we have

invalid operation is taken.

Look into the L spot now.

First time one zero to create it.

Once again, retest all the operations.

1 0 1 first time. So deposit to be done.

Now 1 0 2 wants to withdraw either 5,000.

Now check the condition once second please.

False mode with the drop.

I want to withdraw mode to be with the drive

with drop this condition.

True account number available. Yes, true.

5,000 less Oracle to 10,000. True.

So that balance should be detected.

5,000 will be detected from 10,000.

So the latest balance of 1 0 2 should be 5,000.

Perfect. Similarly, 1 0 3

wants to check the balance.

Balance. It should say as invalid.

Uh, like uh, invalid account number

because 1 0 3 not available in the previous yes.

Invalid account number 1 0 3.

Think that now 1 0 3 is going to deposit.

Now deposit a deposit of 50,000.

Check the operation. 1 0 3 is created.

Now 1 0 3 wants to get the drop 60,000

now inception balance to be there.

Check it. So we did not change

that mode.

That's why another 50 60,000 is added to 50,000.

Now 1 0 3 latest balance, one lack 10,000.

Now he wants to withdraw with the draw.

Let's say six lacks,

but in fact he has only one lack 10,000.

Now check the output

in tion funds or balance amount is one lack 10,000.

Okay, suppose

1 0 3 wants to do transfer,

but we did not write the logic for this.

We have it in the logic only for deposit

with the draw and check balance.

Now the first condition is false.

Second with the drug condition falls, third one,

balance condition also falls.

What is the final option then? Invalid operation.

We got invalid operation in this way.

You can use these nest steps in our real

time regular applications.

Okay, so in the next session we'll discuss about

10 array operators.

So same if condition, you can write in another style.

That is ARI operators. Okay? Okay, thank you very much.

Meet you in the next session.

Video: Ternary IF Statements

With the initiatives in this session we are going

to discuss about tari if statements.

Let's do the practical step of it.

So tari format, uh, of if statement is like this.

First approval, true expression,

then if condition,

yes.

False expression

after hearing in this statement.

The true part or true expression in the left side

and the false expression is after health.

Okay, just let me keep it into that comment.

A simple example we see taking general value,

which is the, now my true expression, I,

I want to written mail.

The condition is yeah. Yes.

I want to written return other

here the value is young,

then it is written true value, which is male.

If is not am it is reding false.

For example, if I give some,

so it is giving the condition is false, so that's why

female, okay, so I want to handle that kind of situation.

So what if gender values,

yeah, I want male.

It is, yeah, female other than I'm men,

like other, other characters out there.

I want in this way. I want to transfer.

Now plan the condition. I want male.

If gender equals to, yeah.

Yes, you need to check one more condition.

In the case of ball, I need to check with other condition.

Again, I want female if gender is equal.

Yeah. Still else I want invalid.

So here I'm, this is my statement.

Previously I'm giving something gender, young,

not check this exactly.

I got married. Suppose I'm giving,

I got female other than I'm in family.

Let's say X, it is inval.

That's a y. Inval.

For example, capital. I'm giving, I want

to handle this house, but here

I'm checking with small letter.

So in this case, what even whether it is upper case letter,

lower case letter, transform into,

then check it again Here also dot lower.

Then you get mail.

If I pass small, small, first capital FI got failure.

Okay? So how this condition is working

is checking this condition, this condition falls.

Then it is coming into else

and the else again, there is one more condition.

Okay? Now check this condition. This condition is true.

Now because of truth, it returns

your first condition falls, second condition falls.

Finally, it returns inval in this way. I have three options.

I can plan like this. Let's break it down

to the multiline.

What happens a second?

Now please like this. Try to say backslash

here, backslash.

Now we can do the same thing.

So backslash is for like a two split state window.

Got it. Left side through value, right side falls value.

This is the format. Based on this example,

you try to think about it.

I want to transform one thing.

I'm taking department number,

department number two, department name.

I want to transfer. Suppose 11.

11 is marketing

and always

HR 13.

It's remaining other than 11, 12.

But remaining all the things I want this as other.

Now using if statement, I wanted faster target

marketing, I want to marketing.

If department number equals two level,

yes, one more condition to make it easy.

Come to the next slide by saying, okay,

previously department is not there.

Let's skip some department for our testing department.

Okay? If it is false, now target hr.

What is the condition for this?

If department number is equal to 12, yes.

Okay, if it is not 12,

Again, I'm checking for one more thing.

Come to the next, next finance.

If department number is equal to

then it returns for finance still not false.

That means still condition. False. Then this is the fault.

Okay, try to do this now

for the 11 I got marked it.

So how this condition is true left side true values there

that stated understood.

For example,

I'm giving 12.

What about the first condition? Falls then come to the L.

In the ls there is one more condition.

Department number equal to 12 condition is two.

That's why I left outside. I got H.

Next 13 I got finance.

So first of false, second false.

Then else else the finance department number equal to 13.

The condition is two. I got one.

Suppose I'm giving 50, it's not 11, 12, 30.

Definitely 1, 2, 3.

All the condition spots definitely will get other.

So this is another step.

One more cost per I want

to transform salary into salary grades,

for example, greater than articles to one.

Like I want A grade A

greater than to 70 5K

and less than a hundred K.

That means one like then grade B greater than articles

to 50 K and less than 70 5K.

Then grade should be C.

If all the values which are less than 50 K

than grade is using tari.

If statement, I want this now faster target.

Yeah, yeah, agree. What is the condition for it?

First take some salary.

Salary put, let's say salary is one like 20,000. Come on.

Target the A grade, what is the condition for it?

If salary greater than equals two, then yeah,

yes, come to the next level.

Now this time target B grade.

So for this, if salary greater than article 2 75

B Ys target C grid

IT salary than article equal two 50.

Then C Ys and fix as C grid.

This is a 10 hour apparatus.

Now check this for one LA 20. The first condition is two.

That's why it written a grid. Examples is 90,000.

Then first of all, second is true, then B grade,

then I'm giving something 60.

First of all, second condition falls auto condition is true.

So that you'll get C grade.

I'm giving 41, 2, 3.

All the conditions falls. So final option, take grade.

Okay, so the center thing thing,

you can direct them into one variable card.

Great.

Upgrade. Your salary is

salary with

grade, just check the output.

No, your salary is 40,000.

With grade D, I'm making it as one like 30.

So this is the style of 10 operators.

Okay, so in last session we have seen

last three sessions S and

after the different ways to write a conditional statement.

This is also one more way

and real time bank example scenario we have in in this

session we have discussed about scenario statements.

In the next statements already, we have used in operator

and not in an in operator,

but with help of inland,

not in one more scenario, one more session.

We'll, in the next session we'll see about each statements

with the in and not in operators.

So we have a name for them for the in knotted,

what these operators are called membership operators.

Okay, with the membership operators we see nests,

the statements.

Thank you very much. Let's meet in the next session.

Bye-bye.

Video: IF Statement WITH IN & NOT IN

About statement with practical.

So in this session we are going

to talk about membership operators

and let's include these membership operators.

In the e statement. What are the two membership

operators in?

And Martin, okay, let's take,

I'm taking to the updates for here.

These are my friends list

money.

And will I think that these are the Facebook friends

or some, any other social media friends?

And I'm taking your friends,

your friends, like

who are the common friends in this?

Above this list?

My friends count sport.

Your friends count is far, but there are common friends.

First I want to find out common friends.

For example, let's text. Susan is actually my friend.

Whether he's available to know friends list

or not, I want to check your friends.

So TRO is not available in your friends.

Same thing I'm checking with not Ravi.

Ravi is my friend and also he is your friend.

So that Ravi is our common friend.

This clue is enough to build common friends.

That's a common friend initially until time.

But I look through my list for Yu

or MF my friend in my friend.

So each trend is coming into, so here's one mistake list

my practice.

Each list, I'm checking in the your if EMF

in your

friends in the ation.

What is the value of? Let's make it as you have.

You face a short term

in ation.

What is available in our friends list? No.

If not, I'm not doing anything. Suppose second iteration.

Money, money also not with you.

Condition falls, extra ation

is available in your friends list.

Then I'm appending that

common up.

Up do, yeah. Now check the final output print.

So Ravi, we no other the common friends between you and me.

Similarly only my friends ever.

That means those are exclusively my friends, but not all.

Eventually I'm keeping empty list.

This time I'm using negation in the place of in operator.

I will use, not in operator, just check it

for my friends.

If Yf not in, if you have not your friends,

It is true.

I'm opening to only my

friends don't up and up.

Yes. Now check it.

Only my friends

check this here.

Virgin and money are my friends, but not your friends.

On the same style. I want to build

only your friends,

only your friends list.

I want to find

initially I'm the list.

Now try send the loop to your friends

or your friends.

Check that to your friend is not there in my friends list.

If you have not,

not in my friends, then add

to only your friends do up

and off, up and off.

What Jeff,

the list is ready.

Check the values of only your friends.

So Dvia, Karen LaMi, your friends.

Now I'm printing all the things at one place on spot

Common Friends that is in common.

Friends, your friends.

And I can only your friends.

My friends.

Only my friends.

So take this list between you and my conference.

You have your own exclusive friends.

That is the Blackman and my friends Astrogen and money.

Okay? So in this way you can take help

of membership operators, like, uh, in and

nor you can use them into give statements

and a statements, any conditional statements.

Okay? So this is what our discussion

in the next session onwards, we go with the for loops.

That means looping statements up to now like, uh,

controlling statements have sent from the next session on

what we, we'll start discussing about looping statements.

The looping statements are like loop while in Python.

We have two loops. Let's meet in the next session.

Thank you very much.

Module 7: LOOPS

Video: Introduction for loops

Hello, welcome to Python Sessions.

In the last session, we completely discussed about different

ways how to use the uh, conditional statements.

So this session onwards,

we'll enter into the looping statements.

So faster we took some basic introduction to loops.

Like loops are required.

Upload the scenario. You have,

uh, some three, three

or four elements in one list, like a hundred, 200, 300.

Take that. The list name is x.

I want to print each and every value.

So suppose without looping,

what is the process without loops?

First of all, I will print x

of zero, then 100 will be printed.

If I say print X of 200, x

of one, which is second element, 200 will be printed.

Print x of two

or minus one, then 300 be printed

here in each and everything.

The statement is print,

but the thing, the same print

statement you have given three times.

What if, if it is a hundred times, okay,

you cannot say manually print off X of zero, X of one, x

of two like this up to X of hundred.

I cannot say. So there the looping

statements will come into the picture.

So advantage of loops is

to execute a statement

or block of statements to execute

a statement

or block of statements.

Block of statement with mul multiple statements.

It relatively the number

of times

in this case you'll be using these loops.

Basically in Python you have two types of loops.

One is for loop. Second one

is wild loop.

So what is the difference, how to use them?

You will see, now

I'm going to talk about the follow loop.

This follow loop be trades through a sequence.

Here. Sequence can be at least

or dictionary or even it can be a

set simply here.

Sequence, you can general meaning you can say it

as a collection of values.

So while working with the collections,

always better practices with funds.

And second thing is while loop here.

So if the given condition is true,

then only we'll enter into the loop.

So that means before starting the loop

only it'll check the condition.

Okay? If the condition is true, then it enters the loop.

If condition falls, it'll come out of the loop.

If the given condition is true,

loop ation will be stored.

If condition falls,

loop execution will be te.

A small example we'll see with the far loop

and in the next session we see practical part of it.

After this, the first exam, I supposed think that I have

five values here in the X where A is a list,

I want to print each way running a workload like this

for a fee in X exam site.

So here actually X contains like, uh, five values.

That's why it runs for five iterations at each iteration,

value is coming into fee.

Here we can give you anywhere.

So in the ation, what is the value of V 10?

I'm simply printing that V.

So without loop, we have to give five times print statement.

Now with thought loop, we have only one time.

So that is like a for V in X print of now it'll be like this

in the first iteration, ten second iteration, 23rd,

duration 50, next iteration, 60 and that hundred.

This is all. Okay, so list has five elements.

That's why five iterations run for five iterations,

the same style

while while loop.

Let's see the syntax and uh, how to operate this.

For example, task is like this

numbers from zero to

or one to 10.

So here the initial value of I some number, I'm taking

initial value I'm taking as well

because my starting point is one here

and I'm giving a condition.

While the condition is I less the

level because up to 10 I want to print it as well.

Less than 10 is condition.

Now check in the beginning, before the loop, I value one.

What about the condition? Condition is true.

Then you can print it.

Print five. At every threshold, I'm giving one increment

to the i i plus equals two one.

So first one will preprint, two printed,

three printed every I values keep on incrementing.

And before going for the next

generation, it is checking the condition.

If that condition is true, then

only fill enter into the loop.

Like this nine reached, 10 reached.

After then 10 I value will become 11. 11.

Less than 11 condition falls.

Then if you break the loop, it won't enter.

So that final, I got this loop exhibited for 10 times.

That means 10 iterations.

In this way, based on the condition, whether loop

to be started or not, to decide that you need

to use while loop

or else like if your sequence is having fixed number

of elements or some, a number of elements you want to work

with each and every element, then you will go with for loop.

Okay? This is the clarity between when to use for loop

and when to use while loop.

Okay? In the next session we see the practical part of it

and how to use these uh, while loops, more examples.

So we'll perform. Thank you very much.

Between the next session.

Video: For and While loop basics lab

Welcome to sessions.

In the last session we discussed about introduction

to looping statements.

As we discussed, we have two types of loops for loop

and while loop and we have got the clarity when

to use what type of the loop.

Now let's implement some practical stuff with that

PA loop format is like this

are variable in

any sequence already.

I told you sequence can be of anything.

It can be listed double dictionary set.

It can be of anything. That's why I simply say it as a

sequence and give the block of the statements.

So

this is basic format of the par.

So with this we do examples

and the Be example, I'm taking a list.

Let's say XX is having some five values like

10, 20, 30, 40, 50.

Check this line of x,

but five values that we have.

Now I want to iterate loop through all the sequence

part V in X.

So the Excels five values so

that the five iterations will happen

at each ation value is coming into V.

We are printing that.

Just look into this. So what if there is no loop?

What you have to say Print X of zero print.

Same thing like a printer X one.

In this for five statements you need to give.

So suppose what if, if you have like a hundred hundred

values, we cannot give a hundred print statements of here.

So simply loop at loop advantages to execute a statement,

a block of statements re okay, that is advantage.

Okay, and for the same sequence, one more example,

we'll see, look into the X value side.

I want to find out what the sum,

of course you have a predefined function thing

that you don't have any predefined function.

How to do, how to write the code within one.

I'm creating one initial variable.

So before looping, the total value is zero here and

after that I want to access each value for V in.

Yes, each value is coming into the V.

That value VI want to add to the total. So the total plus

Sequence.

Now this runs for five times auto.

Of the five times the total value is ready.

So 10, 20, 30, 40, 50, expected one 50 just like this.

Got it. Same kind of uh, demo we see for while. No.

For example, I want to print the values from one to 10.

So the starting value I, which is one I'm giving

while and then condition.

If this condition is true, then the block

of the statements, this is the format.

So I want from one to 10, that means

I hear the condition I less equals

to level.

After. If the given condition is true,

then only we'll get into the law.

Hmm, come on, check it please.

In the beginning, I value one,

check the condition one less can true

because of true you'll get into the low.

So in this, I'm printing

at each, I want to give an increment for that time.

I plus equals two. Just check it now.

So in the beginning I value one auto

accomplish of the fast iteration.

I begin to, again, it'll check the condition.

If condition two, then only it'll get into the loop.

In this way, 2, 3, 4, up, 2, 9, 10

for the 10 also conditions.

Two, once it entered into the loop,

after 10, 10, ation high value will become level

before going for next iteration.

Again, it is checking the condition 11 less than article two

10 condition falls.

Then it'll break the loop.

Just check where it is, not come

check all the valve one, two, and

whenever I reaches 11 condition falls,

then stop the loop.

But this is a basic difference

between follow loop and viral.

One more example we see with the wire,

I'm taking like a,

I'm asking the system for some text.

Let's say continue. We have predefined. Uh, what is there?

That's why want to continue.

This is my variable. Initially I'm giving blank back.

My condition is like this one to continue,

not equals to the word stop,

right?

I say hello and just a test.

Now I'm taking some input.

The input is into one, two,

continue input off.

Do you want to continue?

Even though user has typed upper case letters,

that is cannot into lower case

because of now just check this execution

initially that entered into the loop so that a hello printed

then it is seeking for user input.

I'm saying that yes. One more time.

It entered into the loop.

I'm saying stop now.

Next time the condition falls, the loop is break.

So conditionally want to control your loops.

Then while loop on a sequence to each

and every value you want to work, then go for Far Loop.

This is a basic difference between for loop and while Loop.

I hope all of you understood about this.

In the next session, we go

with the different scenarios based on the

for loop and while loops.

Thank you, Lammi there. Bye-bye.

Video: Finding sum and counting loops

Hello, welcome to Python sessions.

In last session we have seen some basic lab with the

for loops and wire loops.

Okay? Now with the for loop

or any Y loop, how

to perform the different aggregations fields,

the aggregations functions like a sum count.

So in this session, without using any predefined functions,

how to find the sum and count of a list.

We'll see, let

me take the rush list here.

Same previous value, 10, 20, 30, 40, 50.

Running the cell.

That's what it is. Connecting with the runtime

as a first step, I want to find out, count how many number

of elements are there,

number of values, the list.

I'm taking a counter variable with initial values.

Zero, which indicates that no values in the beginning,

before the follow, right, the follow

for we in, yes.

So part the loop, it reads for a number of times where is

and is number of elements at each iteration.

I'm adding one increment two count.

What is the meaning of the center statement?

CN NT plus equals to one means CNT equal two CN NT plus one.

So in the first duration, what's the value one?

The second duration one plus one. Two ation. Two plus one.

Three. Fourth duration three plus one. Four 50.

Duration four plus one, five

out five iterations loop will be terminated.

So finally what you'll get, the number of elements

that will keep it as a comment, print

number of elements that is C.

Okay, now C, totally count value.

Same thing. By using the same follow, I want to find out

total, I mean some

that means total of all values.

So here I'm taking on variable

total X.

So total plus equals two B.

So at each iteration, each value I'm adding to that total.

So, so please try to evaluate this, how it works

before loop total value zero in the first

iteration, what is the first value?

Next values

10, 20, 30, 40, 50.

In the first value, first iteration, we value 10 10, added

to total zero plus total to zero plus 10,

10 in the second iteration, V is 20, 20 plus 10,

30 in 30, duration in V is 30,

30 plus 30, 60.

In the next ation, 40 plus 60, oh.

And next ation in the 50 duration, 50 plus hundred or 50.

Now five times five is completed. Now total is red.

Some of X are, some is

total

one zero count product.

You can find out average the average equal to total divided

by number of elements that is sealed.

Now check the values of this average

total one 50 count 5, 1 50 divided by count.

That is that, okay?

But uh, separate one, separate far loop for for count,

separate far loop for, uh, totally hard rate.

Try to keep both of the things in one far loop

far because that other here I'm starting from the scratch

count, equal to zero, totally equal to C.

Writing a fur loop for V in X.

At each situation, I'm adding one to the count.

I'm adding value to the total, total plus equals to.

So after of this slope, both ready, total ready, average,

right now I'm computing average, average equal to total Y

count print total,

which is intuitive variable print

count, which is in c entering print

average, which is average.

Now on this or

Count ready average, right?

So in the next session we see other different aggregations,

like a maximum value, minimum value

that are without using predefined function.

Okay, see you in the next session. Bye-bye.

Video: Finding maximum and minimum with for loop

Welcome to Python sessions.

In the last session we discussed about uh,

finding some count and average with Pablo.

Okay, few more aggregations.

We'll see how to find the maximum

and minimum values with the Pablo

maximum.

Let's consider some of the values.

Just check all these values, which is the maximum,

but 300 is a max.

So both at the time of finding total

or account initial value, we have slice with a zero.

Our maximum. You should not keep that zero. Okay?

That's why any one value

among this list I will keep as initial value.

Let's say, which is max is first value. I'm keeping here.

Just check this.

The value tab, the first value kept into the max.

Now writing A for V in X,

if that V is greater than max, then I'm replacing

max as we.

I'm assigning V to B max.

After competition of this loop, your max value is ready.

Now print the max in the next Excel.

Maximum value is 300.

How this logic works, right, Einstein?

Previously I have values, I will copy the values also here.

These

are essential values

and we said max equal to X off zero.

That is first value. Kept it here.

What is that first value here? 10.

In the iteration one, check the condition.

10. Greater than 10. So condition here.

False. We

Iteration for the V value.

V value is minus check the condition

V greater than max.

