

Tell me about Yourself?

Thank you given me this opportunity introduce myself, my name is Puspendu, i basically from west Bengal, I have Done my MCA New Horizon College of engineering, Bangalore.

I am a Python developer and the main work in my previous organization totally related to back-end and format-end technology.

I have a good knowledge of backend technology like Python, Django database like Postgress, MySQL, SQLite. Apart from this i have good knowledge of Frontend technology as will, like HTML, CSS, Javasaipt, jQuery, Bootstrap, React.

My Previous Company was Aakins Global Pvt. Ltd One of the start-up Company.

I am Currently Working as a freelancer developing web page for ANNA PRITHIBI SEEDS PVT. LTD

I have total experience in this field 2 year.

That's all about me

Why you live your pervious company?

I had to leave my previous organization due to a family emergency. Basically, my father was ill. However, everything is fine now and I'm very excited get back to work and focus on my career.

Though I've learned a lot of my previous company, from my research about this opportunity, I can see this is better fit for where I want to take my career.

Do you have any offer with you?

Yes, i have offer with one company, but i feel the position they offered is not want i am looking for, so, i am searching for other job.

What is your Expected Salary?

Over the year i really upskilled myself. I believe based on market standards for this Particular role and my experience and skill Set a salary range between 6 to 6.5 lacks Would be fair and appropriate.

why should we hire you?

Honestly, I have all the skills and experience that you are looking for, I'm Pretty Confident that I am the best Candidate for this job role. It's not Just my background in the past Projects but also my people skills which will be applicable in this position.

Where do you see yourself from five years now?

I see myself as a knowledgeable Professional having a in-depth knowledge of the Company and the industry and to be in a Senior position of the Company.

Are you a good team player?

yes, I enjoy working in a team environment, I believe only team work will make more Productive.

What do you know about our Company?

I know that Your Company is one at the longest Company Country in this industry, your headquarter Ranchi, Jharkhand and Founded 2019, there are 11-50 employees working as per your website. That's why i was excited to apply for this position.

What is your greatest Professional Achievement?

Basically, i have to work web-related work, But my Project i need to work other technologies also means app related work. So, i have to Spent some extra time to learn about this technology.

Then i was able to work independelly Other technologies and got appreciate by team lead.

Tell me about one stressful situation you faced in your Project.

I once had two large projects in the same week, and that was under lot of pressure.

However, because i create a schedule that detailed how i would break down each Project into Small assignment manged to complete all two Projects ahead time and avoid Unnecessary stress.

Tell me about your roles and responsibilities of your current work?

We will be assigned works based on agile methodology, my daily job roles and responsibilities will be I contribute to requirement gathering, analysis, implementing, design of new requirements and bug fixe.

I have to handle backend and fronted part.

Tell me about your work process?

My Previous Company was Aakins Global Pvt. Ltd One of the start-up Company. We have total development team 4 involved in our project, i have to worked basically Frontend and backend, other two development team member basically worked API and mobile app. If we are face any issue means someware stack our senior developer help us, and After complet task we have to push code on GitHub, And also we have to update in excel sheet for testing purposes. Then testing team performs the test, if testing team face any issues like found any

bugs they update bug list and we have to solved bug and update status complete, like this waya we have to work our previous company.

Tell me about one stressful situation you faced in your project?

or

Tell me about one challenges?

My Previous Company was Aakins Global Pvt. Ltd One of the start-up Company. We have total development team 4 involved in our project,

i have to worked basically Frontend and backend, other two development team member basically worked API and mobile app. However one of our team members who worked on mobile app, he left the company for some reason, For this I need to work on the app, so, that time i was feeling a bit stressed, But i don't want to miss this opportunity. So i have take up the extra responsibility and I managed everything and avoid unnecessary stress, this is one of the challenges i faced.

Tell me about your project?