That means minus 10 rather than 10.

So this is also false. Nothing will happen. Ation.

Three, look at the third value. The third value is 30.

We value 30 if

we greater than max is our condition.

We value 30. Now max value already

then condition True.

Then what will happen Max equal to

that means latest value of the max 30

iteration four V value

and the fourth iteration V value is 300.

Give 300 rather than 30 condition.

So that max become that 300.

Now iteration five and the 50

ation V value 23 condition falls

23 greater than now.

Latest max is 300. Condition falls.

Nothing will happen still. Max is 300.

So this is the last ation. Ation number six is 57.

57 is greater than max. Max is 300.

Again, condition pulse. So still the max contains three.

So after accomplished of this low for six iterations,

finally you could able to find out the max value,

which is 300 on the

with this logic bot, same

I want to find out minimum, once again,

print the values of x.

I have totally six values here,

as usual, same technique.

I'm keeping past value into this X, into this minimum,

which is 10 here.

This time I'm reversing the condition

for V in X.

If V less than minimum

previously V greater than max,

you said now V less than minimum.

If it is less than the current minimum value,

then minimum is equal two.

Now, after this loop, your minimum value is right

mini check the output now.

So here what is expected,

minus 10 is the minimum of all these values.

So minus 10 is the minimum.

Now for maximum separate loop,

for minimum separate loop we have, I want

to write both the things into one single.

Okay? So the technique max equal

to mini equal to X of zero

by following this technique, both contents, same values,

just check max, contents done,

and also min content stand here.

Data, same value we can assign

to multiple variables in this time, right?

Hmm. Now both are in slides max and win.

Both are in slides with the first value writing A, our loop

for V in X if V greater than max,

the max equal to,

and one more separate condition I'm writing.

If a wheel less than minimum, then minimum is equal to

finally print things, print x, print, maximum

max print, minimum

A, both right now,

check the final output.

Got it. So maximum 300, minimum value minus.

So of course we have predefined functions without predefined

functions for how we are developing your

one required functions.

Okay? So meet you in the next session.

We see more on that follow loops. Thank you.

Video: For loop on different collections

Hello, welcome to Python Sessions.

In our last two sessions,

we see the practical pr practical part of uh,

finding aggregations, like some counter average,

maximum minimum with help.

Okay, so now let's apply the far loop on different types

of the collection data types.

We have different types of data, uh, data collection types,

double

dictionary and set.

Of course there is one more kind

of things fraud set in a separate concept.

We'll separate session we'll discuss about the fraud sets.

So already have seen in the work with uh, for a like a photo

with the list objects.

So suppose if there is a double, let's say info.

I'm creating a couple.

First one is id. Second one is name.

Next one is salary, gender,

and something department number.

No, totally. This uh, info

is having five elements through

I want to access each, each element

or we in in

print that.

So there are five elements so

that five iterations will happen at each iteration.

Each element will be taken in the sequence. Okay?

So for loop can be applied not only on the list,

it can be applied on top levels.

Now same thing we see one example with the set collection.

What is set is a collection of mini items.

I'm writing a set

backup five values here.

Even though if you enter duplicate values,

it won't accept it.

Example of 53 times I'm giving,

but check, let say this is set one,

I'm printing that set one.

So P actually three times here,

but it has taken only one time.

Whenever I try to access that set elements,

I cannot access them using index numbers

only the way by performing a loop.

I can access it after this for

as in set home

Print.

But yes,

so far loop can be applied on even setups.

Okay? Of course here data is not in order.

If we want data to be in order we

to apply the sorting functionality.

Okay? Later we discuss about the sorting.

So after now we have seen Far loop on the list.

Far loop on the top and far loop on the set.

There is one more type of collection,

which is dictionary, DICT.

The full name of this DI CT is dictionary.

Dictionary is a collection of key and value beds.

Key and value beds. Bare means key.

The first one is key, second one is value.

Let's create something

and taking bank accounts.

For example, 1 0 1 is your account number

and our balance amount is 60,000.

Other customer 1 0 2 is balance. 15.

One more customer 1 0 3.

Value on lack,

1 0 4, 50,000.

So here totally bank is having four customers. See it.

Confirm that.

Now through this loop I want

to display each account holder details.

I want to print each account holder details after this.

For K in bank here, bank is dictionary.

So dictionary has four page,

but at each ation of only key part is coming into the K

just to print only.

The key only

1 0 1, 1, 0 2, 1, 0, 3, 1 0 4 account numbers as key.

That is coming into this K. Okay?

But I want to access both account number

and also it's balance amount

K in bank print

account number K,

and balance amount.

So bank of that K.

If I pass key into the dictionary, I will get value.

Now check what happens. You'll get a four account details

like 1 0 1 account number.

The total, like

The total balance amount is available with each account.

That nothing we are coming.

Now I want to find out

total amount in the bank.

I want to find out total amount in the bank.

What is the total amount in the bank with the follow.

I want to find out this. Hmm, as usual, our technique

equal two

sum in.

She zero writing a loop for gain bank

here instead printing the Strat will calculate total plus

equals and the ation K value is 1 0 1.

I want to get this 60,000.

That's 60,000 I want to add to this total.

That's why total plus equal to bank of

that not total.

After compression of this loop, the total amount in the bank

that is ready, the total amount in the bank is

to like 60%.

Got it. So in this way you can apply

for loop on different collection data types, like list,

double set dictionary.

Right? Okay. We see more operations in the next class.

Thank you. Let's make that.

Video: Break, Continue and Pass

Welcome to pattern sessions.

In the last session, we have seen how

to apply the follow loop on

different types of the collections.

Okay, so now

there are some controlling statements.

Those are break,

continue and pause.

So what is break is

when break is applied

And that loop will be terminated.

A second example for this

or I in arrange range of tab

I before printed,

if I equals two six,

I'm applying break tab.

We now actually this loop supposed

to be executed for 10 times.

A range of 10 means zero to nine.

Zero to nine means 10 times.

But whenever you applied some break statement

because of some condition through

then remaining iterations will not be continued.

That means entire loop will be breaked out.

So just check it because of this what output you'll get.

I got up to 0 2 5.

So this is what break is

for example, these are our marks.

So if minimum two subjects failed,

like the,

this is what the requirement think

that past market is at 35.

So for subject pass, second,

subject pass third, subject fail.

Fourth, subject failed already two reached the fail.

Number of subjects two reached then no need

to continue the remaining.

Okay, so I want to break that through.

So here, let's say you have C

and T where have C and Ts per failed.

That is zero. Writing a loop for Yamin box

check at yam greater than 35,

greater than Oracle store 35 is true.

I'm giving one increment for the C

when now this is reached two

and I don't want to continue.

Okay, so finally

if you have C and T print, FC

and check what happens.

Loop only four times.

Okay, for more clarity,

we said greater than Oracle 35 than FCT is less than Oracle.

Recall less than 35. We need to give it less than 35.

And that is a fail. The fail count is plus

if a CD value is A two, once it reach two, then break the

that.

Yeah, check this.

Hundred, 220, 23.

Fourth God fail, then no God. Finally you are printing.

Printing the minimum criteria matter. That is true. Got it.

So it won't go for the ones that condition.

Two, once you break it, it won't go to the next level.

That is what the break and

second thing is continue.

Let's check with the respond. Next statement is continue.

Once continue is applied

current ations and next step will not be.

I mean next steps means next statements

will not be executed.

Let's look into this

or I in range of tab.

If I equal to five,

simply saying that continue.

That means only current iteration, current duration.

Next statements will not be executed, then goes

to next iteration.

The next iteration will be continue.

Here I'm printing.

So the output here, column missed

except number five remaining.

All of these because once the condition met

to the file, okay, continue.

So that next spot will not be executed.

It is going to the next generation.

So this is with the cutting.

Same thing pass.

If pass is applied,

nothing is done.

So generally we at what scenarios we'll use.

This is observe this loop or I a range of,

but if I equal to three,

I don't want to do anything, then I'm keeping pulse.

Yes. If condition falls,

then normally I wanted to do something.

This. If I equal to three,

if the condition is true, nothing to be done.

If condition falls, that means INZI use not three,

then only, only remain takes to be the spot to be executed.

This is what my expectation, except three.

So here again, the column missed

zero.

1, 2, 3, 0, 1, 2, 4. The three is not printed here.

This is about pass.

Okay, so this, all the things will be used uh,

regularly in our applications.

Looping statements are very important. Okay?

So all dimensions of the thought loop

and while loop we have seen, okay?

See you in the next session with a different topic.

Thank you very much.

Module 8: Strings and string Methods

Video: Strings Intro

Welcome to Python sessions.

So in the last class, uh, as part of the,

we discussed about break, continue pass and

before that, like uh, different styles of looping,

different aggregations, how to find

by using looping, we have discussed.

So in this session we are going

to discuss about to string handling.

So how to process the python strings, how

to create the python strings.

What are all the different string methods

these things will see here.

Uh, first how to create a string.

You can create the string,

you can cap the string values in between.

Actually what is a string? First of all,

string is a sequence of characters.

Okay? So here in Python, these strings can be kept in

between single chords or novel chords.

Okay, So let's check one, uh, practical demo.

Let's say as equal to, I'm saying hello word

and one more string.

I'm reading by using double codes also

and integrating the same.

Hello can use single ports

or you can use double codes.

Both result is same.

And you can apply some of the regular

aggregation functions like uh, LAN on the string.

So now it counts number of characters into this string.

Look into this. Uh, part like, uh,

you can apply operators on some string,

actually basically two operators allowed on the string plus

and multiplication.

So multiplication

or asterisk, uh, symbol, which is start for example,

between the string, if you apply the

plus operator, what happens?

Yes. One I'm saying

yes.

Two I'm sayings.

So both the strings, I'm going to conquer net

to con net different strings.

Plus is the operator here.

S equal to S one plus yes. Two.

Check the value of S.

Now both string values got cond here,

Okay?

So between numbers of pluses applied addition will happen

between plus between uh, strings.

If plus is applied, the string of conation will happen.

Similarly, you can apply the surplus in between uh,

A list objects also.

Okay, so depends on object types.

Where you are applying this plus depends on that.

Different functionality will happen.

So technically in object oriented programming,

so this feature is called operator overloading.

Okay? Once we step into the object warranty programming,

we'll discuss about how

to do your own operator power loading.

Okay, so here in between these strings, S one

and S two strings have been perfectly con coordinated.

But between them and I'm expanding a space, then we have

to say like this C SQL to S one plus

after that one space, I'm expecting again plus then S two,

then now print this S two value

print S.

Now between lan you'll find space

and also you can apply multiplication operator also

for the strings.

For example, hello is one string I'm giving? Yes. In. Hello?

If you say yes in to three,

the three times the hello will be repeated.

Check this. Hello. Hello.

So generally this kind of things is used

to construct any underline such kind of things.

So while you suppose if you are formatting a report out,

certain number of lines, like you want

to print some underline, you can use this technique other

here, print,

I'm using hyphen.

Hyphen is a single character multiplied,

multiplied 40, 40 times.

Now same underline is printed here 40 times.

Check the output. Got it.

You can use any symbol just string into string into

that any number that repeats that character,

that many number of times you said 40.

That's why 40 times the hyphen symbol is repeated

here, right?

Similarly print, you can use some star,

star into 40 or 50.

So 50 times the star symbol is repeated.

Got it. So in this way,

automatic operators also can be applied,

but not all the operators, only cent

multiplication operators are allowed.

Addition and multiplication operators are allowed

between the strings

and still a few more things are there relational

operators between the strings.

You can apply all relational operators like I mean

comparison operators between strings.

So this technique we generally use

to compare two different streaks.

So let's execute those things.

For example, S one.

So this is a different uh, cell now come to here. Yes.

One equal to hello

and yes two equal to hello.

Now both values are equal or not.

If I say S one equal to S two, it is showing false

because here small letter H here, uppercase H.

That's why definitely those are not equal.

You got false, okay?

For example, I'm saying that yes one lawyer,

both the things I'm com converting into lower case

after converting them into lower case,

I'm checking their quality.

Now I got but in this way the given two strings are equal

or not to check that the relational operators also can be

applied even on the strings.

You can apply in greater than less than such kind of a

higher layer comparison operators for example,

yes, one is equal to yeah, just two is equal

to B.

Now as for the sequence, a small B is bigger.

Now can you check like uh, yes, one greater than S two

condition falls reverse it, yes.

Two greater than yes.

One condition is

you can apply greater than, greater than

or equal to less than, less than or equal to anything.

You can check it, especially if there are date strings

to compare with the date.

Uh, like to compare date strings,

which which date is earlier, which date is uh, uh,

latter one to compare that also this will be used

for example, I'm taking date one.

Let's say 2022

and March March 25th.

This is one date and the date two

2022 still on the

March and 29.

So that means here, uh, uh, 25th is earliest date,

lateral latest 29.

Okay, so just to compare it, which is uh, lower value,

which is bigger value, it is showing

DT one greater than DT two.

So we did not run the previous sale, not on the same thing.

I got false because 25 is lesser than this 29.

Okay, so check with the less than D one

is less than dated.

D one is less than dated.

Now it is, got it.

In this way you can compare different strengths.

Relational operators, you can apply

the next we see index numbers

generally on the list it couples.

We are applying index numbers to read the values on strings.

Also, you can apply index numbers as you know,

index numbers, basically three types

like positive index numbers, negative index numbers

and negation of index.

So index numbers, arms,

strings in python, I have three types of index numbers.

What are those? Positive

and negative index and one more

negation of index.

Okay, if it is positive index numbers starts from

zero already.

We discussed this part as uh, at the time

of uh, list object.

So again, once again we are discussing the same

start from zero.

The two rates from left to right

and negative index,

it starts from minus one.

It rates from right to to left

negation of index.

This is uh, starting with uh, this tail symbol.

You can use zero. Okay,

but please starts from zero here also

from right to two

Here.

These are the three different

varieties of the index numbers.

Let's uh, start playing with the positive index,

which is zero, starting from zero from left to right.

Look at this word Computer.

Now string value is computer. I want to take post character.

That's all. So it

takes post character.

Similar last character I want

without using native index number

or a index, I want to say this one.

So in competitor, how many words are there?

Totally eight characters are there.

Okay, we now we need not to count it to,

if the index number is starting from zero,

last character index number is seven.

Okay? So for this without competition directly can do that.

Len of yes, minus one len

of ass well is 88 minus one means seven.

So yes, sub seven means the last character,

which is R will be printed.

So these are the, these two are about post minutes.

Similarly, AYA, I want to print this P.

The P is in fourth position, yes of

4, 4, 4.

Fourth one means index number three

because uh, positive index is starting from a zero

right Now let's look into the negative index.

Check the value. The value is not better here.

I want to take lost character, lost the first character.

So negative index, I'm applying by saying minus. Well

that is or similarly lost second character.

Want yourself minus two.

Okay, lost second character. Getting.

Now let's apply the negation of the index.

Negation of index. That is starting with the tool symbol.

Generally if you say a subzero, that is first character.

If you apply the negation symbol, tool symbol for it,

that is last first character.

It'll take it. Okay, last first character.

Similarly second character. Want your soft one?

Generally say if you say til the S soft one last,

the second one, it'll take it now just check it.

We got E. So these are the three types of index numbers

are supported by the Python.

Okay? So in list that you are applying slicing

just to create the subsets.

Subsets from the list.

Similarly, you can create subsets from the given string

by using the slicing.

So our next work will be here

slicing on strings.

So

slicing runs here.

Range of index numbers.

Range of index numbers.

Look into this one. Once again in our string

there is a word called computer.

I want to take first three characters. Yes.

So actually the format is like this.

Your string or list

spot index column.

So this is what the format,

so here the rule is start index is always included

and index is excluded.

For example, I want PUT puts from.

So what is the starting question number?

Uh, p starting question for that means index number three.

I'm giving a soft three column that is starting index

from the third index.

How many characters that you want of PUT?

There are three characters. Okay? Three plus three, six.

So three columns. Six means what?

Index numbers will be selected.

Index number three because that is included

as a starting index.

Number next four, next five. The sixes won't hit.

Six won't be taken. Now just to check it exactly, we get PUT

as part of this expression.

This is what a slicing

similarly faster three characters I want.

I'm saying start next

and index faster.

Three I one simply say three in the end.

Index plus start index. I want from the beginning.

That's why no need to say any start index.

If you say like this, if start index is missed, is skipped,

you'll get from the zero.

That means the faster three characters you are getting.

Similarly, the last three characters, I want

look into the SA or else like last five characters.

I want last five characters are here.

Put I want to get the put now.

Yes, saw last file.

How to get the last two 50 character minus five call.

This time I'm skipping y index.

If I skip an index, it takes still last.

Okay, so all these are the basic operations on the string.

In this session we have L.

So in the next session we start using different

predefined methods on the strings.

Later we'll be learning in the functions concept of how

to create our own functions for anything.

Okay? So for now, like we are going

to use a predefined functions on the string.

We have for different purposes,

different methods are available.

So generally independently,

that is a function when now this is a part

of an object in object oriented programming.

The same functions are called methods.

Yeah, what the strengths? We have different methods, okay?

Which have some predefined functionality.

So those things will,

we are going to learn in the next session.

Thank you very much. See you next time. Bye-bye.

Video: String Methods

Welcome to Penman Sessions.

In the last session we discussed about how

to create the strings and how

to apply operators on the strings

and uh, relational operators on the strings index numbers

and slicing we have seen

and as part of the index numbers, we have seen positive,

negative, and negative of the indexes.

Now in this session we are going to work

with the string methods, especially here, find an index

strip, yellow strip

or strip under case related functions,

lower, upper and tight.

So these things will see,

let's say a string, the title part, our session

string methods spot one,

I'm grading a string.

Now

let's say the line equal

python is simple and is,

so this is a line.

Line is having like a lot of uh, different words.

So one of the sub I will take, for example, I'm taking easy.

I want to find out whether easy is available in

the line or not.

For this, you have an operator

that is called membership operator.

You can say easy in line.

The word easy is available in the given string.

It returns to otherwise it returns false. Okay?

Same thing. Something programming the word programming,

but programming what is not available here.

That's why it s false.

So already as part of the membership operators,

we discussed in not in, let's apply the not in also here

for the programming, not in life,

actually programming not available.

That's why it returns because of not in operator.

So in operator

what it can explain is whether the word available

or not, it can explain it,

but if word available in which position

that is available, it cannot serve.

I want know the position of it.

So then you have to use find or index.

So I will tell you, once I demonstrated these two things,

I will explain what is the difference

between fine and index.

For example, I'm going to check the word easy, okay,

what is our object Do index

or I'm checking with the, the word easy.

Check this. Now the word easy is available.

That is available at 22 index. That means 23rd character.

The starting version of easy is 23rd character.

That means index number 22

because star index is starting from the zero.

Same thing I'm applying with the find dollars,

find dollars, line do five.

Oh please, same output.

You will see Now here in both the cases,

you don't see any difference here.

Now I'm going to show you the difference line index

of for example apple.

The word apple, the word apple is not

available in the given line.

So for the index it returns minus one.

So I'm sorry, you got same thing I'm applying

with the fine.

The line do fine

kind of happen.

Actually apple not available. So that absorb here

the find is return minus one.

So here it can understand the difference.

This is a comment of the find find method.

If given substream

available, it returns

index to index extent

if not available

rate

returns minus one.

So same kind of statement for the index

you've given substream

available with

Start index number same as five

if not available it it throws at

by using exception handling concept, we need to handle that.

Okay?

So programmatically you can operate like this.

Suppose if apple

I'm applying line do kind apple,

the result is minus one, then print

word apple, not a value,

I'm printing something.

A word apple.

It's available at start index.

Find find of that apple.

Now check the message. Actually apple not available.

It is displaying that the word apple not

available in the given stream.

For example, I'm checking the same thing with easy.

The word easy

it is showing that, sorry,

it is a start index of it again.

Once again here you fix it did not change here. Yes.

Now look into this one actually easy available.

That's why it is showing the word happened is available at

start index 22.

In this way conditionally you can operate

by using fine function.

This is not possible by the index function.

So what are the next functions?

We'll see I want

to remove spaces, but these are

how three options, three functions here.

Strip to l, strip or strip.

So firstly work with the L strip

what each function is doing.

Post select right L strip

removes remove left to side spaces

including invisible characters.

Invisible characters for example like a backslash

and backslash to such kind of things.

Backslash and backslash T such things are

called invisible characters.

Okay? Similarly

or stream removes right side

both spaces and invisible characters and a strip.

This removes both sides.

So for this, try to understand the difference.

I'm taking my line. Let's say

Barak three

In both spaces are there,

Okay?

Left side spaces are the right side spaces.

Also there I'm applying line dot, yellow street.

Check the output. It cuts off

only left side spaces.

Still right side spaces are available.

Apply the R strip now line dot r strip.

Now it cuts only right side spaces still left

side spaces are available.

Now I'm applying strip which can cut both left

and right side spaces.

Got it. This is what the strip functionality at the time

of reading data from files.

Very frequently we use these three types of functions

like L stream, L strip, or strip and strip.

Okay, let's check the next function. That is a ch.

That is about changing the cases like a lawyer case,

upper case and there is a title.

So generally the standard for the title is if you want

to apply any title in the document in a word,

the first letter could be in upper case.

Okay, remaining letter to be in the lawyer case.

Let's test with this lawyer. Upper and title

lawyer

words, upper case letters

into lawyer case.

Let's see, the string is hello,

H-E-L-L-W.

I'm typing in a mixer cases. Some letters are in upper case.

Some letters are in lower case.

Lets me keep this uh, H to be in the small other here.

H is in lower case E, upper case second yellow,

lower case second L, upper case w lower case.

I'm saying that yes, not lawyer.

All the things will be converted into lower

case check.

All letters are into lower case.

Similarly upper case, yes, upper,

even lower case characters will be

converted into upper case.

Now I all are in lawyer case.

I want to make it as a title yes, title

here, only first characterize in upper case remaining all

letters, sorry, lower case for example, one more word.

I'm taking let's say line equal to

keeping some in the middle.

Also I'm keeping some upper case letters,

which

there are mixer case letters here.

No maam. Now I'm going to apply the title.

What happens, just second line.

Do title

check for each word.

Actually in the there are two words for each word.

The first letter is in upper case remaining all letters in

layer case, even in the second word serum also first,

first letter is an upper case remaining all in lower case.

So this kind of thing will happen with the titling function.

Okay. As part of this session, we have learned about

majorly eight functions.

Find an index, strict L strip or strip

and lower upper title.

Okay, in the next session we are going to discuss about

different other, other functions like uh,

ease visit is alpha, is alpha numeric.

Okay, ease space

and starts with an end with join split more functions.

We'll see in the next session. Thank you very much.

Let's meet in the next session. Bye-bye.

Video: String Methods part 2

Hello, welcome Python.

So in last session we discussed about some string methods

like uh, find index stripping,

strip with the strip, L strip or strip.

Okay, so in this session,

few more important methods we'll be

discussing as we'll.

Say about ease visit and dissolve.

For example, I'm taking string like this in this,

I'm keeping all the purely numbers, just

what it is connecting in front.

Now I'm testing with the X dot ease visit,

not run it true.

For example, I'm saying that ease alpha,

that means alphabets if the given string

is alphabet or not is testing.

Let's change the values. I'm saying A, B, C,

1, 2, 3, 4, 5, J, B, C.

The both numbers and characters are there.

Now yes, is time testing false? That means it returns true

all cards,

all numbers here.

Okay, now let's remove this.

Okay, before that one more test with the ease alpha.

So ease alpha is a false.

Even the alphabets are there,

but all characters are not alphabets.

So that means when it returns true returns true.

If all pairs outlets are alphabets,

Okay, let's change it with other parts.

And I'm saying Y this

time purely characters.

I'm giving Ys

written

one second.

I'm changing the values on a, B,

C one is a mix of numbers

and also alphabets.

Now in this case extort, he's all nu.

He's all nu means he's all for numeric.

That means it is a combination of

with lum cell characters.

Then it, okay,

so on that way you can verify it.

So he's all nu just you have check right now

and one more function is there

and the given string is that space or not.

If space available, it returns true.

If space not available, it returns false. Okay,

let's check one thing.

I'm saying X equal to balance string.

Now there is no space in the

given string do e space.

It has written me false. Try to use some space there

at least one space.

Now that's the same function E space.

Now it is false.

So still it is showing false

because wait,

that means all together spaces.

Let's test it. Multiple spaces I have here

is space.

Now check the final output.

That means it is ing through

if all gaps, all

pass spaces.

Okay,

let's check the next functions.

So it is all for numeric and use space. We have check rate.

The next functions starts with a specific character

and ends with a specific uh, character.

I'm taking a an example.

Let's say string is employed, right?

I'm checking whether it is starting with EMP and starts with

now.

Finally starting with the letter E, m, P, the set

of Charact, CMP.

That's why it

similarly, whether it is ending with the data

or not, I want to check yes start.

That's the data

other here I'm taking sales data.

Now we start starts with, yeah,

got it.

Okay. Now I have some in a list based on this some uh,

we play something with the list.

So thing that in the list, my file names up there like uh,

the employee dxt, the employee one dxt

that sales do dxt

and sales dot CSV

sales two

list of different files 30

and the products.in different files.

Now here file of Python of I want to filter only LY files.

If it is LY file, have a clue.

The file is starting with EMP. Okay, only LY files.

I'm going to get it now files.

So in Python list they have a comprehension technique.

A format is like this expression or loop in right side.

So This is a format.

On this format we'll apply this EMP files.

Let say here my loop is for fin files.

If it is file the clue that is starting with the starts,

it starts with our E.

So I teach it, the file name is coming into Y

and we are checking with the yahoo dot starts

with if this condition is true, it is ing that file

and only within the EMP files.

Only employer related files to be there. Now check it.

EMP files.

Got it. Only employee files are here.

Similarly, I want to take only sales related files.

Now yes, files equal to same kind of operation.

I'm going to apply on the files.

If you have done starts with starts with our sale.

Now check this. Yes, files

only sales files to be here.

Sales or txt sales. CSV sales. Two txt like this.

I want to take only text files, not the CSV files

and not remaining files.

This time I'm going with

yeah, in files, yeah, yeah.

DOT and suite because I want text files.

Each text file is ending with the do DxD

now into the dxt files

and verify the same thing with adjacent.

Now all these files are EMP, EMP one, dot XT sales,

sales two products.

All these things are text files

in personal scenarios.

Whether it is starting or ending with this one

or not, you can understand same thing.

We are going to separate Cs P files. That's says C files.

You are ending with do Cs.

Now print C files while you see,

right?

These are the examples. Still check other functions.

Now we have seen starts with an end split come

to the giant and split the next level.

These two are also very useful functions.

First I show you split

and then I show you the

giant split.

It splits the text based

on given delimiter.

So

further split method.

Default is a default to deter is space.

That can be single space

or that can be sequence of multiple spaces.

Default to deter is space.

Space or spaces. So let's check this.

I'm taking one employee record.

The record is like this one, one.

His name is Ravi salary, 80,000.

He's a male department level.

And from after here, if you treat this

as an employee regard, there are many entities like id name.

These are the fields, field values.

Each field value is separated with car.

Each field value is separated with comm.

How many commas are here?

Just counted

1, 2, 3, 4, 5.

There are five commas. Five commas means five plus one.

Six fields. If there are end commas, 10 plus one fields.

So here field to field is comma based on the comma.

I want to split them whenever I apply the split.

Split returns list.

Split returns list object.

So here five commas means in the list

you'll get six elements.

Now just observe record.

Let's say the employee list equal two record

dot split splitting based on comm.

Now check E list. Update is a list.

Now it has six elements up in six values.

Got it? Now I want to take the salsal equal to now.

E list of two.

Third one, the two I'm converting that into in

this was split.

So on the pure text world that also you can apply it.

Suppose I have a plain text like this line is equal to

python supports all

programming paradises.

So this is what my text now there are multiple words.

I want to split this line into the words. So word to word.

What is the separatory here? Space is the separator.

So line dots simply say split.

That's a word. Sequel to line dot split.

I'm not giving any delimiter here.

So what is the default limiter?

Default deli space.

Now check the words. What is the words?

Data type is list within that.

Each element is a word python,

one word supports all programming paradigms.

All these things that is there.

This is about the splitting.

Now there is another function, giant function is said.

The purpose of this giant to con coordinate

list elements as a single enemy to con

list elements into, into

single strength.

Let's look into the words, which is about what this, about

what I want to recon what into like string line.

Look at the values of the words once again.

Now each word to be separated with

format is like this deter.

Join off the list for this format.

Okay, now line equals to I want like let's say iPhone

as a deter between word to word as a separat race.

iPhone dot jar. Join off works.

Now just check the line.

But it Python.

iPhone supports iPhone. All iPhone programming.

iPhone paradises like this. I don't want this one.

I just want to say like this line equal to space

as a separator that to single space as the separator

dot join words.

I'll take the value of the line. Now this one.

Okay, now

after this one example I'm taking,

I'm taking a numerical content, which is a hundred,

200, 300, 500.

There are four values I'm trying to join

with these things based on some comma that

as a single string line, I want to convert value

to value the is comma to, pardon me?

I want to form it as a string. Don't join off.

Yes, you will get yeah that

because we now we want

to coordinate string elements means each

element should be a string.

Okay? But here each element of the list is number.

The reason join needs

each element the list

as string, string data.

Now for this, let's come out each value into the

string, the still, or of V

or V in X.

So X has has four values at each values coming into V,

you're applying string function.

For that V, the numeric data will be converted

into string data plan.

Now look at the latest values of the X.

Now each value is a string log.

Now I'm going to form it as a line.

I want ter between each value dot join the X.

Now we don't get any error. Now check the value line.

You got it? Okay.

So ham comm 200 copies center thing is as a single string.

Understood. So this is

what like a string methods were here.

So in the next uh session, we'll discuss more advanced tasks

like a sorting, sorting string already join.

You have seen for this sorter

content, how to apply the join.

Okay? And a string formatting such kind of things.

We'll be landing. Okay? Let's go to the next session.

Thank you very much. Bye-bye.

Video: String Methods part 3

Hello, welcome to Python sessions.

So today we see different tasks based on the strings,

especially sorting like, uh,

displaying data in a formatted way.

Uh, for example, at the signup of uh,

signing up off into the websites, Hmm, we are, uh, giving

password and confirm the password, how to validate that,

that kind of operations.

We'll see The first, let's work on the sorting.

So for this here, the sorted is a function.

So the sorted function can be applied on the numerical data

and also on the data.

For example, I'm taking one string

here as SQL to bar.

I want to solve this sort off.

Yes, it sort the data in a selling order,

but the list of characters, it is given

just the output of sorted.

So is a small letter.

So from small case to from small letter to two.

Big letter in the alphabetically, yes. Sending order.

So yay, A-B-H-R-T like that it has arranged to the data.

So the written data type is listed on the

teacher element character.

I want to ate all these things as a single.

So already has in the previous session has in the giant.

Now I don't want any dealing adjuster.

I want to con coordinate all the independent characters.

That's why it's a blank space.

I have kept, I did not give even space a blank space.

I have join off

that list, sort it off.

Yes. Now check the output of it.

A, A, B, H. That is as a single.

Same thing if you wanted to do into the river order.

That means a descending order.

Then I have to say sort, yes,

sequel, but default.

Sequel to false. False means descending order.

If it is reverse is two that is in descending order.

Check the output of it.

Now D came first. D-R-H-B-E-A.

Okay, so in descending

or alphabetically in descending order,

it has given the list of characters.

I want to con coordinate that dot

join off sort it sorted off

as but reverse equal to.

So this is our uh, first option.

Next I'm going to display the data in the format.

Okay? Format of those strings is like this.

Formatting of the display,

formatting display output.

I'm taking on variable name.

That means let's say

H I'm going to print it.

So generally without formatting, the style is like this.

My name is after that comma, then name

after that, I am

after that, yes, value I want,

but again, like this, you have to print it.

So this is uh, like when our string is upgrade,

you are keeping the string into the coats after that comma

and after that variable, after that comma again,

and then string again like this, it is really tough.

Now the formatted way, the same thing I want

to print in a formatted way.

Print. Yeah, the formatting of the strings

the coats have started with.

Yeah. So within this here, my name is,

I want name value here.

Whenever I want name, value, variable value, keep, uh,

keep the value into, keep the variable into ly brackets.

After that dot I want, and

after that I, um, immediately is value.

I want H yes,

word like this.

Now usually you can format the expected output.

So here, output. My name is ra, I'm 47 years word. Got it.

So this is what like a formatting of the strengths.

Next chapter, one more task here. Confirming the password.

So you have set the time off,

sign up into the any application

and website of web applications you are giving your

user ID and password.

So password. So first time you give one password immediately

it'll ask you to confirm your password,

but it'll accept that.

Otherwise it'll say yeah,

that means the password not matched.

I'm going to give that one. The second task here,

password equal, that's a key two.

Success is my password

and I'm confirming that password.

Key two, success.

Now both the things you just check it.

Password is equal to confirm if both

values are equal to otherwise false, it returns false means.

So in the password, the KK small letter,

but in the confirm the K is upper case letter.

Now try to give the correct things

I'm running in the next cell.

The first letter I'm keeping in capital letter,

upper case letter, the second case, also upper case letters.

Now check both the things.

Now it is okay, how already have seen equality operator

in the strengths in our, in our previous sessions.

Okay, just we are performing the task.

Now we into the next task.

Credit card details.

So of course you have your own credit card details,

but credit card, uh, details to not to be displayed, uh,

completed to the public at the time you are going

to mask it, okay?

Generally your credit card contains is 16 numbers.

So at the time of masking,

accept last four characters remaining,

all the things will be mask.

So how to do that, that operation,

I'm giving a value here.

That's a card number.

I'm giving the card number.

5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 60

total.

Just confirm the level but uh, characters here.

I have 16, but I want

to mask this except last four characters revenue,

all the things I want to mask.

The masking format is like this.

Mos equal to xx iPhone

after that again four iPhone X, iPhone

and I want to display only last four characters of the fix.

This is the mask of the country.

Now card equal to

plus time giving mask plus

between those strings pluses applied

string conation will have.

Okay? Hmm.

Last four characters

of the card number I want to take already.

You know the string supports index numbers,

negative index numbers and also slicing

by taking a negative slicing.

We use this one last four characters. I'm going to pitch it.

Card number minus four

column I have given the sorting number.

Minus four indicates last four character,

last four character here to I did not give any index

or until last it'll take it.

Okay, now just display the content of the card.

Now, now mask wear.

It is showing you the output. Got it.

So this is what like, uh, as part

of this session, our learning.

So mainly the summary,

what you have learned in this session.

One is, uh, like a sorting in a

sending order, descending order.

We have seen in our practical part and displaying the text

and along with the variable values in a formatted way.

After that, confirming password depends,

verifying both strings are equal different.

And finally, how to mask your core details.

So these examples we highlighted in the next example,

in the next session, we see more on

different tasks like a domain name and email violation.

Okay? And, uh,

converting string into the pal drums, pal drums.

Pal drums means, uh, if you reverse

that string still same value that is called uh, pal room.

How to create your own pal Drums, how

to check whether the given strings are a pal, not how to do

that will be see and uh, in a day, in a date.

String value. How

to extract the required things like a month, year and a day.

And gram strings are there.

Anagram strings are, uh, mostly used in my,

in our NLP things natural language process.

If you observe this, uh, listen silent.

These are the two separate words,

but, uh, what are the with what characters that listen,

the word listen is formed

with the same characters without using any extra character.

That by using same set of the characters.

Another string is formed called silent.

Such kind of words are called anagram strings.

Okay? Whether given string is anagram

or not, like, uh, we need to check it.

How to do that check will will be and rearranging the cases.

Rearranging the cases means like, uh,

lower letters into upper case, upper layer,

upper letters into layer case.

So that kind of activity is called rearranging the case.

And in a given, uh, uh, comment

or text, there are some punctuation characters out there,

especially looking to the first line, like a text.

Hello Bar, how are you?

So after how there is a comment bar,

there is some thing explanatory.

Mark and opera, how are you? There is a question mark.

I want to remove those punctuation related characters.

I want to form a, a new text from that. Okay?

So such operations we will, we are going

to learn into the next session.

Okay? See you there. Bye-bye. Thank you.

Video: String Methods part 4

Oh, welcome to Python sessions.

In our last sessions based on the thingss,

we performed some tasks.

So a few more tasks we'll see in this session.

Look into the first example here.

There is some email I, okay, from this email idea, I want

to extract the domain, okay?

Think that the domain is with organization

and then.com, okay?

Uh, for example here, organized name is example example.com.

I'm saying, okay, I want to extract the content from

let's say email equal.

I'm giving@therightexample.com.

So I want to separate this domain,

then domain equal to first.

You split it step by faster by step

splitting into the words list of words, email, dot split

split off splitting based on at the red symbol.

So before at the red there is a name after at the red.

That is now how many at the red symbols are here? Only one.

That's why it is splitted into two words, two elements.

Now check the values of this. Uh, email that words

and domain.

The required thing is, domain is in second.

One domain equals

words of one or minus one.

Now look at the domain.

Similarly, from this domain, I want to take, I want

to extract only the organization.

Now, once again, split it.

SAW is equal to domain dot split

this time splitting based on dot.

Now look at the value of the dot.

How many dots are there in the given text? Only one dot.

That's why. Dot left side one word

and dot right side one word.

Now example is organization name.com. Okay?

That is website extension.

Now I want to take the organization name, watch equal to WC.

Now look at the value of watch

you are getting organization name right?

Come to the next task.

Converting string into pal drums.

Okay, so that means the pal drum generation,

what exactly the meaning of pal room is.

For example, I have a content of there is

reverse this next

V, yay.

K-A-T-A-K-I-B.

Okay, so in this is okay,

but in English, this is not perfect.

Let's say YZ Aja, if you reverse this,

type in a post the right to left postier Z

and got it.

So this is what like uh, in four, same in river sort.

Also same value things are called

just for improvement.

These kind of examples that used, for example,

my string is yes, yes is A, B, C.

I'm going to add something. S soft

minus one, just check what happens.

C, B, M, right? Got it.

ABC is a, ABC is in power order. CV is in order here.

What exactly here, double minus one is doing is jumping.

So jumping in the river, product positive is taking minus

one means a C minus one.

One more minus one is added. That is becoming minus two.

Pop that minus two. One more minus one is added.

That is so minus one, minus two, minus three.

Then it is taking C, B, A, like this. Got it?

So I'm going to Concord.

This pal is

yes plus yes off I call minus one.

You'll get a pal stringer bring

that value in both orders.

You just see A, B, C, C, B,

a reverse reverse it also A, C, C, B, A.

Okay? Now this A, C, CB is a.

Now any value, the given value is

or not, let's check it how to get that in a reverse order.

For example, the given reverse

or let's say a sequel to

I'm taking Azure.

Reverse string Yes of

column, column minus.

Now check if yes equals to S

print given two

strings given string in a formatted way.

We'll write it already has in the formatted way,

given string, yes is eh,

for example, I work condition falls.

There is not pal, right?

Four given string.

It is not given string. Yes.

That value I'm keeping

is not,

yeah, the variable I gave.

So given string is actually his idea.

River, it still the values is his idea.

It both forward string and river string.

Both values are same.

We are printing, you're deciding whether

it is a pal or or not.

Now is a pal.

Let's test with now reverse.

There is not palant.

So this is what our next task here

and

more task will perform.

Look into this one. There is a date string. Okay?

17th or zero four means April. So first there is a string.

The string has the three date components like a day,

iPhone month, iPhone eight.

From that required things I want to extract, I want

to separate the date components.

That second date

in yy.

Yeah ma'am did the order I'm going

to take suppose now this is 2025.

After that month zero four after that date.

Today's date is uh, let's say 17th.

Okay? This is the date I want to, from this date string.

I want to separate this year path month part

and dev part that two, I want them as a numeric data.

I want to extract the numbers from that. Okay?

So in this case, just to split it,

split splitting based on this time iPhone,

whenever I split it, there are two iPhones.

You'll get a list. List contains three values now,

but each value is a three.

Just confirm this,

okay?

Now I want to extract the year, year equal to

W zero.

That is the first one that you do.

Similarly, I want to extract month

ymo and equal to in W.

There is a second one after day. I want a day equal to in W.

Last one so that you can say minus one.

Now all required things are support, uh, separated.

Now prints year,

month

required.

Things are separated. Got it.

So this is what our next task

look into ana anagram strings.

Listen, what characters are involved here?

IE yes, T, L, l.

Same thing with the same character set.

The second characters are the second string is also now,

but what billings are same.

So now if two words formula by same set of characters,

that is called anagram anagram string.

Now given string is anagram or not, I want to check it.

Let's keep something like, uh, I'm taking user input.

Let's say S-T-R-S-T-R is a is a reward

input type your strip.

I'm keeping it to yes, what other

characters has typed always better practice.

There might be spaces I'm removing the spaces

of both left and right side.

In our previous examples, we have seen strip function,

yellow strip power strip strict.

Okay? And

after that I'm converting them into lawyer like this.

Everything is a string. Now, so this is string.