I worked one of the client Project Tomes, it is Web and app based application, web part developed using Django framework that is online based application and app part developed using react Native that is online and off-line based application.

Our Project basically users Can Work Online and offline mode, even if there is no internet any place user can work our project.

The project focus easy and reliable Platform to manage different domain electrical, mechanical, Construction sector.

This application Consists of bunch of module Like- signup, signin, Forget Password, signout; Welcome Screen, Question, submodule ,Test Plan, Document and so on.

Super admin make specify of how many admin, manager, planner, tactician work in our project. App application only access technician, planner and manager.

In our Project one of the most important module test plan

Before going to test plan we have to perform Question, Submodule, form this three module fully based on test plan.

Admin can create question like form structure

Submodule basically collection of question, Admin cam mapped multiple question here And

Form basically collection of Submodule.

After that admin mapped final from into the test plan also assign which technician, which planner can work.

In app we have a sync button that is basically used to get all data from to the server.

Do you have any questions for me?

Can you please share the day to day responsibility for this role?

Git --version (Check git version), Git config --global user.name "Your name", Git config --global user.email "email_id", Git init, git status, git add <file name> (Convert Untrack file to track file), git commit -m "message of commit" (commit), git add . (all file staging), git branch (show all branch), git branch dev (create branch), git PUSH, git PUL, git clone <Git hub link>, (dev)>git merge anuj (merge branch), vim <fileName>.....eq

1. Pip install django
2. Python -m Django --version
3. Pip freeze

4. Django -admin startproject sahu
5. Python manage.py startapp sampa
6. Pip freeze

Model Inheritance

- * Abstract base class
- * Multi-table Inheritance
- * Proxt model Inheritance

7. Python manage.py runserver (makemigrations, migrate, showmigrations, showmigrations
AppName, createsuperuser, inspectdb)
8. Python manage.py runserver 8080
9. Python manage.py runserver 0.0.0.0:8000
10. pip install virtualenv
11. virtualenv env
12. cd env/Scripts

DISWE (Message Framwork)

- **Cookie**---> request.set_cookie('name', 'puspendu', expires=datetime.UtcNow()+timedelta(days=2))
- Name = request.cookies.get('name')
- request.session[name]='sonam'
- request.session.get('name', default='puspendu')
- request.session.set_expire(10) -----> views.py
- session_cookie_age = 400---> settings.py

Session

- database-backed (Default)
- file-based (django.contrib.sessions.backends.file) (Local)
- cookie-based (django.contrib.sessions.backends.signed_cookies)
- cached



Directory Structure(project)

Manage.py, __init__.py, settings.py, urls.py, wsgi.py- This is an entry point for your application which is used by the web server to serve the project you have create.

Directory Structure(app)

__init__.py, admin.py, app.py, migrations, models.py, test.py- App unit test automation class include here.

Generic Class-Based Views / Generic View

Display View – ListView, DetailView Editing View – FormView, CreateView, UpdateView, DeleteView

Signals – Two improtent parameter in signal sender and resever, Django provide many built-in signal dispatchers like -- pre-init, post-init – send before or after __init__() method is called pre-save, post-save, pre-delete, post-delete, request-start, request-finishd

DjangoAdmin-

version, runserver, startapp, startproject, createsuperuser, makemigrations, migrate,

Exception- The Django.core.exceptions module contains the some class EmptyResultSet, fieldDoesNotExist, viewDoesNotExist, MiddlewareNotUsed, Fielderror, validateError

Select_related & prefetch_related- Both of the function are used to fetch the data. In simple word **select_related** used a foreign key relationship using join the query. **Prefetch_related** there is a separate lookup and the joining on the python side.

- 1.create AWS EC2 instance
- 2.configure security group
- 3.Connect EC2 instance
- 4.Update ubuntu
- 5.Install python & Nginx
- 6.clone project
- 7.Create virtualEnv then install package from requirement.txt file
- 8.Install Django & gunicorn
- 9.setting Django project
- 10.Deactive virtual env
- 11.create socket file for gunicorn
- 12.enable gunicorn socket
- 13.create socket file for nginx
- 14.enable nginx
15. restart

Git --version (Check git version), Git config --global user.name "Your name", Git config --global user.email "email_id", Git init, git status, git add <file name> (Convert Untrack file to track file), git commit -m "message of commit" (commit), git add . (all file staging), git branch (show all branch), git branch dev (create branch), git PUSH, git PUL, git clone <Git hub link>, (dev)>git merge anuj (merge branch), vim <fileName>.....eq

UserModel

Custom UserModel--> 1.AbstractUser 2.AbstractBaseUser

```
from django.contrib.auth.models import User
```

```
fields = ['username', 'first_name', 'last_name', 'email']
```

```
class Customer(models.Model):
```

```
    user = models.ForeignKey(User, on_delete=models.CASCADE, related_name='mypage')
```

```
    comment=models.CharField(default='not available', max_length=40)
```

```
    students = CustomManager()
```

```
    def __str__(self):
```

```
        return str(self.id)
```

```
stu=Student.objects.order_by('name').first()
last()
latest('pass_date')
earliest('pass_date')
```

ModelForm

Header payload signature

```
from django import forms
```

```
from .models import User
```

```
class StudentRegistration(forms.ModelForm):
```

```
    class Meta:
```

```
        model = User
```

```
        fields = ['name', 'email', 'password']
```

```
        labels = {'name':'Enter Name', 'password':'Enter Password'}
```

```
        # fields = '__all__'
```

```
        exclude = ['name']
```

```
filter(name__exact='Sonam') #ClseSensativfe
filter(name__iexact='sonam') #In clseSensativfe
filter(name__contains='u')
filter(name__icontains='U')
filter(id__in = [1, 5, 7])
filter(marks__gt = 60)
filter(marks__gte = 60)
filter(marks__lt = 60)
filter(marks__lte = 60)
filter(name__startswith='S')
filter(name__istartswith='s')
filter(name__endswith='L')
filter(name__iendswith='L')
filter(passdate__range=('2020-04-01', '2020-05-03'))
filter(passdate__year=2019)
```

per Site(settings.py)

```
CACHE_MIDDLEWARE_SECONDS = 30
```

per View(viewes.py)

```
@cache_page(60)
```

template fagment(temp.html)

```
{% cache 30 course %}
```

```
<h1>Course 3</h1>
```

```
<h1>Course 4</h1>
```

```
{% endcache course %}
```

```
{% include
"template_name.html"
%}
{% extends 'base.html'
%}
{% block title %}
Home
{% endblock %}
```

docker build -t ImageName ----> create image

docker run --name containerName imageName ----> create container

docker run --name containerName -d imageName ----> run container in detach mode

docker run --name containerName -itd imageName ----> run container in interactive mode

docker pull imageName--> Get image from docker hub

docker images----> show all images

docker ps----> show all container(only runing container)

docker ps -a ----> show all container(runing and not runing container)

docker stop container_name ----> stop container

docker start container_name ----> start container

docker restart container_name ----> restart container

docker rm container_name----> remove container

docker rmi image_name----> remove image

```
student_data = Student.objects.all()
average = student_data.aggregate(Avg('marks'))
Sum, Min, Max, Count
```

```
student_data = Student.objects.filter(Q(id=6) &
Q(roll=106))---->|
student_data = Student.objects.filter(~Q(id=6))
```

```
data3 =
```

```
Page.objects.filter(user__email='contact@geekys
hows.com')
```

Authantication View

```
path('accounts/', include('django.contrib.auth.urls'))
accounts/login/
logout/, password_change/, password_change/done/
password_reset/, password_reset/done/,
reset/<uidb64>/<token>/, reset/done/
```