I'm converting. So like, uh, I want

to sort this one in a sending order.

Okay, let's say

this is string one doing the same thing.

Type your string one

and S two input off

type your second string,

same operation I'm doing here string and then lower.

Okay, so let's say yes,

four one, sort it out a room.

I sorted one.

I'm going to sort the first one simply the function sorted,

sorted off as one in ascending order sorted.

Two here, sorted off

two here.

I did not say any reverse option.

So that default is ascending order. Okay?

And then my next work I if

sorted one equals

to two both values then

given string is anagram

given strengths or anagrams.

Yes, print

given strings are

not grams.

Now let's run this. It is asking you to type the string one.

I'm typing the string one fast forward is a listen.

Then next word, what is the next word? Listen.

And silent. Silent.

So what is the confirmation message given sir Anana?

Okay, let's run one more thing.

Lesson.

Second word is lesson.

These two words are not anana, okay?

Just check. What is this? Uh, yes.

One yes, one is here.

Yes, two is less. Let's keep S two.

I'm modifying this S two, S two as silent.

Okay? Yes. One is LS two is silent.

Now both sorted apply, let's say sorted one equal

to sorted off.

S one similarly sorted to equal

to sorted off as two.

Now check both values. What you, what you get, you can see

print sorted one and print sorted.

All characters are ing has sending order.

E, i, I and ST. Same thing for the second house.

That's why when you compare it, it says that both are equal.

Now sorted one equals to sorted because of true.

Previously we got output as another.

If the values are different, you'll get not another. Got it.

This is what the previous task bot we did.

And one more thing. Arranging the cases, rear the cases

here, there is a special function in Python for the strings.

That is swap, swap case already.

Case related functions we have seen like a lawyer,

upper title, such kind of functions.

What lawyer is doing, converting all into lawyer case.

Upper is connoting all into upper case title.

First letter into upper case remaining all into lower case.

But the swap case is different.

Lawyer will be connoted as upper,

upper will be connoted as lawyer.

Okay? That is rearranging the task.

Let's do the practical part.

I'm saying s equal to hello python.

Now here HP or in upper case remaining all in lower case.

Now I want to reverse that.

Simply can say yes dot swap case.

Now check the output, capital hedge will become small hedge,

smaller hedge remaining, all, all into layer.

Now those into upper, this is called swap case.

Let see the final task of this session.

I want to remove punctuations.

So to do these punctuations, we have a, a special

model here called string.

The string related special functions are used here. Okay?

First time importing the module import string.

Let's run it now string is imported.

Let's say now, string dot punctuations

check this what all the different punctuation

characters that we can see.

So these are the different punctuation characters,

which are special characters.

So, but in the given text there can

be a punctuation characters.

I want to remove those punctuation characters.

That is what the task, okay?

This here, there is a text. Hello. Ha, how are you?

So what are the punctuation characters here?

There is a comma, there is a explanatory mark.

And also there is a question. Got it?

There is a punctuation characters.

I want to remove that one. I want, uh,

my expected output is like a play plain text.

Like hello Barath, Hawaii. Got it. So observe this.

We do suppose a comment line is like this

line equal to hello

or, or

this is a thing.

Got it? Now I want to check whether this, uh,

whether the given charact is, uh,

whether given characteristics part of

that punctuation or not.

Just check it. I'm saying comma, whether this comma in

string dot punctuation

comma is a part of punctuation.

If yes, that true, for example, I'm keeping h,

h is not a punctuational character is alphabet.

It returns false. This technique is enough for us.

I'm performing a loop. Left side

expression, right side for loop.

The pan loop is like this for C in line.

Each character I'm reading here, just check

what happens here in the place of, uh, expression.

I'm keeping that C. Now I got a list.

Each characterizes separate element. Okay?

So, but I don't want all the things.

If the word, if the character is not

punctuation, then only I want right?

Said I'm giving if condition. What is that?

If condition, if that C not in

string punctuations,

uh, in the first, in the first, the C value is h,

h is not available in punctuations.

So that this condition is true.

If this entire condition is true,

then only it returns that value.

Now check it here. The sixth element is actually comma.

Okay? Now the punctuation will be eliminated.

Now karma got eliminated.

After explanatory symbol will be eliminated

and after how are you?

Question mark will be eliminated. Eliminated.

All the things have been eliminated.

So this entire thing I want to con coordinate

with a blank ter dot.

Join all this list.

Now, no pun,

look into this one, no pun.

Now the given text is this one

where we have typed, hello, how are you?

Within this punctuation, characters are there,

but in the final output we eliminated those punctuations.

Hello ra, how are you?

All these other different styles

of hosting string operations in our upcoming uh, lessons.

We are going to use all these different techniques in our

different applications and different application programs.

Okay, let's meet in the next

session for the next lower topic.

Thank you very much. Bye.

Module 9: List and Tuple Operations

Video: List and Tuple Introduction

I am going to talk about collection data types.

Now

the first one is a list,

the collection of items.

Second is the double.

Your item means value

is a collection of items.

Both are holding or holding multiple values.

But here, purpose.

Collection of homo engineer items.

Yeah, reverse it.

Now.

It was your items. That is a basic difference.

Observe here to make understand about these two points

and taking your qualification.

Keeping this as a list. Your first qualification, BTech.

After that, we did mtech

after that.

PH.

How many values are here? Three values. What is the first?

First one? Purpose of first one.

That is your qualification name. What about second?

Say third One. Same purpose of each element. Same or not?

Such Kind of same kind. Not same type.

Same kind of values are called homogeneous items.

Such kind of behavior is called homogeneous behavior.

That means purpose of each element

means you can say

saying kind of values,

but technically whatever the data types you have given list

will accept it, but we don't use

that now.

Functional differences possible. Observe later.

Technical differences. We'll see.

I'm going going to talk about heterogeneous.

Purpose of each element

is different.

That means a dissimilar kind of previously same kind,

dissimilar kind.

I'm taking info, changing the symbol.

For example. First one is your name.

Think that your name is Ravi.

25 is your age.

Is our location.

Yes. Yes, he is our designation.

Senior software engineer.

How many elements in this collection? What is first one?

Second element is talking about h.

Thought is location. Both.

Now what is this behavior?

It was in his behavior. Understood.

Now this is functional difference.

Try to understand the technical differences.

Video: List and tuple Operations

The list, technically list is

mutable opposite.

Top is immutable.

If it is mutable, they allowed operations.

Number one,

read second one.

Append. What is it you can append values

that is already have like the three elements.

Fourth element, you can add it. Okay, that is upend.

I can update also.

Last year, your age 25. Now what is your age?

26. I want to update the world value to the 26.

I can last month.

Your salary 90,000 This month. Your salary? 30,000.

I can update that. Okay.

Suppose when apply got resigned or we terminated.

I want to delete that. Delete.

I mean all these operations are possible too.

List. List is a mutable.

Mutable means which allows all these poor operations.

Okay? Now you mutable

allowed operations read only.

The word only means what? It cannot allow other operations.

That means

not possible

update, not possible.

Delete, not possible.

Okay, so these operations are not possible.

Only allowed operations are the read.

All these are technical differences.

Come answer. So

to make your understanding, I come to one scenario.

I'm taking your friends.

What data type do you recommend? Suppose

Gary.

Siri. These three are our current friends. E Tomorrow.

Can you add one more friend?

So that means happened a lot, right?

Tomorrow I got a fight with Gary.

You want to remove him from your friend circle. So deletes.

This is also allowed fundamental

everything aloud.

Actually his name is, uh, eth.

So allowed operation or not update? Allow.

So your friends please.

What at a time please.

Tomorrow, Mr. When you want to upend him up

to your friends list, we can upend one friend,

two friend, multiple friends.

You can upend if you want

to remove any friend, you can remove.

Similarly update by default.

Everyone allow is allowing reading operations.

All these are operations are called work.

All these kind of operations are what

database is good here.

Which update allows list is allowing,

I'm taking your family hierarchy.

Okay, my family hierarchy.

I'm my father

KY grandfather.

Samba. What is your

data type of each element.

String

command.

If it is a collection of strings,

what data type do you recommend?

Family hierarchy.

Definitely not a single value is a

collection list or couple.

First one is son. Second is not son. Second is father.

Third is anyone is not son. Anyone is not father.

If you want father, always we have to pick not winky.

Second one. Understand. Uh, who is my grandfather?

Samba How you are saying, how you are saying.

Third one, you are picking who is sending this story.

First one you are picking if you want further details.

Second one, you are picking position. Important or not?

Okay? Think that this is army. Left side is army.

Uh, anyone. So anyone soldier want to send today?

Can you, you can pick anyone

because anyone is a soldier you can send.

But come to this one. I'm asking send son to the war.

Not anyone, right? You have to pick pastor position.

Important. But here, position doesn't matter to you.

Anyone is a friend. Everyone is a

friend here in the left side.

Understand? So here first element is talking about what

some second element talking about.

Father thought about,

uh, and follow.

Homogeneous or heterogeneous. Heterogeneous, okay?

Reason number one, why it is top to be double. Hmm.

Okay, come to the left side. Can I add one more friend?

Yeah. Can I add one more Father? No

Can add Uhhuh.

This is my hierarchy.

Later my daughter will come into the picture.

My daughter will create her own hierarchy in

that my daughter's hierarchy.

First one is my daughter.

Second one is Barak, third one is Vanguard.

But this is my hierarchy. My story. Same thing.

Your story different, your son's story, different,

your wife's story different.

Understanding the point.

Uh, here, can I add one more father?

That means appending should not be done right.

Can I add one more grandfather? No. Understand the point.

So append, appending restricted, right?

Uh, tomorrow I got a fight with gri. Who is my friend?

Can I remove him? Yes.

Tomorrow I got a fight with my father. Can I remove him? No.

Even the fight happened always. He's my father.

Understood. So that means

that delete also not allowed, right?

Not allowed. Delete, not allowed. Understood.

Uh, what is one more thing pending? I don't like my father.

Can I update him? No.

Update is also not possible here. Understand.

So here, uh, what operation is allowed?

Finally only reading whether I read same information.

You read same information.

Anybody reads it, same information to be presented.

That's why that should be mutable behavior.

Immutable behavior is required here. Immutable behavior.

That means what? Operation. Allow read only. That's why.

What object data type I need to give.

That's a family hierarchy.

Uh, here each element data type is what data type,

even those things.

But dissimilar kind, right in the left side.

Homogeneous behavior in the right side.

Andogenous behavior, understood the point. Got clarity.

When to use the least. When to use stoppel.

Video: List Operations Lab

I'm going to talk about first, let's say creation.

Think that is a list

keeping some values total 10 values.

I'm, I'll just check a type of X.

You'll get this as a list. Reading from list object.

Basically how? Two ways.

Number one, using index numbers. Number two, using slicing.

Yeah. Index number. Python indexing starts from zero.

Number zero. That means if you want first

one or the first one.

Index number zero for second. One. One, one.

Similarly slicing.

The word slicing means the range of

index numbers, so what you want first to five elements

are last three elements.

At that time you will be applying ranges.

The ranges to be applied with the technical slicing.

I'm going to talk about index numbers, so here three types

of index numbers servicing numbers.

One positive index starts from

zero from left to right.

The second type negative index,

it starts from minus one,

river sorter from Y to two left.

The third category, negation of index starts from

zero from right to left.

What is the meaning of each one?

We try to understand, look into the your list first.

Totally. I have 10 elements here.

I want first one. What is the index number

from left to right means?

Sir, what? It will take 10 or a hundred.

10 will take 10.

Okay. This is to get faster.

Similarly, I want this for the fourth one.

What is the index number? The fourth one

I want, so three, so

that will get four.

That is 40. The same positive index style.

I want last one X off.

What is

10 yof?

10 means what? Yeah,

because counting starts from zero, zero yof len

of X minus one.

I'm saying what I will get,

I'll get hundred.

What is len of X here? Len of X is 10.

The X of 10 means what?

11th element X of 10 means 11th element, right?

Is there 11th element? No. So that error or not understood?

If I say 10 minus one, what is the result? Nine.

The of nine means 10. 10th element.

10th element is the last element in this case. 10 elements.

In the next case, there are 200 elements in the next case.

Again, 5,000 elements. It doesn't matter.

In the place of 10 hot coding, I'm fixing X.

How many elements are there? Minus one.

If there are 99 elements, what is the line of X value?

99 minus 1 98.

The of 98 means the last one or not? I will get the last.

This is a positive indexing style. Understand.

Now negative index starts from minus one,

minus one from direction right

to two left.

Hmm. Come on, apply. I want last. Last one.

I want last one off. Minus one.

Understood. Last third one.

That means I want the 80. Last third.

One minus three. DEXA minus three.

Got it. Now I'm going to apply negation. What is xal? Zero.

I'm applying negation symbol. The till symbol.

Reverse it last first of all.

Okay, for the positive, just reverse it.

Last first one you'll get. What is the last first one? Yes.

One similar. You tell me. I want third one.

Generally third one X off two.

This is positive style. Reverse direction.

I want X off til

this is last.

Understood. Three types of index. What are the three types?

Slicing in all example in these three ways,

you are getting only single value,

but I want multiple values.

I mean ranges and you have to sell like this. Slicing.

Slicing means range

of index.

Index. The format is like this.

Your list object start index, colon

and index list of start, index, colon and index.

Here the rule is like this.

Start index is always included

and end index

is excluded.

Suppose if I say zero column five,

what indexes will be selected?

Think that start is zero and is five.

What indexes will be selected? Zero included.

1, 2, 3, 4. What about five? Excluded.

Excluded, understood. Seven. Column minus one.

Tell me what is the meaning. I'm sorry.

From the eight onwards.

Correct Bill. Last one.

You can say last, but the accept last one.

That means up to last, but one, you'll get it.

Understand the point on that way. That means uh, seven.

Index number seven, what indexes will be selected?

Index seven after that.

8, 9, 1 be selected because nine is there. Last one, right?

Understood. Seven. Eight indexes will be preselected.

Once again, look at the data

and tell me based on this technique,

I want from third onwards coming four elements.

30, 40, 50, 60 from third onwards.

My task is from third onwards coming.

Four elements off on, on left side,

start on right side end.

Start index is what coming? Four element. So two plus four.

How much? Six. Six. Two plus four. Six.

That means what final indexes will be selected here?

2, 3, 4, 5, 2, 3, 4. Five means this. Four element.

Come and check the output. 30, 40, 50, 60.

Understood. Then append is the function.

Understand. Hmm. What is your least object? X. X.

Execute and check the value of the X.

Previously only 10 elements. Now 200 added or not.

Got it.

300, 400.

No wrong because the entire thing

will become a single value.

This single value will be up in the last,

so last element will be list object,

but I don't want like that.

Just check this after that. Print X. What is the last one?

For the last one, I don't want list.

I want to open two values.

300, 400, but the first I will remove this

or removing de.

Which one? Last one. I want to remove X of minus.

The last element is completely remote.

Mm, so using a pen,

if I set 300 comm 400, it is not allowing me

error, but already we have a technical,

the list plus a list, right?

The technic I can use above this

jack equal to y

plus 300.

Comma 400 after equal.

To look at the expression that to left side y,

what is the data type list?

Author plus what is the data type list?

Plus list conation of the list.

So final result will be restored into X.

Now, previously it has 11 elements, including that 200.

It has 11 elements. Now

how many elements will be there into X 13.

Just check 11

of XI have 13 elements.

No, just print all the values.

3300, 400 total. 30 values, sir.

Video: Tuple Operations Lab

So,

so totally.

How many elements are here? There are five elements.

So first one is here,

what is a schema?

Schema instruction. First one is Id second one.

Age, age. We

after that agenda, the last one, department numbers.

Immutable object.

That

that is what it does not supporting.

Right?

Update, but

indirect, but indirect.

Two as

what?

Two ways of list. Number one. Using index numbers.

Number two, slice.

I'm gonna talk about the index numbers.

Three types of what?

These are the three ways.

Once again, look into the structure.

There are five

elements I want.

This is,

this is your info

you get, so try

to apply the negative index.

Last one. Department number. I want number index.

Similarly in the style is third

one minus.

So is also supporting native index, right?

I'm gonna apply negation.

What is the meaning of it? I want one

in this.

I want

in positive style.

I want gender.

Gender is last. Second one.

Minus what?

The meaning of five Minus two. Three. Three.

Understood. Three means what?

In positive style. Three means what? Fourth one, right?

Fourth one is gender, not

the second one.

In the negative side. In four info

minus In negation.

Slide info. One. This is positive.

Apply the negation. So here

you'll get last second.

Last second. Put up all second.

This is not possible because that is indirect.

Try to do this. Four, two.

Which is third one which is salary.

I trying to four. You'll get

but at any cost, I wanna change it.

What is x? A tool. What is x? Apple?

I'm saying list of x list is a function

which can into list, understand

by example Y.

What is

is here.

The couple of files converting by taking help

of these two functions you can perform operations on.

What are these? Look into the info. I have five values.

I want to update this salary directly. Not possible.

That's why I'm converting this into list.

Equal list of now

what is the if

of, I want to make this as prox.

Now Sally got updated

but happened at level, not at

I'm recon.

Indirectly or not previously.

Nowadays directly. Not possible.

We're using indirectly to

same style

up.

This is also not possible

in

info

designation.

Senior software engineer. That is designation.

You'll get is there is

attribute in

I want indirectly

it valid or not.

Now, yes, we can that into

Previously five elements.

Now there should be six elements. Designation also added.

Understood this

same style.

I wanna remove

this

six element.

I wanna remove S.

So designation. So in this way by using indirect, you.

Module 10: Sets and Dictionary Operations

Video: Introduction to Sets and Dictionary

Suppose you did BTech in the year

2015 after that Mtech

in 2017,

after that PhD in 2020.

With this information, I want to create a object this time.

I go for DIC stands

for dictionary.

Dictionary is a collection of key and value pairs.

It is a collection of

e and value

pay means you know what is

per pay means what?

Yeah, among two. One is key.

Second one is

value, but there are some rules.

Key should be unique.

What are the meaning of unique here?

Not duplicates. Key should be unique.

Second

value can be

right.

Come on, try to try to make this as a dictionary. Sir.

What is key? What is value here?

That qualification name is key value that year passed.

That is value. Okay, come on, make it.

That's a qualification. Q code. What is the symbol for it?

Curly. Curly bracket. First one always

pay is left side one value.

That means left side.

One component, right side one component here.

Pay before left side key before colon key.

What is that?

What is the pass out here? That is as well. Next page.

Amtech. What is the

key next?

PHD 22.

Understand. Now I want to know when did you pass out Mtech?

You welcome. It is passed.

Value will be return.

I'm saying queue of mtech mtech value or not.

So what is the reply?

2000 of

What? I will get 2000

of BTech.

2015 Q of MBA.

What is that? Address? E.

Understand how you are reading values

of list index.

Number first one you want means zero.

Second one, you want one like that?

Double values how you are reading. Double values.

Index number, how you are reading dictionary Values.

Values by passing key. Understand.

By passing key you are getting values. What is set?

Set Collection of

unique items are values.

Unique means you know no duplicates. I'm saying yes.

Se equal. How many values I have total.

Nine. Nine.

Values Line of As? What is output?

Four. Why?

Actually the value is 64 times right,

but it takes only value.

Ten two times takes only one time

value contains one unique values.

Just check the value.

10, 20, 40, 50, and 60.

Video: Set and Dictionary Operations Lab

I said

3, 6, 3 or six, still three.

What is the rule? He should be right?

So what will be the output of this one?

What will be the output of this one? A.

What is the value? No, 80 B.

What is the value? What are the latest value?

What about C? What is Mike?

If I have given duplicated case, what is happening?

Always latest will be available with you.

Finally, how many pairs you have? Three pairs.

Confirm this,

still have this.

Three. What are the values of X? Just check it.

What values you have. Yeah, latest value, right?

For B, also as well,

I'm taking your qualifications.

Suppose your qualification level. BTech you passed in 2018.

One more qualification. We

have 2020.

One more qualification. We have 20.

Now look into qualification.

I have three page,

my phrase.

List how you are accessing elements

if it is listed

or couple how you are accessing elements

by using index numbers,

but for DI dictionary index numbers will not work out.

Key. How to pass key.

If keys passed, value will be returned.

Suppose I want to know where Mtech passed out.

Qualification of MT or not.

If yes, it will get a value. The value is 2020.

Same style.

What is that? 2022 pastor

qualification, PH PhD

available in the dictionary Army.

No. So that you will get a key error. Key error.

The error message. Suppose if such kind

of statement is applied in the between the program

at some statement some keys passed into that,

but that key not available in the dictionary,

error will be thrown.

Your program execution will be stall.

You cannot go to the next state.

Understood for this, the solution

after here, if

before that, what is the dictionary name?

Qualification dot get function.

In the get function, I will pass the key.

For example, I'm passing BTech plus come

from a BTech available or not.

Yes, Yes. I will get value.

I'm passing mtech. Mtech.

Also available value be return

passing MBA, the MBA also available

2022 passing PhD

available or not.

Previously what you got? Yeah,

right now you don't get at that.

You get one blank object. You get one null object.

The nu object in Python. That's what I have used.

What is that? YE None. None.

I get none because of none. Nothing is displayed here.

Just apply the print. You can see print off.

Understood. Now tell me what is the functionality

of get function here.

Take the key into considerations

and get the, if there is no value,

Legal modify.

If given key exists, it value,

it does not none, but it is not throwing error.

Right? Okay. Okay.

Already know if statement.

What are the if statement format, if condition,

if I'm taking your bank accounts,

think that this is the bank in Shelly started

more customers to this bank.

You are the first customer after.

Now this is every, I'm running

this now.

Your first customer, your account number is 1 0 1.

You are going to deposit that 10,000 rupees. Understood.

Come on. Bank of account number.

Let's say the account number is key.

The value I'm giving one zero

amount, 10,000.

So which is key, which is value.

Tell me among these two account number is

key amount value.

Why? Account number is key.

It is unique because your account number 1 0 1 means

that should not be assigned to other person, right?

That should be unique, but your balance can be 10,000.

My balance also can be 10,000. Right?

Value can be duplicated.

Understand bank of account number

equal to now print over bank.

What is the reply? This is the way how

to insert a pair into dictionary initially.

How many pairs initially? How many pairs in the dictionary?

Zero. Now first time you are going

to insert one pair into the dictionary.

Now just check after accomplishment of this what happened.

One period added or not.

That means a dictionary is allowing a pending a pay or not.

But a syntax might be like this

statement might be like this.

So based on this dictionary is immutable or immutable?

Immutable. Understand.

Okay, first time 1 0 1 account inserted

after that, that's a second customer.

1 0 2. He is depositing

30,000 previously.

How many pets? One.

Now I got two pets.

Now my bank has two customers.

One more customer I'm adding later we play different styles.

Customer 1 0 3. The amount one

lack is depositing.

Now I have three account holders.

I mean three customers in the bank.

Okay, enough this, the number of customers are, okay,

your account number is 1 0 1.

Now we want to withdraw 3000 rupees.

You want to withdraw 3000 rupees.

Come on sir, help me.

What is your previous balance after withdraw? 3000.

That should be 7,000.

That means indirectly you're going to update it or not.

Come on. How to update Just sub

bank of what is your record

number minus SQL

to minus.

Because when our is done that should be deducted. Right?

Minus SQL to 3000. What should be the latest balance of you?

7,000. Check it.

Correct or not.

Now your latest balance 7,000

because sometimes you are 1 0 3.

Suppose if you are 1 0 3.

Your current balance, again, you are depositing 50,000.

This time you are depositing. Come on tell me.

Bank of 1 0 3 equals to

what should be the latest balance,

correct?

Yes. The dictionary

is allowing update or not.

It is allowing update. Allowing a pending, yes.

Update hours. Understood.

Okay, you decided 1 0 2 decided to cancel the account.

Delete help me. Uh, what is the statement?

Delete, But before deleting what should happen?

You are 1 0 2. We want to cancel account.

What should happen, first of all, first of all,

what should be the first functionality?

First you need to withdraw the money.

Understand first, withdraw the complete 30,000.

Then cancel account. Understand. Come on, do it first.

Withdraw complete amount.

Bank of one euro two minus SQL to bank of 1 0 2.

When our bank of one euro is passed, what will be returned?

30,000 will be understand. 30,000 will be returned.

Now bank of 1 0 2 minus SQL to 30,000 will be applied.

So that what should be the latest balance of you.

That is zero. Now daily you can cancel icon.

Understand. Now you can delete that.

Come on, tell me Dell Bank of one.

Now check the bank. Previously, three customers.

Now two

customers understand the point.

Okay? Now this is bank.

I have two customers, 1, 0, 1.

That is you one to withdraw 10,000.

You want to withdraw 10,000. What is your balance?

7,000. What do you want? 10,000.

But fund is not available, right? It has to show you.

Suppose you are asking 4,000 available or not.

Then that should be with the draw. Can you help me?

Not 4,000 statement. Come on.

10, 20, 40, 50 and 60

of zero.

What is the output?

This the next number six won't allow.

Understand but, but I want to read it.

I want to read it. Perform loop.

Perform loop, set interval, but it won't allow index numbers

or X and yes, X

got it.

Why disorder? Different order. You got it?

Yes. So in order collection,

every time you're using order will be changing.

Understand? That's why you cannot fix

like a proper index number.

Practice. Understood.

Based on this, some work.

These are the genders of your employees. Cost is male.

Second one, female. Female.

Again,

male, female.

Female. Totally. How many employees we have?

Seven. Seven. How many males?

How many females? My question,

all unique genders in our organization.

How to get it Master. What is the data type of gender?

Data type of gender list.

Convert that list into set first

when you convert into settle mail, only one time, right?

Email also one time, right?

Come on now, check the output.

No duplicates. Now for this apply land.

Understand there are two vendors.

Similarly, there is a sales transaction.

Your sales transactions are from different cities.

Let's say first transaction from Rabi, second, Bel.

Next again from Rabi, Rabi,

Delhi and

think that in this way hundred sub transactions happen.

Totally. How many transactions? Seven.

From how many From how many cities

your customers are from

or from how many cities that transactions happened.

This is my question. How to get it

set

answer should be.

That means what happened. Finally,

which function executed faster.

Set function.

Within that other function I have given

always faster in function will be executed.

The output is possible to upper function.

That means faster set is executed.

That is at least positive.

Now final result of line is, is three. Understood.

Now my student qualifications,

ptech we did from university.

Second student ptech from J and u.

One more. B, S, C from ou.

Next one. BO

from ou.

Next one. B, S.

What happened? This what will happen?

Key should be unique, right? Key should be unique, right?

So here BS available two times,

but this is a different student.

This is a different student in this case should not,

you should not maintain this as a dictionary like this.

Understand you make it like this.

1 0 1 is my first student. That is key.

That is valid. Understand if I

maintain student details like this in the dictionary,

my student number is key, is education details.

Sir, value.

Understand that in the education details I have couple

in the couple two fields are there.

What is that Your qualification name

and university understand.

What about second student 1, 0 2.

What is the value is a couple BTech from J and D.

Next 1, 0 3.

B-S-N-U-O. University.

Next

4, 1, 0 4,

The last student 1, 0 5, B, S,

A from U

somewhere it is missed, correct?

B. Where

we need comm because that is a couple.

We are keeping

total of five students for that five pair.

Now my question is like this lesson. Question.

Question. How many universities are there?

Three. Three. Three universities.

One. How to get that

set off?

Q.

Q dot values. Values.

Okay, enough.

Is it correct? What is expected? Answer?

Wait, wait. What is expected answer

I got like this

values what you get

as we did not run this now run this.

What is the output? All values have been collected

or not from this.

What do you want? What is your question?

How many universities? That means you are not interested

about complete story, right?

You need to pick only university. University. Come on.

Come on. Do it. Only university. Name one yesterday class.

The expression pa.

What is the far for university?

Pure values. This is for low.

I'm taking that university now check the values.

University is a double in the double low. Where?

Where is the university? Second one.

Second one means index number one.

Now check it when

university separated or not.

What this supply set,

what this supply set what you will get.

Only three J to UENU apply land. Now

what this result? I'm applying

Learn.

Check this. Now

what is our question?

How many, How many universities?

Finally we got it right like this.

Okay, just practice it.

Module 11: Functions in Python

Video: Functions Introduction

Discussed about the three types of collections.

One is list capital and DC.

One more thing that is set.

What is this collection now?

Unique. The basic won't allow duplicates. Okay.

Only unique portal. It will be taken.

So for this symbol, same as D,

but in the dictionary, key value base will be there.

Here only direct value. For example, I'm saying yes.

Whereas per set, that's a

total six values are here, but how many unique values?

10, 20, 30, that's it.

10 repeated for multiple times and 31 time.

20 multiple times,

but it takes only one time, just this.

So 10, 20, 30, even though it passed duplicate values,

it would similar to that.

I'm taking a list of my employees

that the department numbers I'm taking.

So first, second, third employees are from

11, 12, 13 departments.

Next employee from 13.

Next from 12, next from 11, next from 11, 11, 12.

Now DNO is a list.

How many elements in the DNO, which is department number?

There are nine elements.

How many unique department we have Three. Three.

I want to find all that.

So DNO is a list just to convert this into set,

to convert a list to set.

Set is a function set up.

I got only 11, 12 30. So duplicates eliminated.

But your question is how many departments

that we have set of?

Yeah,

I have three unique departments in the organization

but similarly.

One more info is a list.

Each element is the top.

So table has its own.

What is the meaning of first one?

Id second is name, thought, gender.

Fourth one is city location.

So that is first employee

second,

the third employee.

The last, last word. What?

The last second I'm giving.

So what is the type of info? High level, that is a list.

Each element is what I'm taking first element

and testing the data type.

That is topic.

So finally it is a list of topics.

Now my question, your implies are from

how many unique cities,

how many unique cities?

Three. Three. What are those?

Yes, I want programmatically that reply.

So I'm asking what unique cities first

to separate only cities, first to separate only city.

Let's say cities. It,

I already know this.

Transformation expression I follow

we in list.

The list is, so each element is coming to VV is a apple

in the place of expression.

I'm giving minus

because C post is is last one.

Now look those cities

only city spot now apply set for it.

Unique will get

Delhi and pool that if you look into this data,

data is in a sending order.

Faster. Delhi. Next, next pool.

But supply lab.

So by seeing this output I can

understand from three different cities.

Got it. Same style. You tell me answer.

How many genders are there in our organization?

Programmatically please.

I'm asking about gender first to separate that gender,

let's say gender people left side

expression par bar V.

So V is the in the January 3rd one, index number

or minus two also you can say, but

because that is the last second, now look into this.

Genders

Duplicates

have two in.

Got it.

Okay, So list

technically what it is called.

Technical, not functional

is beautiful.

Be Object,

face operations allowed read

and double

opposite is immutable.

Object. The pace it allows

only, right?

Remaining operation is not possible.

Dictionary immutable or immutable is mutable.

What about set? Just a try. We did not try it.

Let me keep one side here. A hundred, 203.

Oh I'm saying yes, sub zero.

Whatever I

did not get it is not allowing you index number.

If index number is not allowed. Can you update that? No.

That basically is not allowing, even reading us

reading a particular development is not allowing understand.

So whenever you want to update what is a technique?

Indirect ways convert that into list.

Do your operations secondary convert thele. Set.

Understand. So this example we did for

couple, okay, one example.

Let's say SL is equal list of S.

Now look into this sl where list

now we can update, but this time there is no guarantee.

Hundred will be into the same beginning position.

What are the element you want to upgrade

to that application?

Said

that is already one.

I'm making it a thousand.

Okay, now we can order that into set.

Set up. Yes.

Now look into this. Yes, into

you'll see modified hand.

Got it. Now QU is 200, 300,000. Got it.

Uh, but that's a small question here.

So in the third line, can we use append?

Uh, here like S one SL append,

Yes, cell can use, but yes.

And what happens yesterday?

Because assist said I'm giving 2000.

Look into this, yes,

but accept.

Understood. So if you want to append any value

to your asset, same process.

First convert that into please list

append. Got it?

Yeah. In the list only only can we uh use a

append keyword, right?

Yes we Can. Yeah. Okay. Okay. Yeah. Thank you. Yeah,

Please list. We

can append it because that is understand

or not

list is a, that means we can perform Loop on that.

What about trouble? Double is a

what about dictionary?

Dictionary also et. Okay.

Lateral will come back to the sector.

Let's take a sample dictionary. That's a D.

Default dictionary.

Thousand value 2000

CCK 5,000.

I'm saying

again 6,000.

How many pairs will be there in today? No three pair.

'cause we have a rule key should be unique.

Value can be duplicated.

First time yay hundred pay 8,000 pair will be

created after that.

B 2000, C 5,000 after that. Yay. 6,000.

That updates past year.

So finally what will be there into this one? Check it.

I'll get only what if

while it can be any data time.

Understand. Okay. No, totally. How many page are here?

Three page dictionary is

or not Just confirm it

for KD 20 K.

So at each iteration it is taking only key part.

The first ation, what does taken? Yeah, second iteration.

B ation C understand,

but along with each one I want that value.

For hk I want value text name is what? D. D of K.

If I pass key, I will get value.

Understand, I'll check it.

D 6,003, 2000 C 5,000. Understood all of you.

That is what is your comment? DIC is A or not. That is it.

What about sector it or not?

Let's come this.

Look into this.

Yes. Yes. This sector, we have three values.

So index number, it is not allowing reading,

it is not allowing for I'm writing a loop

for VS print.

Yes.

What happened

is covered.

Why? Just guess what is happening.

Yeah. C set. Set has only three elements.

Actually what we are expecting in the ation

ation 200, we are expecting second iteration 300,000 we are

expecting but it is reding.

What entire set is reding

Because of three elements.

Three times entire es. Right? Got it.

Because index number support is not there.

Generally what happens if your C list to

what happens in the fast ation?

Yes. Sub zero will be upright in the second iteration. Yes.

Sub one will be up, right? Ation has soft two.

But here, index number sub not allowed. Right?

Entire ES up right here. Got it.

So that is the reason. So that means like uh,

what you can say here about the list, not interval.

Got it. That means each element is not a trouble here

but at any cost I want that each element then

understood into list

and perform your operations.

Got it.

Okay.

Now let's look into the transformations.

Transformations means

and a collection.

So in Python, technology transformation means is this one.

You want to do some operational each event

that is called transformation.

Already know Python, exclusive style,

like a left side expression, right side of faster.

We see all the transformations with the traditional way

and then the Python exclusive style.

I'm keeping some values

I see running.

Okay, 60, 9,000.

Okay, total 10 values that we have.

Now the task is for each element I want

to add a hundred different waste.

We will see way one,

I'm saying something why?

Which is a new list. Which is a new

list which is empty.

No elements into that. I'm gonna run a look through the X

for VX what I wanted to do hundred

for each, I want to add hundred.

It's not for the X, it's for the v.

V plus equals to hundred.

So in the first, what is the value of V 10?

Four 10. If hundred is added. 1 10, 1 10. This one 10.

I'm going to upend to Y.

Let's call now check the values of both X and Y.

X. Print one.

We did not run the first cell.

Now look at the same.

It start adding Y you have to upend B

because the expression, the result again restart.

I viewer. Okay, now just

so top second row is transformative for each element.

Hundred days. Got it.

Uh, this is a traditional programming stack

in any programming language.

This is a technique we follow first entry list

after that far loop within the far loop we perform

over transformation.

I mean you, you'll apply your expression.

The result you are appending to the MT list. Got it.

It's a traditional style, uh, python style.

Mm expression partner.

What is the expression? What is the for loop for V in

X for V Next.

So each one is coming into V.

Check the Y, same

clear.

Uh, similarly

and taking our names.

Six names are that two. The names are like this.

Some letters are mixed and cases

where this is a story name is having five elements.

First letter cannot into

auster into lower.

So for this manually we try something. Suppose que

if blank space.

I will because foster IES is a space

to cut off that I'm applying strict S equal to.

Yes, broad, strict. Now just say a sub zero.

Zero is fast. Next out spaces are cut off.

The first one is yes,

but a subzero, I'm saying upper.

Let keep this entire expression into C.

Where for first character,

next remaining

characters, one call.

I'm not giving. Second index, end index, what will happen?

Second element onwards. Secondary character onwards.

All characters will be taken. Check,

hand check.

But that is capital I'm converting the center thing into.

Now why upper case That is into ffc

Charact lower case that is into rc.

I want to concor these two things.

The RC plus FFC plus RC

string plus string is what sir?

String Concor. Now check.

So here I have done this only for one element.

I want to, I want to do this for each and every of the list.

Faster traditional way,

faster traditional

way, empty list faster.

That's a new name equals to

and blue.

Cut out the spaces and do

street,

not street.

Next.

Hmm enough.

Zero. I zero.

Generate a statement. Two. Separate then how I just said

I read it in upper case.

Uh, how to take the first character enough.

Zero. Got it. Now first Charact is separated.

What is the expression for last character remaining.

All characters. Yeah.

Now one polar, not

this thing I'm keeping under

RRC For remaining characters.

Now finally I'm going to append

you have C plus RC into new name, new name

do append, you have C plus R.

It's new now.

New name

STR.

It has no attribute called upper one P. Extra time,

same thing.

So for names, faster spaces cut off,

then fast cannot into lower case remaining all characters

lower case upper remaining into of course

for this you have predefined functional.

So later we will see predefined function.

Try to implement the loss. Got it.

The traditional way I want

this is traditional way Python style.

Understand Ah me.

Let's say for new name

expression for

what is the expression you'll give

zero pardon

Plus enough dot.

Do we get it? Do we get the reply?

Can I say one color? Yes. Is it enough?

Do

Yes.

Got it Again here and do

understood

anything I can read your mind.

What is there in your mind? Why double times to pipeline?

Yes. Think it are storing anywhere

that we result.

No. Understood.

That's why second time I'll say and step up line. Got it.

Uh, come on, you tell me

same thing happened or

suppose I did not say step for the second,

second expression.

What happened?

Okay, what about the money?

What about DU only space cut off,

taken only for first expression.

Understand, but the second prior expression

that is not applied for second expression also have to apply

strip

Independently.

These are the four expressions you have here.

All these four expressions you have

applied into this expression.

Think that here in my logic there there are a hundred

expressions in my logic there are

a hundred express forming a hundred expressions

as a single expression is very tougher.

Definitely do some mistakes in

that case we follow our special technique.

I will create my own

function, okay?

I will create my own function.

I will call that function too. The expression place.

But how to create a function in python that is a key word.

Depth

that says some sample function

for testing that up.

Twos

some Yes speak.

I want to written that. Yes.

Well this is a style of creating a function in Python.

So after now the function part is ready, I want to test

that function, calling that function add off

hundred comm 200.

What is this? Hundred goes to.

Yeah, 200 goes to.

So mapping first one to first one second one to second

that the different also you can do

by specifying the parameter map those operations lateral

here I'm executing that function.

Got it. So on the same function style, I want

to download this function to convert first letter into upper

remaining letters into lawyer.

Okay, for this I want to develop a punch.

Let's say my function name first

input is yes, it is strict,

strictly spaces are not.

It doesn't matter. First cut off, then quel

do everything.

You are connoting into case.

Okay, next first character.

Yes, all zero. I want it as upper.

Next remain characters. Yes, all one column.

That means second. And what all remain characters?

No. Already in the beginning the thing is already employed.

Understood. Now new name equal

have c conation

have I'm going return.

Understand Now just test this function.

Stop one. Oh yes.

In the place of S I'm saying hell,

what expected should, should be cut off

after that H should be capital remaining off.

Got. Now if you look into this one, multiple expressions,

but if you look into this side, this one, that is it.

Single expression. Now look into your name.

There are finance names.

I want to convert each name into

passport and remaining lawyer.

Come on directly. Python style.

Each name is coming into Y

and I'm calling with the first upper off.

Now check hidden value.

We got it. So this technique

we'll use whenever we want

to apply multiple expressions in a expression place,

keep the mini of function

and call that function the expression.

Please clear. Hmm.

Based on this one more example.

It's LAC 20,000, 10,000

80, 90, 72

lacks.

Sorry, two lack totally. How many?

One more. Okay,

total seven value.

Sorry, I want

to classify each ly based on the salary.

I want to classify each employee into grades.

My requirement above grade equal two, one,

lack salary grade year 75

to one like grade B, 50 to 75.

Grade C. Less than 50,000.

Less than 50,000 grade D. What about this?

What is the grade of him? Yeah, grade.

D. Grade less than 50 T.

Grade less than 50 B. He is B grade 75.

Two. I'm like B

grade seven days.

C grade two, LA B grade.

And why I want to perform conditional

question.

Four minutes one team, did you understand this

greater than or equals to

help me Faster, traditional way

less faster, traditional way.

Uh, our learning of yesterday and day before yesterday.

Yesterday means last

and last class yesterday and dictionary.

Now today, na defend is far loop.

Understand to perform this transformation

faster, traditional way.

What is traditional way? First empty list.

That's a great equal to

empty list then far Loop.

Yes. Total is seven elements.

Seven iterations. My requirement is what?

Greater than or equals to

how many grades I want to classify four grades.

A, B, C, D.

Last one is DI make it as default. Okay?

G, R, D equal to oh, target. Yeah. Grade.

What is the condition for this?

If yes, greater than one equals two.

One like then grade equals two.

Suppose above condition falls Alice

and then you now

target it.

It should be greater than equal to 75. Less than one lack.

Yes. Greater than equals to 75.

And yes, less than one lack more

to say the second condition.

Understand why no need.

If a faster condition falls,

then only you are reaching the second condition.

Faster. Condition falls means what?

Salary already less than one. Like understand that's why.

Yes. Greater equal. 75 then grade equals to B, B grade

the next ALI salary greater equals

to 50,000.

50,000. Then grade equals to C grade

because you are reaching a third step means salary.

The first condition already falls.

So here we should say one more. Zero.

First condition already falls. Second condition. False.

Then only you are reaching third condition.

You reached third means second already failed.

Second already failed means value less than 75.

Understand that's why grade equal to C, not

to check for that D.

First of false second falls. Third. False means what?

Already you said grade value. D. Got it?

Hmm. Now finally I'm going to return, sorry,

this is not a function.

I'm going to append grade.

Now look.

Both print salary,

great grade.

Look at this password like means A grade. Next one.

20,000 D, 10,000 D.

80,000, 90,000. B 70 C. Yeah.

Understand.

Uh, our python style.

This is general program style, our python style,

the expression for,

but to view that expression, you have lengthy steps.

Okay? Definitely we do mistake. That's why create function.

Auto to create definitely.

Let's say my function name is two grade input is what?

Salary keep the same grade equals

to D.

Default target A grade.

If salary equal to

that G equals to yeah.

L. If salary greater equals two

75,000

B grade salary greater

than equals two 50,000.

And finally that is C. C grade.

So function ready first I will test this function

working fine or not.

Two grade is our function. A one lack 40.

What is a grade expected?

Okay, got it.

Two grade off.

Two grade.

7, 8, 2

Grade off.

A grade.

Less than 50 grade

for the third grade, just check it.

Lower local variable.

GRD reference before assignment.

Where is GRD here?

What we're using GRD.

First we said great, well we got issue

with GR d now run the same thing.

And so

30,000 is degrade.

78 is big grade and 55 years. C grid. C grade.

Understood. Uh, now in our Python style, I'm going

to transform all the salaries.

These are original inputs radically on it.

Let's say, let's see,

but two

grade for V.

So try to map and list one by one.

List off zip off S Correct.

Just check this

all while is or not.

Just second.

Hmm, correct.

So in this way you can perform transformations.

Final transformation. One more thing. What is the time now

still?

One more example please.

You have employees like this

from the departments, different departments.

That's not

totally 12 employees think

that these 12 employees are from different departments.

Totally. How many departments you have? Okay, no.

5, 11, 12, 13, 14, 15.

I want to transform these numbers into department names.

Suppose 11, marketing 12 means hr.

13 is finance and 14, 15 remaining.

And I want to target only 11, 12 30

remaining all departments as other what?

What we are expecting if it is 11, marketing, hr,

finance, hr, marketing,

marketing, sorry.

And 30 finance. 14 hour, 15 hour.

Like on that way I want to transfer. Uh, come on directly.

The second wave, style or python? Exclusive style.

First creative function.

F two two department name

input is department number.

How many departments you have?

Marketing, hr, finance,

other totally four things you are targeting.

So default value.

If department number equals to level, then D

marketing a

department number equals to 12.

That department name HR

still department number equal to 13.

The department name.

If it is not level, not not 13 already.

Other, I'm going

check this, I'm going to apply on the transformation.

D name SQL two.

What is our function? Two

V, r, V in.

Now each number will be

transformed into the department names.

Understand, try to map them. List of,

so you'll see where our level is there.

That is marketing 12, HR 13 finance. And 1415.

Look into this other. Got it.

Now with the second way better way. I wanted to do this.

One more style.

I'm creating one more function.

Define 2D where?

DN four department name DNY is input.

I just wanna add only three departments.

11, 12, 13 create a dictionary.

DFO equal two seven.

What is the symbol? What is the key?

Department number is a key value market.

12 means hr 13 means

14, 15, 60 remaining.

All the things should be other. We cannot give keys for them

because how many numbers are there?

We don't. That's why.

What are the things that you want

to target to give them only?

Hmm. Now I have three

dm.

Mm What we are passing

department number are pass

after, not just we'll check uh D info of level.

What is expected? Sorry, two dm, two

DN is our function level.

I am passing.

I got marketing.

Perfect. Well

that should be HR 13.

Finance. Finance 14.

I will get error because 14 key is not

available in that picture.

Okay, easy. Tal already know this.

Not get off. Okay.

What it returns, what gut is doing, if available,

it returns value if not available, it returns from now.

For that, now I'm applying 20 default value.

First I passing 11.

If 11 past the marketing, I will get if 12 past,

if 13 past finance.

Other than these three,

which is not available in this dictionary.

If I pass, I will get other now test it.

But 14 simple.

We not in general languages,

we are using a switch case statement for this purpose,

but we don't have switch case statement in Python.

So that's why we took help of dictionary. Understand.

Okay, so in real time,

Can we use any techniques here?

I'm sorry, pardon? In real time we can use any technique

right here to get the output and on.

Yes. Okay. Any, any technique you can use.

But this is simple technique.

Okay. Understood.

Because traditional programming style like conditions,

conditional style, if you want to go like a DN vehicle equal

to 11 DN, vehicle equal to 12 DN equal to 30.

Like this thing that if I have a hundred departments,

how much lengthy of each statement then got it

instead going that directly can use diction.

Got it. So finally this function is working fine.

I'm going to apply this sum to the transformation.

Look at the DVOs. This is your original input.

I want to transform them into department names.

Let's say DN equal, what is the expression?

2D. Now let's say V,

R, VN, now D.

So each number is here. Come into department names.

Understand you can use anywhere.

Okay, so for today, we close this session

and tomorrow we'll go to the more transformations.

So finally, what is uh, like a land agenda of today?

Faster set after that.

Transformations with the traditional way

and our python style of way,

but more expressions are involving into this.

You are creating a function, how to call

that function into your transformation

before applying into the transformation.

How to test that transformation, test that function,

whether it is working fine or not.

If everything is good, then applying to transformation.

Got it. Okay. Thank you.

Video: Transformation Functions

Welcome to Python Sessions.

This is a sample structure of a record.

Each card has a double here.

That means each apply has a double here

and names are like this.

This is from 1213.

One more last record I will give here.

I have five employees here.

What is the data type of this info here? That is a list.

Totally five elements. Each element is what?

So totally input number of fields are file.

I want to create more number of fields.

Some fields I want to update. Some fields I want to create.

Okay, so my my required logic here,

although the expected transformations,

so Id, I don't want to disturb.

I will write my requirement here.

I it should be as id no change

if it is named the transformation expected

post letter into upper case

letters into my case.

We have created a function for it.

The next transformation for the gender salary

based on the salary, I want to create tax.

That's a 10% on the salary, hr,

20% on the salary,

and then net salary I want to target.

So three new fields I want to create

and also based on the salary, I want to

transform each employee into grades.

So greater than equals to one lack. Yeah. Great.

So greater than article to 75

and less than one lack.

That is B grade.

Similarly greater than articles to 50

and less than 75.

That is C grade and less than 50

degrade.

So this is my transformation expected

and also one more transformation department numbers.

I want to transform into department numbers

like 11 as marketing.

Follow US hr. Yeah.

13 as finance

remaining, I mean other than 11, 12, 13

as other.

So this is what I require that

with one single function, I want

to transform all these things.

Okay? So without like a complex functionality like this,

that means more number of transformations are involved.

Faster. Independent lead develop function

for each transformation.

Ladder club, all the things into one more function.

Let's target the name, what is expected

First letter remaining letters into layer

the function I'm creating here.

First up, passing some name here.

So given name might have like a left and right side.

Some right spaces cut off the

everything is converted into layer.

Separate first character enough.

Convert remaining

characters enough.

One color, that means second and words.

All remaining characters already.

Those are inter lawyer not need to say

finally you have C, bless us

string conation first character and romantic characters.

You are concor native. You are returning this expression.

So this is a logic for first upper test it

with some sample value or stopper off.

So space should be cut off

and that first letter head should be in upper case remaining

all the things into lower case.

So function, working fine.

Similarly, what is another transformation? Gender.

One more transformation I will include here

in the case of gender.

M as male. Similarly, Fs, S,

female, female.

If any other thing is there,

I want it as invalid.

So gender for the gender.

I'm going to write one more function here. The function.

I'm deaf to gender. Pass the gender value here.

Whenever you are getting a string,

always better practice first to cut up the spaces.

Cannot that into even the upper case value passed.

That is cannot.

Gender equals to saying

that in as a default value.

If G equals to yeah. Hmm.

What should be the general value

as if G equals to?

Yeah, the general equals to

if, first of all second.

False already default value,

which is inval parallel.

I'm going to return that gender.

Let's do test of it.

Whose gender of ya?

Even though if you pass capital letter, it'll work out.

If we keep some spaces also it'll work out.

Got it. I'm testing with the, yeah,

that is female testing

with some X

in value.

So function is working fine. Mm-hmm.

So tax that is a single line expression.

HRA single line expression. One more net salary.

That is also single line.

I'm not creating a function for it. Uh, come on next.

What's the next? I want salaries into grades.

Okay, let's say the function name is two

grade input is salary, whereas

for salary, how many grades?

Four grades we want to classify default lost grade D

and fixing as a default.

If salary greater than equals tools, one lack,

then I want grade value

as yes.

If alternative salary,

if salary greater than equal to 75,000 and salary.

Less than one lack, no need to say this one.

Because faster condition falls means what?

That is already less than one lack. Okay.

Then your grade is B.

Still salary greater than oracle equals

to 50,000.

Definitely that is less than 75.

Then grade equals to C

if it is less than 50 already.

Default value I have set here, which is AS and B.

So finally I'm ING grade.

So function ready for this, I'm going to test it.

What is the function there? Two grade one. Like 20,000.

I'm passing. I should get

a just 10,000.

I'm passing D.

60,000 C,

80,000 B grade.

So function is working fine.

The next step, what I'm going to do is

for the department numbers yesterday, we work

with the department number in two ways.

Okay? One is one help is with the dictionary directly.

The second technic, I'm taking define

to D, I'm passing DNO

into this as input.

That's a dictionary. DFO is a dictionary

of department numbers

and it's department name values

what the thing we wanted to call target.

Eleven, twelve, thirty. Just keep them. 11 is key.

11 is what? Marketing. 12 is key.

Hr

13 is key values.

Finance

the info dot get of the

OR to now just check it.

What is, what get function is doing if the given key

availability, it returns value.

If given key not available, it returns non empty object.

I mean non object. Just call

it 2D.

Name of

13, what should I get?

Finance because 13 available. That's why it's written.

Finance for example, 14 amp 14 not available.

I will get empty object, which is not,

I did not get any other here.

Suppose if the value is not available. Default value.

I'm fixing it here as other.

Now retest. This 2D of 40,

I got other understood this technique.

So this function I will use into the transformation.

Now I'm creating one final transformation.

One more final function.

The file

transform entire top.

I will pass into this one.

Let's say the top is E for employee.

So in our uh, input object info,

what is a high level data type?

That is a list. Each element is a top. The entire top.

I will pass into this now is at Apple.

Where is id? How is the schema fit?

You have the schema. First one id. Next one name. Next one.

Salary, then gender, then department.

So this is the structure of your apple. Come on.

Separate each entity. Id value EF zero.

Name EF one for this EF one.

I'll apply this past already. We created this function.

Understand? Hmm. Come on. Similarly, very salary.

That is in the second one. Second one means third one.

Third one means second. Index.

Based on this, I want to find out tax

salary into how much percentage?

10. That means 0.1. Next HRA 20%.

I want salary into 0.2.

Next net. Salary.

Salary. That's HRA minus tax.

Understand. Next, the grade

of the grade already have a function grade equal to

two grade two grade of salary.

Same thing for department name, master department number.

I will separate department number equal to what is position.

Last one. Yeah.

Minus salary already separate.

Next one. What is spending Department number is

spent two.

DM is a functional recent function.

Next, gender. Gender values.

Yeah, we have, we need to transform them as a male female.

Like that Gender equal to

first to separate that one.

Yeah. Minus two.

Minus two means lost to second one.

What is the function? Two

function, right?

All required transformations ready for

Id name salary tax.

HRA net salary after that.

Uh, gender.

After gender I want uh, department number. Department.

Department number. Probably nine fields that you are

input has five fields.

Output my nine fields. Got it.

I'm going to return a couple off.

I'm going to a past idea I want. Next. Next.

Next facts.

Hr, A net salary after that grade.

Then gender. For the gender. I set gender.

Then finally report number, department.

Totally 10 fields, I think including the department number.

So finally, input what is for this function?

What is the input? Input is a double

what this function is returning.

Finally, a double. Got it.

So this is what our required transformation

just to first test it before applying on the transformation.

Uh, what is the function name? Transform.

I'm passing one. Input as a pu.

I'd say 1 0 1

salary.

90. Gender.

M the value department number. Let's say 14.

Now look into the output of it.

Id name salary tax. HRA net.

Salary grade.

Gender, department number, department.

Okay, got it. So function independently working fine.

Now I'm going to apply this transformation on the data.

So what is your original data here?

The original data is info.

I want to apply about transformation

to each step directly.

Two ways of transformations, you know, traditional ways.

What list

after that far loop within the far loop apply logic

and then up the value to the T list.

That is traditional way.

Our Python exclusive style within the list.

Within the list. Left side expression, right side far. Okay.

Mm-hmm. Try to go with the second way directly.

New employee equal to

list expression.

Far what is a far look far? E in info.

So each value info is the least.

Each value in the info is a double. Now E is a double.

I'm passing the tap into transform function.

Transform off that. Now look into new EMP.

So each employee, because

of entities showing vertically line by line.

Let's try to print for V in

new m print that V.

Got it. So input has five fields.

Final output has then

fields later.

I want to write this result into file.

I can write this into file.

I can write this into some database. Understand.

So later, once we step into the file concept

and database's concept, we'll see how to load

and read from the files

and uh, from the tables of the database.

Okay? So in this way,

multiple transformations at a time we can, uh, perform

within a single function.

Got it.

Now final output is a new, new EMP.

Now I have a small question here.

Fa, what is a total net salary?

I don't want total salary. Total net salary reward.

All, All the employees.

Total net salary award.

Where is the position of the net salary?

This is salary tax. HRA.

This is net salary. That means what is your position?

Six to one. Six to one means index number.

Module 12: Analytics Functions in python

Video: Statistical Function

Very basic level mathematics.

I'm bringing you all,

you know this mean mean means

what? Average.

Average. You have three types of means.

One is automatically

second one geometrically.

Third one

harmonic mean.

So while developing these functions,

you will learn a lot of logics here.

Suppose automatic mean means is a simple average.

That means

if A is X somehow of extrover by number

of elements, okay?

Geometric mill is the end route

of product

Of Variant.

For example, if x value is

10, 20, 30,

what is the product of X?

10, 2, 22 30

which is 6,000?

What is the number of elements here? Three.

What is geometric mean?

Now the

of product effects,

that means cubic ru of

That means 6,000 to the power of one by three.

Right? Sir? This is geometry.

Similarly, it is harmonic me.

Sma sigma means some

one divided by X for the same.

What is HR 10 value? How much?

One by 10 plus

one by 20 plus one by 10.

Understand seems to be complex,

but with python, with easily we can

create functions for this.

Let's look into this

custom manual without function.

The aim for automatic, normal mean

let's say X equal to

some five values.

I kept. What is the

total sum effects?

What is a count and

what is the M total by?

Yeah. Great.

A automatically.

Yeah.

What is the sum here? 10, 20.

Total sum is one 50. One 50 by five 30.

So what are the independent expressions you have seen?

I want to make them as a function.

The file. Let's say Amy, I'm passing a list here.

Here X is a list.

Hmm. Total equal to sum of x

length equal to line of x written.

Now check this function

of x.

X has five values.

Similarly look into the second formula.

Geometric main function.

So what is geometric means is

and root of product of that vari Product means what?

Multiplication of all limits.

Faster manually without function.

Look into this. X. Again,

that's a PP four product.

Initial value zero

or one, one.

I'm going to multiply. If I give zero utilization

zero and anything is zero, right?

If I multiply, multiply a value with the what?

The value won't be changed. One.

That's why initialization starting with the one,

writing a loop for V

and X, P in two

equal to B,

print P, B the does this

logic works or not

in the ation before ation P-value work in the ation.

What happens? One into 10.

One restore into P, second iteration.

10, 20, 200 third iteration 200 into

36,000.

Fourth ation two, lack of 40,000,

6,000 into 42 lack

40 into 50.

Next iteration, one core 20 lacks.

Okay, so

after competition of all these things,

your product is ready, just check it.

One core 20 lacks here.

Here, okay, for this,

Taiwan means what?

To the power of one buyer.

How to apply the power value in python.

Double star. Single star means multiplication.

Double star means power valve. Suppose two.

QB one, sir. Two. Double star three.

What is the result of two Q eight? Two.

Two. Two. Two. Two, eight. So here, double star operator.

Suppose I have value a hundred square root.

I want square root means to the power of one by two,

one by two means 0.5.

Double star 0.5.

What is square root of 110?

Square is a hundred square. Root of a hundred.

Is that, so this technic use

what is the end value and value?

You have five values now.

Okay, now product is B.

Double star one buyer.

Your expectation will be failed.

Your expectation will be failed. Why?

Because suppose

yay two BRA plus B In two C.

What will happen?

What will happen?

Huh? Ah, faster multiplication.

Then the result will be added to a

understand by example.

If I say why

into D, what will happen faster divided by

faster priority for divided by

that division result will be multiplied with the D.

The total result will be added two.

Yeah, understand.

For example, if I say double star

C, double star

four, R, uh, sum, E.

Now faster priority, double star power

second priority slash division.

Third, priority multiplication, fourth priority.

Either it can be plus or minus, it doesn't matter.

Understand. Okay.

Now what I did product,

product means what?

One buyer. One buyer.

What will happen if I say like this?

What will happen faster?

P two, D four, F 1 0 1 will be applied for the total result

and divided by end will be applied.

Understand? Check this.

Our expectation is wrong.

I got 24 flags.

So to work on this problem, what you have to say,

keep one by and in.

Now separately, one by one will be computed.

That result will be powered to the P. Clear All of you.

Now check this previously automatic mean result is what?

30 geometric mean value should be less than two.

The automatic, which is 24

or 26 point something, which is less than two 30, right?

So perfect rate is worth. Got it.

What are the independent expressions you have?

I want to create a function for this. Define G.

Let's say Java input is X

your plus I need product.

Product. What is the logic?

Logic onto the notepad?

X. Uh, B star equal.

Now P, ready by the shot. Now number of elements

I want gm.

That means your GM

is ready, written.

Now your function is ready.

Hmm? What is the function name? X?

Just a silent. Got it. Same result.

We got it top, not now. G. Ready?

Similarly harmonic mini one,

look at the form.

If I have the values, 10, 20, 30 number of elements,

three divided by sum of one by 10,

one by 20, one by 30 like that.

But in our given values, 10, 20, 30, 40, 50.

Not one by 10, not one by 20 like that.

Now we need to transform given value, size one

by 10, one by one.

Got it In Python. That is very simple.

Look at X again,

already know.

Let's say this is called reciprocal ax.

The expression far loop, this format, you know, right?

The far loop I'm giving for V in X. How many ations? Right?

Five at each iteration value coming to V.

What should happen to that V one

by V in the first ation.

What is the value sir? And

that 10 will become one byte.

That means 0.1. Next iteration 21 by 20.

Understand on that way this will work. Just look into this.

One by is 0.1, one by 20 is 0.05 like this all remain.

Finally, what I need, just by seeing this speak,

I need sum off one by 10, one by 20 like that.

Apply some function for it, some off Rx.

Point two, two eight.

Final formula, number of elements divided by

sum off reciprocal.

Hm? X

already is there.

I can use Y divided by some of

Rx this.

Hm, value definitely should be less than two.

Geometric mean what is the automatic mean value?

We got geometric mean 26. 26.

Still it should be less than

then only your value is correct.

21 point. Hmm. What are the expression you have seen here?

Make them as a function. Come on.

Yes. Is the list

I need reciprocal one

by V for V

and XI want some of it,

some of Rx I want land.

Land now YX or rx, anything.

Hm. Yeah.

Divided by some of Rx.

Got it. Final

returning four.

Just I'm testing this H been off

X previously.

How much we got same work. Understood.

Now just check this. I'm creating a function

means one single function has

to return all the three types of means.

Got it. Input is XI want automatic

mean already.

Already. We created one function that is

GM I one, G, G ofx.

Yeah. HMI one

H mean ofx

return at apple off

am GM H.

Okay. Now if I call this function,

it'll written all the three tabs.

That means the function within

that other function you are calling understand.

Let's test this

function.

Ready? I'm going to call means of X.

What is first one? Automatic man,

geometric man armor.

Understand.

Okay, similarly, one more

simple clarity sir.

Especially data analysts and data scientists are here.

Please try to concentrate on this point

sir.

My body weight 78 kg

grams make this as a value.

78. I can say three.

Oh, this is my body weight. Your weight sir.

7,800.

Yeah. Among us, among you and me, who is fatty

ya?

Among you and me, who is fatty?

8.2. Sir, what is the difference between you and me?

200 grams just because of 200 grams lesser you are slim.

I'm fat in. Ah huh.

Got it. So

whenever you are comparing two person weight,

you are neglecting the decal point

or not understand

whenever decimal point is neglectible,

I mean fraction is neglectible use.

Yeah. Yeah. A automatically

understand the point.

Okay, all of you touch with the cricket

at least not play at least watching cricket.

Uh, then super betting then super.

Such an average 70.

I don't take young generations. I take my generations.

Gogoli generation gang average

40, who is best player?

Average runs first one, Achin, second one. Gali.

Who is best player based on the runs?

Simply you can say achi,

that at least you have some expectation

tomorrow if he entered into the field,

he can bring the score near two 70, maybe 70 plus

or maybe 70 miles.

Some kind of estimation it is giving you based on

that high average you are prioritize.

For example, run rate.

I'm talking

6.8, run rate 6.2.

Who is best? What is the difference?

Point to four, sorry, 0.6 is the difference.

Point six, which is less than one only,

but still lot matters when you talk about the run rate,

when you talk about runs that this will not important

to you, but when you talk about run rate

that that's more important.

Understand when our fraction is

important, use

Geometry Geometric understand.

Okay, you traveled to your home

after completion of this class with Uber.

The bill applied is 1 36

0.35.

That is a bill over home.

That is 1 36 rupees 35.

Okay sir here, question.

You have given 1 37

rupees how much you should get back.

No, nothing will get, he won't give you price.

Understand, even though he, he doesn't give you,

you don't object it he neglected.

Okay? You have given only 1 36.

Yeah, he'll also agree. Okay?

Because that

that is small fraction neglected over price.

DEC one neglected you can use. Yeah, yeah.

It's not a overpriced, you are,

my bank has given you a loan.

The loan value is

1200.78 cr.

What is the meaning of it? Okay,

I have given you 5.78 CRT as a loan.

Uh, tomorrow. Do you ask me just

by rounding the 0.78 I will pay only file.

Can you No here decimal important or not important?

Got it. Use

gm.

So always am is not good always. GM is not good.

Depends on the situation. We need to apply the things.

Got it. What is our class times are here?

9 15, 9 15.

What time I entered sir?

9 21 being a Indian. That is my fundamental right.

Five to 10 minutes late is my fundamental right?

Okay, in this case that is accepted. Okay?

Tomorrow, China India war started,

China has launched a cell.

Let's take a break. Have a smoke cell.

No, they're not The minute, not the second, not millisecond,

millisecond, millisecond is also important.

Minute to minute, decal point also important to you at

that time use our

understand the point sir.

Yeah, and depends on the situations

we need to use proper means.

Otherwise, our entire analysis say we predictions

everything will be failed.

You got it? Okay.

This is the story, but like unfortunately

Python don't have functions for all these things.

That's why we should create our own functions.

All of you understood?

Okay, this is waste discussion.

I'm just, okay, one more function.

Such an average runs 70

again ganglia average runs

69.

Who is good player in terms of runs, backs?

Achin a gang. Sir,

I got 34 marks in the mathematics.

I failed. I studied in 10th class only.

You got 35 marks, sir, you passed it. You entered.

You entered into intermediate.

Uh, who is intelligent in mathematics?

You are me just because of one mark extra.

The 2 35 score. Do you become mental?

Ah, yes, yes, yes.

Understand here these two persons mean is

same or very close together, right?

If two means are close, who is the best?

We cannot decide it. We cannot decide.

The first example, 70 40.

Then you can say such is good

now 70, 69 almost close together.

We cannot take the duration at that moment.

We take help off

variance understand.

After this a simple story.

Satin scores are like this in the first match,

62nd match

70, next match 80.

What is the total sir?

Two 10. What is the average sir? Seven.

Look at gang.

Zero second match 10.

Next match. 200. What is the total sir?

Two 10. What?

The outage,

but who is risky tomorrow match

Pakistan to Indians

whom you will say gang, why not goli?

His mood is good. Score 200 mood is bad.

You cannot predict, you cannot estimate it.

Sorry sir. All of you let that is disconnected on my,

so please confirm it.

Do you see my screen? Yes.

So yeah,

so here X is your variable.

I mean list in our python.

That is list X

bar means that is mean.

Okay? Let's say automatic means.

Suppose if the X values are like this,

what is the mean of X

proton 60 by three 20 Y minus X bar wheel spot.

10 minus 20, minus 10,

20 minus 20, 30 minus 20.

Okay, after that what I want X minus X bar, whole square.

Okay, minus 10 square,

zero square, 10 square

total two.

Oh understood.

Now what is the variance value?

200 by three. Understood.

Suppose suchin variance of such

I got hundred variance of Ali,

I got

1700, which is less value.

Such a variance, such a less risky.

That means less variance.

Less variance indicates less risky.

Where is high value with gang is high risk?

Understood the point sir. Like that?

Okay, forget about that analysis, how

to find out this directly help with the function.

Define variance input is X

first what?

I need the average

of X award for average of X already have a function.

Let's say MX for mean of x,

mean of X exam, right?

First manually we test it.

Look at this. X,

X

after that what I want X mean

of x ready X minus mean of XI want

first before checking the output, tell me what happens here.

If I say the statement X, yeah,

mean of x, what?

Output minus x value 10

mean value 30.

What happens? Not minus 10, not minus 20.

You'll get error because what is the data type

of the X listed?

What is the data type of mean?

Single value float list minus single inval.

Understand, but from each value I want

to deduct that mean value.

We use our comprehension technique. I said deviation.

Left side expression, right side for loop for V in.

Yes, at each value is that V

Here I'm saying V minus mean of x,

I mean amen of x.

Got it. What is the aiming value? 30 in the ation.

What is the value of 10 V? 10? 10 minus 30.

That is minus 20 in the second duration.

20 minus 30 ation 30 minus 30.

On that way, that deviations will happen here.

Got it. Next, what I want

each value square I want for this

in the same place I'm giving out,

apply the square double star.

Now check this. Square squares, right? Mm.

What is pending in the formula

for all these squares?

I need some.

Okay, let's

say yes.

Yes for some of

no entire numerator part, right in the formula.

30,000 Divided by

number of elements.

How to get the number of elements we

have now?

Yes. Buy variance equal

to yes buy.

Yeah,

understood.

Now create a function for it.

Def define variance first.

I need mean now. Facts.

I want to mean deviations.

X minus V minus mx.

MX is a meanwhile for VX,

for all these expressions, I'm applying square

for all these squares I'm applying.

So

our variance is equal to that's a

our we are equal total divided

by now.

Total formula already.

Now return, we are,

I'm going to test this.

We are fx. We have five values.

Same thing. We got it right. Let's try with the Chen.

What are the previous search runs?

60, 70, 90 gang runs

zero 10.

I'm going to test with these two varis of such

66 variances of

who is less risky, high risky.

Understand the point like this.

Come on

one more final.

Let me keep this pop off.

Standard deviation

simply as three a square root of

that is what is the variance for block

already have variance,

but that if you apply square root you'll get standard.

So simply define the function

V equal to already.

We created previous now strategy,

SQL to V to the power off one by two.

One by two means Final.

This thing you can use.

Please help me

Off.

Yes. Standard deviation is equal to V

to off 0.5.

Final return. Yes.

Now check this variance of X,

how much we got 200.

Then standard deviation of x.

Similarly, standard deviation of suchin,

8.16.

Standard deviation of

what is the average for both of them.

70. 70. How you can understand this one,

look into this picture, especially data

scientists and data analysts, please construct it.

I'm going to talk about such.

What is the average?

Meanwhile how much, sir?

Seven. What is the standard deviation?

What is the standard deviation We got 8.16.

Let's make it as eight.

That's the value. Eight. Uh, 70 plus eight. How much?

78 70 minus eight

60. Eight

60.

Most of such values are fluctuating between 60

to two 70.

That is whatever at understand.

Uh, tomorrow will be the match by achin.

He is going to play. Can you bet on the Sachin

that he will give 90 score?

He cannot dare it.

Understand the points, sir. So more so here.

This graph is explaining you.

Most of such values are in between 60 to 2 78.

Of course there are some values less than 2 62.

Yeah, about to 78,

but most of values, hmm,

same compared with ggo.

What is the average here? Still 70.

What is your standard deviation? 92.

Okay, let's make it a uh, easy calculation. 90.

It has devalue 90 plus,

70 plus

90, 60, 70

minus, 90

Minus, minus.

That in cricket runs no chance of minus 20, right?

That is what your understanding.

If it is zero, most staff such as

that gogoli scores are fluctuating between 0 2 1 60.

Just by saying this, who is good

but Sachin more good, right?

He can give up to one 60, right?

He cannot say, you cannot say that.

Understand. Okay, one more, uh, clear example.

I have two daughters, sir.

One, two recently.

Both has written their examinations.

I asked my elder daughter

how much score you'll get in the exam.

Her answer is between 62 70.

Okay, so what is my expectation on her

mostly 62 or 59.

I have some estimation on her,

but second order is smarter.

She said that between two, two to 98,

how much score you will get her arms raised

between two to 98.

Then what is my expectation that to 10th class,

what is my expectation with that?

Hypertension. I need to go to the bar, understand the point.

So where is less variance here?

Less variance, less variance.

D one, doctor one, D variance.

D two less standard deviation.

D one, high standard deviation. T two who is risky?

Two, T two. Less risky. D one on that way.

How to interpret that data. Understood.

So tomorrow, once you succeeded in the interim,

HR will ask you how, how much package you want.

Don't say like between 5 2 95.

So you'll get heart attack, okay? Yeah. So that is for you

Bar, sir. Hi,

Yes sir. Hi,

Uh, uh, bar, if you can explain a little more about

standard devi and, and and what is the basic difference.

Uh,

Both are like for example, indirectly.

Both are explaining you spread, spread in the variable.

Okay? Previously, if you the such values, such values,

sir, almost all the values are close together, right?

Like a 60, 70, 80, maybe. Next match 75.

Next match, 68, next match. 83.

Almost all values are close together, right?

That is a less spread here.

But how is the gole value one value here? Next value here.

Next value here. How high

spread here?

Less spread. The measurement

for the spread is variance.

More clear measurement is standard usage.

In our previous example, Achin got 70 average,

what is the standard deviation?

Eight 70 plus eight.

Okay. 78. 70 minus eight.

That means most of achin values are in between

62, 2 70.

Understood. Tomorrow he's entering into

the next, uh, match mills.

At least I have some rough estimation. Minimum 60 score.

He'll give some estimation. Of course sir.

Tomorrow may be dec out, but my estimation, most

of the cases that I can achieve from the

search understand the points.

So here, less spread is always lesser risks.

High spread is always

Understand The points.

Yes. So in this way,

different statistical functions.

Tomorrow also same concept, more interesting thing,

but more challenging for the development.

Also in all today we have created,

uh, I think five functions.

What about this? You mean

geometric mean harmonic, mean variance standard

in all these functions, input is only one variable,

but there are some complicated algorithms,

complicated functions,

but that input is not, um, input is more than

understand at that time how to deal there so

that operations will learn today.

So that if you practice these things

logically you'll become strong.

Sorry, it doesn't matter.

Uh, monitor, remember the formula simply your client will

give you the formula, but you should know how

to develop a function for it.

Understand, of course in python, lot

of predefined functions are there standard deviation.

We have predefined function variance.

We have predefined function,

but no predefined function for are mean

of course in core python, main function, not available,

but library we have,

but we don't have predefined function for geometric,

we don't have for harmonic in some cases at any cost.

I need to apply them at the time.

I need to know how to write my one function.

Don't worry about formula.

If you forget the formula, Google it.

Ask chat, GPT, it'll give you formula.

But by seeing that formula,

how you are creating function, that is important,

Right?

Okay, sir, all of you, thank you very much.

Video: Correlation and covariance functions

Welcome to Python Sessions.

Today we see some complicated formulas

that will work with multiple variables, multiple lists.

Let one formula, core variance.

This is the formula

two list.

Stop. Lets here X and Y.

We exceed each value of the XX experiments Average

Again, Y and Y bar divided by a minus.

The X minus X bar is x deviation.

Y minus y bar is Y deviation.

Again, the element to element to multiplication.

First element. First element. Second to second.

On the multiplication for all these multiplication.

Some waited. Got it.

So first try to explain me.

Is the recording going? Yes.

Think that these are the values of it.

These are the values of what?

So what is the mean of it?

Three. Three plus four. Seven. Seven plus five. 12.

That means average.

That is three. X minus x bar means what?

One minus three. Two minus three.

Four minus three. Five minus three.

This is X minus x.

Similarly, what is a Y bar?

Eight. 15. 26.

26. Five. Four.

Six. Five something. Let's say six.

Then what will be Y minus swiper?

Three minus six. Five minus six.

Seven minus six. 11. Minus six.

Got it. After that, what I want

elementary element multiplication X minus x

bar two.

One Y minus five.

That is first element to first element

minus two to minus three.

Six minus one into minus one. One eight to one.

2 8, 2 5.

After that I read sub.

Got it. What is the number of elements?

Eight four under each four.

Now what is uh, co variance?

18 divided by four minus one

18 by that

is understand.

So how to use that later will expect first how

to implement this manual.

Use

that.

This is temperature data.

Simply, I'm saying

this is ice cream sales data.

When temperature is done,

ice cream sales recorded is 20 lax for ammo.

Ice creams. When temperature is

12, it's 25.

When temperature is 15,

it is

29, 39, 42, 38, 31.

You can observe one relationship here. Yes.

Temperature is increasing. Ice cream cells is also eating.

I want to understand the dependency between them

for this coherence.

Technical values.

To avoid the confusion, simply I'm saying X equal to D,

Y equal to ice.

First. What I read mean effects mean

of I I directly.

We don't have mean function. Okay.

Creating a mean already. Yesterday we created, once again,

I'm created

total sum of effects count.

Then

calculating mean of X.

Similarly, Y what one?

The temperature. Wifi, ice cream sets.

Just look into the values.

Temperature average 15. Ice cream average.

Next step. I want Yex minus X bar.

Simply, I'm calling it as a deviation of X,

X minus X bar.

If I say X minus mx, wrong

because XL is m max is a single.

We have comprehension technique, X expression. Follow.

What should happen to that? VV minus.

Got it. Do the same thing

for

YB

minus V.

Now check both values

printed D

that supply some route for easy, simpler values

round up there.

Also, I do say

I'll check these values.

Got it. Now what should happen? First element.

First element should be multiplied.

Again, second to second thought to thought like that,

but this we have a technique.

Convert this into list.

Check the values now

to first, second to second.

Like that. Everything is mapped perfectly, right?

Um, now I want multiplication of each one.

Let's say D, x, Y

equal to expression

B, A for first, B for second.

D, X in the first ation.

What is the value of Y value of B?

These two values to be multiplied. Two.

Understand. Now this DXY.

Just round it. You can uh, check it round the result,

right?

Look at formula. What is ready?

X minus X bar two Y minus Y bar.

For all these multiplication, I need sum

yes for sum.

Some of the X

1 69

Numerator in the denominator, half 10 minus one

over SQL to yes divided by

XRY minus.

Now your

28.16 positive value

or negative value if result is positive, meaning is

both variables have positive dependence.

Positive dependency means if one variable increases

second variable will also be increasing.

Understand if one variable decreases

variable second will also be decreasing.

That means both does the same direction journey

as temperature is increasing.

Your sales is also increasing.

If it is ice cream, understand that kind

of understanding you get for example,

okay, let's make it function laterally.

Test with negative. I want to make this as a function.

Come on, tell me define,

let's say my function name is COVR four.

Two things. I'm passing variable.

One variable

first I what I mean mean of x

already have a function three central with all of X,

Y of YI want

deviation of

XV

minus M max

for

V-X-D-U-Y

do the same thing.

B minus and Y for V one.

Now both elements. Element, element multiplication.

I want three for this what we did, D, X,

YA

Into B, A comm B

in zip off.

B, X, A

for first one, B for second one.

Got it. Got. I need some.

Somehow I want count

x, r, Y, anything whole

variance equal, some divided

by young minus one.

If I don't give bracket, what happens yesterday?

I told you faster divided

by will be executed from that result.

The minus one will be, but we are expecting

young minus one to be separate.

Got it. That's why I keep them in brackets.

Finally, I'm returning see.

Got it. Let's test this

function created.

I'm going to test it. COV of temperature and ice cream.

Temperature T is pass to XIC is pass to Y in the function.

What's the value? Previously we got

our function name is

C-O-V-A-R-O.

Wow.

28.16.

Previously how much we got 0.1, say 28.

So it indicates

if value is greater than zero, what is the meaning?

Positive dependence between

temperature and ice

cream cells.

Got it. For example, I'm taking sweater sets.

Look into T. I'm taking the

sweater sets, but

temperature is 10.

It's businesses a hundred.

Temperature 12, business 90

15, it's 60.

The 18th it's bottom

when it reaches 20, 30, 18.

41 16.

Let's say 88.

Now check this. Four variance

between temperature and sweater cells.

What is the result now? Minus 90.

That means yes, temperature is increasing.

Sweater cells will be quickly.

Do we do shopping for, for the sweaters in the summer?

Understand like this,

as temperature increasing sales is decreasing.

How you can say by seeing this negative, understand.

So here our understanding today is less than zero,

not the meaning negative, negative defendants

between Variables, I mean temperature and sweater.

Uh, definitely. What is the relationship

between ice cream cells and sweater cells?

Tested four variance of ice cream.

Four sweater,

no negative only in

the summer, what will buy more

understand, but at the same summer,

satisfac cells will be now both are doing opposite.

If one increases, second one decreasing. Understand.

Similarly, if COV

is zero, approximately zero

or deadly close to zero

or equal to zero, meaning

no dependence between

variable understood.

So tomorrow temperature will be 32.

Whether our class will happen or not.

Tomorrow, temperature 32 class will happen or not.

There is no dependency between temperature and class.

Understand tomorrow, Saturday.

That's why no class, but

class will happen or not.

That doesn't depend upon that happens.

Understand, in

that moment you will get a coherence value zero

or close to zero, like 0.00, something like that.

Understand?

Okay, One more formula.

I'm bringing you a little complication.

This type I go with correlation.

What is the formula? Just check it.

This is a formula. Somewhat little complicated when compared

with the previous, but generally style is same.

So try to make, uh, with a single express, you want

to build entire expression just

to split them into parts part by part, just two.

Then it'll become easier. Okay?

I just take this copy for you.

This is the formula up to non numerator part, same, right?

Numerator, same. Come on, finish up to numerator.

First

directly function attack.

Hmm?

You file correlation.

Two things we need to pass.

What first mean of X holiday?

We created a

number.

What gave of y

deviation of x.

What is the expression? V minus M, X

or V In x Do same thing

for Y deviation

of YV minus one

or V in one element.

Element multiplication.

Two second. Yes.

In dx,

ready?

Entire numerator power sum off. Okay.

First lemme take a yes separately.

Yes. Sum off. DX one

after.

Now numerator part, part. Next. Number of elements.

Wait, numerator, part ready,

numerator part entirely into yes.

In the denominator. Again,

complicated expression.

You explain minus expert already there in the dx.

For each element, I need square For each element.

I want square. Again here, Y minus Y,

but that is there into DY, each element square, ever.

Now sum of all squares of X, some of all squares of one.

Got it. Let's split this into left and right.

Latter V, multiply. Mm, come on.

Look at only left part of the denominator only left part

EX square.

MM expression far.

Yes. Each gentle should be squared.

We double star two.

Understand. Do the same thing.

YSPE,

double star two for B.

Look back the four block, what is ready?

X minus X bar, whole square ready?

Y minus Y bar four square. Ready, sum sharp pending.

Mm-hmm.

But this I'm applying some

say SX, some of

say SY, some

of BX.

Sorry. BY hmm.

Next, both. AI already left sum to right sum,

yes.

X, Y SX two.

Yes. One for this entire thing, square root, I want

SX, Y.

Double star 0 1, 0 5.

Five

means point, right?

Point five. Uh, the entire numerator part? Yes.

Denominator part. Yes sir.

Correlation equal to yes. Divided by yes.

Return.

Understood. Doubler. Doubler is function. Next.

Next we are going to tripler. Okay, let's test it.

Look into the function in this way.

Four block can be any bigger. Just to speak two pieces.

That will be easy. Okay?

Function. Ready? So always correlation value will be

between minus one, two plus one on

minus one, two plus one.

If it is out of the range, your form

or wrong, your implementation run, no chance

out of range of minus one, two plus one within the range.

Got it? Now let's try

correlation between temperature and ice cream.

Point to nine, eight. Understand

correlation between temperature and sweater.

Double R

minus 0.9, understand

correlation between ice

cream and sweater.

Zero point minus nine,

greater than zero positive, less than zero negative

for which positive dependency temperature

and high sleep positive if one increases.

Second one will also be increasing.

Understand for example,

correlation between MS ft,

Microsoft, Infosys in fi.

These are the trade symbols.

I got value minus 0.78.

What is the meaning later to

if Microsoft share increases,

info share will be, uh, tomorrow one.

Now data analyst organization concluded

tomorrow infos is going to down 10.

Then what will happen? M, Microsoft, Microsoft share will be

almost infectious that immediately

transfer the funds from info system Microsoft.

Understand the point on that.

Web positive dependency, we can take advantage.

Negative dependency. Also, we can take advantage.

For example, if my result is a 0.9 between vPro

and in fee, the result is 0.9.

That means if FI fee is going to down means

vPro also will be going to downright.

So with withdraw funds from the in fee, don't invest on to

Got it On that pay, the data analyst will take it.

Got it.

Okay. A simple explanation for this. What?

What are the ranges of this? One minus one, two.

Less, less, less.

If value, greater than zero

and less than Oracle equals 2.5

meaning weak,

post weak post positive dependency.

Is there weak past?

That means both are doing same direction as me, but bond.

Bond bond is weak. What is the meaning of it?

Weak relation means what?

So actually, when I was uh, doing studies,

my mathematics marks regularly 33, 34.

These are the marks,

but my friends are getting 95, 92 like

that I observed.

Okay, Dalia, I'm reading two hours.

They're reading eight hours. Okay, okay.

Then I decided I just should read eight hours.

I increased my practice time, but

after the exam, my score is 35

positive or not positive relationship

between study hours

and my score, positive dependency or not,

but weak dependence.

Got it. Too much of practice hours. I increased less.

Score increased. That is called weak.

Understand the point. Weak bond, right?

Okay. Then I stopped there spending more time

because of no use.

Understood. Okay. One more point.

Try to understand if the value is greater than 0.5

and less than

one opposite word here.

Strong. Strong cost.

That means if one variable increases per 10%,

second variable will also be increased approximately

to the 10%, maybe 9% or maybe 11%.

That means strong bond between them. Understand.

Similarly, if the value equal to zero

or close to zero,

no dependence.

Understand if the value less than zero

and greater than minus 0.5,

the negative side, that is four

weak, minus negative.

If the value greater than minus one

and less than you are greater than equal,

less than minus five.

Five. Then

strong negative.

Got it. If value equal to one

meaning perfect, positive,

if value is equal to exactly minus one.

Perfect.

Yeah, perfect means if X increases 10 points means

definitely Y also increase 10 points.

That is what perfect means.

If X decreases for five points, I mean 5% is decrease.

Y also will be decreased. 5%. That is called perfect.

Same direction, positive opposite direction.

Negative, perfect, positive, perfect, negative. Got it.

Just for regular practice,

it's not a algorithm for practice purpose.

Think that I have a matrix.

1, 3, 5, 2, 3, 4.

You know what is a matrix? Oh, how much?

What is the size of this matrix? Two rows. Three.

Three columns. There is one more matrix.

What is the size of this matrix? Three.

Three rows, two column.

I want to do matrix multiplication. Hmm.

What is a matrix? Multiplication rule.

Number of columns of left side.

Number of right side should be equal.

Hmm. First, I'm just giving clue.

This is your assignment for you.

You need to fetch this row.

You need to fetch this column.

Element two. Element

application one.

Width, 5, 3, 3. Width. 5, 3, 4.

Five. Width four. Hmm. What is the result? One to five.

Three to three. Five. Eight to four.

Total.

Got it. Total 34. Ah.

Same thing with the two.

With one with the two. Come on. Tell me one. Eight. Two.

Two. Two. Three into one.

Three into one, sir. 5, 8, 2. Total.

Similar style. This one with this, this one

with this.

I will get one value here. One value here.

Finally, what I will get two rows

understand in python, this matrix is what list?

Offer.

List, got

One comma, three, comma five. That is first element.

Second element two, comma three, comma four.

Now list is complicated.

Let's say this is as x next to y,

Y is a list five.

Comm

one, four.

Now actually is list of list. Why is list of list.

List of list means high level

that is listed each element in the that list.

Again, understand.

Now treat that X as one metrics Y as one. Metrics.

I want metrics, multiplication. You know

how metrics multiplication.

If I say for V in the X, what happens, sir?

In the ation, what is the value?

That entire list is a value. Understand?

Mm. Same thing. If I perform with V, what is each one?

Each list, each element is a list.

You'll get this, but here,

challenge here, row.

I should take here.

Column I should take after that element.

Elemental replication already, you know, element,

element, multiplication.

Understand, but those multiplication against some.

Then you'll get first value here.

After that, with same value, you'll have to take a

second column, same process.

That means here you will have

to write a loop within that one more loop.

After the, you have only single loop.

Understand, so just to get grippo

or the logics, understand to get the BO

or the logics, try to implement this one.

Okay? But in Python you have a library.

In library. If you say Y off Y

matrix, multiplication will be completed.

No need to write any logic.

Understand, but if you go to the interview, they'll ask you,

I don't want with num, you implement your wrong logic.

Understand? So for that practice,

just implement this, try it.

If not worked out our Monday first part,

I will explain this, okay?

Right?

What is that?

Multiplication? How to create your list

is a list.

What values I have again, say new values. 1, 3, 5.

This is one element. Second element,

come to the Y.

Start, please observe. Here is a list.

4, 6, 1, 3, 2, 4.

You got these other two tric here.

I have two rows, three columns here,

three rows, two columns.

I want matrix multiplication, so

what will be the size of it?

Two by two. Two

by three is multiplied with the three by two basis.

We'll get a two by. Okay, so that result,

I want try this.

I shared this document.

Same in my yesterday document when I extended this, right?

So all of you.

Module 13: Exception Handling with Python

Module 14: File Handling With Python

Video: File Modes

I'll upload some sample file.

Here I go.

In the collab here,

there is a folder symbol on the left side.

So here upload option is here.

You can upload your local files and start working with this.

First I'm uploading one comment file,

which is about the reviews.

Just look into that file comment.

And

Here there are eight lines in this spell.

Past six lines are about class related comments.

Last two lines says is not pool is not enough.

Okay, what are the things maybe that is non-class related?

Uh, the yesterday as a part of the beginning, uh,

example, I explained you about

separating only class reviews after that.

In the second example we discussed about

distributing class reviews into separate file,

non-class into separate file.

Okay, so

before directly start working with this,

what are the file you want to work with That?

First of all, you have to open the file.

When you open the file, file up that is created.

Let's say you file equal to open off.

What is the path of that file? So what is the current path?

The current folder, you need to check it. Simply say pw.

Present to working directory.

This is a line x command.

So under route content is the folder under

content, your file is copied.

Okay, I'm going to open that file.

So for your notes purpose, I copy that. Uh, text also here.

So before start working first to create a file

with this data, then start doing that.

So your file name is common dot

D and this is the data.

I'm going to open that file in file.

Equal open is the function

and give the path under

that error file is common.

Dt we have to specify opening mode.

What are different modes?

You have the right up and still other modes are there.

First we work with read.

If you don't see any opening mode by default is read mode.

Okay? Now

whenever I open the file record pointer will be on the first

line record pointer will be on first line.

Suppose I'm saying that in file dot, okay, AP juster

run it in the next cell.

I'm saying in file. Read line, singular function.

Read lines by say all lines will be read.

Now where is a record point?

A record point is on the first line.

So that only first line will be read. Just check this.

Got it. Clause is good. That is a first line.

Once the first line is read the record

pointer goes to second line.

Now current, current record pointer position is second.

Once again, if I say in file read line, it'll read

that second line clause is boring.

That is second line. One more.

Now where is the record pointer? No.

Third, whenever you open that is on the first.

After reading the first line. Record point is on second.

Now that is it. Third.

Now observe here in file.read line again.

Clause is lagging. That is third line.

Now where is the record point fourth in file

read lines this time.

Now current position is fourth, right?

Fourth onwards all the remaining rules.

Fourth onwards all the remaining rules. Just check it.

Total how many lines in the data, right? Hmm.

Eight lines in the data already.

First three lines are right.

The remaining file lines are, so

that means fourth line onwards.

All the remaining lines. I'm closing the

file in file.

Do close

file got closed.

If file is closed, once again,

if I say in file.read line, what happens?

I got that because file is already closed.

If file is closed, we cannot perform any IO operations.

IO means read and write operations. Understood.

Okay, again, to work with that file, what we need to do,

we have to open that file.

Come on, open it. Tell me the statement

in file equal two.

Open up content.

Comment do next.

Now file is open. If file is open,

where will be the record point on the first line?

Now if I say in file read lines, what happen

first and on what all the remaining data is.

Now there is a record pointer.

Where is a record pointer Now

record point is on not eight lane.

Eighth next, which is empty already. Eight lanes completed.

Now record point is on the last end of the file.

At the end of the file there is no data.

That's why if I say once again in file.read line,

what I get, I got blank

or else in file do read lines.

I got MT list.

Now there's no

data understand the point.

So this is the thing like how

to read data from a particular file.

All of you understand the difference

between read line and read lines.

Okay? Now I'm keeping all the data

into data.

Object in

file.read lines.

What is the data type of data List?

Data type is list. How many cards will be there into that?

Zero records. Why? Zero records.

The record point is on end of the file.

Record point is at end of the file, right?

That's why zero records

to start reading data from the first.

What you need to do, once again I'm closing.

If you'll reopen that the file will be closed again,

will be open automatically the path is like this

content under that file, name,

comment dot p.

Now file is red. Next onwards.

I don't want that because already my records are in debt.

Object. Understand

file got fixed.

Now check the data. What is the data type of it?

That is a list.

Check the data. You'll see it records here

At the end of each line.

Now what is there backslash

and is there the backslash and reason?

Yesterday I explained when you presenter backslash

and will be recorded, backslash and always a new line.

Understand. So look at the last for the last record,

there is no backslash.

Okay, so I want to write,

so here some comments are about class,

some comments are about infrastructure section.

I want to separate only class related comments.

Already I have read the data into data.

I'm writing a loop, but line in data

check whether this is class related comment or not.

If class in data,

not data line in the ation,

what is a line in the ation?

What is a line plus is good?

Uh, within that, within that line class.

What available or not? It is if available I want to print.

Print, okay.

You see, just for printing check only

class related comments, right?

Understand. Similarly, do one thing give its opposite.

Accept the class remaining things. I what is the change?

Not not in. Okay.

If that butler class word is not available,

then only the condition is true.

Now check this condition. AC is hot.

Cool is not enough like that. Understood.

Now what I wanted to do is I want

to write only class related comments into one file.

Whenever you want to write something into your file, first

of all you have to open that,

but I want to write right with the right mode.

I want, I need to open what

is the option for the right mode?

W Uh, let's say out file equal.

Open off the path.

I'm giving content.

Mm, let's say plus

reviews.deals.

That is my file. Okay?

W for writing next time, writing a loop for data,

for line

data for line in data.

Check whether class available in that, in that or not.

What is the statement If class

in, if available I want to write it in stock printing.

I want to write it. What is the file object

dot, right?

If I execute this, what happens just

after the right is the function

to write a record into your into your file,

but you don't see any data here, check it.

Plus review file created, but there is no data.

Where is data? Actually data is in ram.

If you close the file

that only data will be flushed from ram to disk.

Understand so that

before, once your were completed,

after completion of the far out file

dot,

now check the same class

reviews previously empty.

Now data is that only class related comments are available.

Similarly, a small extension to the table.

Once again, I'm repeating entire process.

The out file, class file

inflow file two files.

I'm taking all class related comments into one file.

Inflow related comments into other file. In both files.

I want to write that data so that two output files you have

to open, open off

content under that class.

Reviews, not list.

I want to write into this, but already

with this name already file is there what will happen?

Our rights what, right?

We have that means previous data will be lost.

First data will be written.

Do the same thing for infra

file content.

I say infra

means here also.

Now write a loop for line in right?

Tell me sir,

if the word class in

always better practice, convert each line

into lawyer

because some class might be in upper case,

some class might be in lawyer case, right?

That's why convert all the things into lawyer case first.

If class in line, definitely that is about class comment.

Then class file dot

write, write off.

Yes. If the condition falls with definitely

that is not about the class.

Hmm. Then infra file dot write.

Write off.

Hmm, your work is completed.

You have to close up both the files.

Close the first file plus file dot close.

Come on second. Infra file dot close

plus and

plus running this.

Now just check what happens.

Both are done.

Class reviews, XT

and infra reviews eht within the class reviews only.

Class comments, same thing.

Check for infra, only infra comments

Based on this idea, I give you one task.

Now you tell me I will implement.

Okay, you get this one.

How did this details

about here at each side,

how many commas each field is separated with?

Comma. The how many commas are linked?

Four means how many fields? Five fields already told you.

If there are young delimiters, young plus one fields

sir and getting my voice, you are not audible to me.

Someone please speak. Say

We can hear you sir. Yes.

Yes. You are audible sir. Okay. Correct.

So there are five fields that two,

the structure is like this.

Id name salary, gender, and department number.

Probably five years. Sir, if you look into the gender,

there are two genders, males, females.

I want to separate this data as males

and females, all males into one file.

All females into other one. This is a small task.

First I keep this into text.

Uploading e mp one dot text.

Done

file is MP one.

This is the data.

My task is like this.

They should put all males into males xt

females into females.

First I need to read the

employee EMP file.

Mm, open up path.

EMP one pt. I'm not giving any opening mode.

Read.

Go on, read data from it.

That it P file.

Do

read again.

The next one. What? I don't want e emp file dot because,

because the reading is already complete,

I want to write it into two separate files.

Males and females.

Open content.

That's a males dot based

write two.

Same thing for females

content males do.

Yes, right?

Start trading data from this.

Check each record whether it is a male or female.

If male write to males file.

Otherwise females file tell you

Line in what is sample

row 1, 0 1.

Gender

to check whether it is a male or female.

If, can I say him in

for example, this is fu name is Amal.

If I check like this, ya available in or not

The letter ya available in or not?

Yes. And then what?

Uh, what it reads as a male,

but in fact she's a female, right?

You should not check like this.

How to split it first.

W is equal, equal to line dot strip.

What strip is doing. Mm-hmm.

Of course in this we are concentrating only on

the agenda Don of strip.

Because if backslash N is there, that will be

for the department number, right?

Hmm. Okay. Lawyer. Why lawyer?

Some genders might be in upper,

some genders might be in lawyer base.

That's why.lawyer do

split spliting based Done how many commas?

Four commas means how many elements in the list? Five.

Five elements in that file. Last second one is gender.

Separate that gender W off

minus gender separated.

Now check this. If G equals to Y

understood If G equals to Y means suppose this is the record

that is separated condition to

or false G value actually or false.

False. Then Y spot will

suppose if the given condition true, true that is male, then

males dot right write off.

Right? Yes.

Males, sorry.

Females dot right of the,

once every work is completed, males are closed.

Hand slot object. Check this,

check the data.

What records are here? Males file. Females file.

Look into the males. Only males are here.

Five males. These other

males look at females.

These are females, okay? Have distributed them.

Video: Joining of Files

Which is DE pt

within this 11, 12 30 in 2021, my employees,

we are considering on 11, 12 30.

Okay, this is the data.

So these two files, I'm uploading

first for practice purpose.

Let me keep this data into our notes.

Around 90 are following. The past file is profile tx.

This is the data for first line is header input file two.

What is that? Input file two, dpt, tx.

But here, no header. What are the fields here?

Id the department number, department name, location

manager id.

So here no head.

The schema is like this.

Department number, department name, location manager.

Right? Sorry.

The second file data will copy.

Task, join two files

based on department number call

the expected output skill.

I last name self gender,

department number, department location.

So in a single file I want to write a complete information,

previously information available in two separate files.

Now I want in single file,

I just Ed to open the file.

Let's say e file,

open off.

And our collab, our path is content

from that content file.

Yes, it's in the reading.

Same style department file

content or

in the first file there is a header

in second or header.

I want to separate that. Uh, header E had equal

E filed on read line.

Our record point is on second E data Equal

EFI read lines only the data part will be read

next in the file.

There is no data. That's why D data equal.

There is no header here.

Defi read lines next onwards.

I don't want those input files straight away. Can close.

Now check the data. What happens

will upload those two files

silent.

What are the first five profiles?

So these are the topics. Check that data.

Feed data

only employee spot separated D data.

The departmental information separated.

And what is the joining column for department number

in which that is unique in the D data.

So first I need to keep this into textile

dvo equal to dictionary

and start working with the D data.

For DD data.

First I'm spreading the information.

D nvo equal D

split

apply.

I get what list? First one is department number

I to now check what happens from each line only department

number will be separated

to print.

DM only,

department number separated do not.

But I need remaining information also.

That's why W equal to

strict lawyer split.

This is always better practice.

Now W is the list what fields you have in the list.

Dm O being location manager, right?

These are the fields that ITing the DO,

Zeroing the department information,

remaining information.

Let's say info equaled

W one quota.

All the remaining information is into D.

So this department number I make, I, I will make it

as a key info as well.

D info of D equals.

Now check this. Well,

okay, 11.

Marketing, 11 as P and that it's details says value.

Similarly. 12. 12 details. 13. 13 details.

How many departments we have for all the departments you get

Now in which file, the department number is duplicated.

That is employee understood.

I'm going to read the data from employer already have read

the data into e data.

I'm going to start reading the data

for lining E data here.

What is the schema?

This is the schema. So here

I want this information.

I want this department number information.

But in the department number in the last

backslash is upgrade.

I don't want that because I want to con coordinate.

That's why W is equal.

Now totally five fields are here. Post five fields.

I'm treating as info.

Okay, just keep the WSW lettering,

the department number W.

So for the first regard, department number values is level.

I want to get level information.

I will pass to the previous that

said DEPT, info

dvo involved

department, complete employee details.

R in W department. Information available in this.

This one here. What fields are available?

Five fields are available here. What fields are available?

Three fields. Department name, location manager. Right?

These three things are available.

I'm going to con coordinate them.

Info equal to w plus DEPT info.

DE PT info here. How many fields? Five fields here.

Three fields total.

Eight fields, but all fields are now string only.

I want to join them as a single link up to now.

Just check what happens. Print info check the red.

Complete information is available in single phase

or not previously.

That is into two separate files.

Now everything is into single.

The complete information

of each ly is available in a single drawer.

Now comment. Anusha one. Anusha one. Like 10,000 female.

Paul following hr, working at Delhi, his manager.

Yeah, of course information might be available

in other file.

Understand the point. Now I want

to write this information to the file.

That's why we has to, before writing the data into the file,

first we need to convert into string line.

Already in our last two sessions we have seen how to con

outline equal to dot join.

Join off in At the end of the line, I need

back, because of writing one line,

it has to come to the next line.

That's why backslash. In the last, just

before writing, confirm the output print

outline previously listed.

Now it is a string line or not. Each line is a string line.

Understood. Now I can start away.

Write this information to the file.

Instead of printing, I will write it.

Output file out. File equal to

under that given an name

E-M-P-B-B.

Because both files you are joining combined.

I want to write, I need a header.

I need a header equal to

id name, salary, gender, department number,

department name, location manager.

Right after that, back

before uh, loop.

I will write this letter into the file

out file dot required.

Write off it

now within the loop instead of printing,

I'm writing the same outline.

In the outline. I'll say in the last backslash.

Is that

out?

Understand Then

after accomplish of your work, you have

to close the file out.

File dot close.

Let's do this Implementation

here in a fresh cell.

I do it. Data is written.

Just confirm this. What is the file name You have E-M-P-D-P.

Complete information available single file account.

Okay, previously in two files.

Now one file.

Next, developers directly will use this. Understood.