API Status

1. Informational(100-199)
2. Success(200-299)
3. Redirection(300-399)
4. Client Error(400-499)
5. Server Error(500-599)

Open Command prompt -> Install psycopg2 -> connect postgres with python -> create cursor object -> Execute query

ACCESS_TOKEN_LIFETIME
REFRESH_TOKEN_LIFETIME

```
class StudentModelViewSet(viewsets.ModelViewSet):  
    queryset = Student.objects.all()  
    serializer_class = StudentSerializer
```

API Views(Function Based)

Get,Post,Put,Patch,delete

```
from rest_framework.views import APIView  
class StudentAPI(APIView):  
    def get(self, request, pk=None, format=None):
```

Concrete GenericAPIView(----)

ListAPIView
createAPIView
retrieveAPIView
updateAPIView
DestroyAPIView
LC, RU, RD, RUD

```
from rest_framework.generics import  
ListAPIView  
class StudentList(ListAPIView):  
    queryset = Student.objects.all()  
    serializer_class = StudentSerializer
```

Permission

Project level

1. AllowAny
2. IsAuthenticated
3. IsAdminUser
4. IsAuthenticatedOrReadOnly
5. DjangoModelPermissions
6. DjangoModelPermissionsOrAnonReadOnly

```
from rest_framework.authentication import  
BasicAuthentication  
authentication_classes=[BasicAuthentication]
```

viewsets(class Based)

list(),retrieve(),create(),update(),partial_update(),
destroy()

```
from rest_framework import viewsets  
class StudentViewSet(viewsets.ViewSet):  
    def list(self, request):
```

GenericAPIView and ModelMixin

ListModelMixin
createModelMixin
retrieveModelMixin
updateModelMixin
DestroyModelMixin

```
from rest_framework.mixins import ListModelMixin  
class StudentList(GenericAPIView, ListModelMixin):  
    queryset = Student.objects.all()  
    serializer_class = StudentSerializer
```

```
def get(self, request, *args, **kwargs):  
    return self.list(request, *args, **kwargs)
```

Authentication

1. BasicAuthentication
 2. SessionAuthentication
 3. TokenAuthentication
 4. RemoteUserAuthentication
- ```
from rest_framework.permissions import
IsAuthenticated
permission_classes=[IsAuthenticated]
```

## List

```
Find cubes using list comprehension
```

[https://docs.google.com/document/d/1ewFNld5APkSec\\_0hc3DwctO\\_wYJY9atpBx-Mlt4baxM/edit?usp=sharing](https://docs.google.com/document/d/1ewFNld5APkSec_0hc3DwctO_wYJY9atpBx-Mlt4baxM/edit?usp=sharing)

```
sort list with out using sort function
```

<https://docs.google.com/document/d/1MrHUFFJAmd1DybUULmcxNZT6hauuxMgmwa-TbZe4FFQ/edit?usp=sharing>

## Dictionary

```
Reverse Dictionary Keys Order
```

[https://docs.google.com/document/d/1b\\_19gER5v4pTNDJ4U\\_lxYA5y8HBb\\_MZXh3w4nf3VCMw/edit?usp=sharing](https://docs.google.com/document/d/1b_19gER5v4pTNDJ4U_lxYA5y8HBb_MZXh3w4nf3VCMw/edit?usp=sharing)

```
Sort by dictionary(Convert tuple to dictionary)
```

<https://docs.google.com/document/d/1vqIXw7676cW79A9RupJoLLClp4u56NU6CfkJ87asYcc/edit?usp=sharing>

## Generator And Iterator:

```
Fibonacci using Ganarator
```

<https://docs.google.com/document/d/1IYwoolRmDj2VNmjpQENIAsC-ly0mLnybuFCiBYPxel/edit?usp=sharing>

## File:

```
with, read, write, read & write
```

[https://docs.google.com/document/d/1S7siGx3Vk\\_RftxrURVDJ6lrJo04lxxgdrFSJAqRUvew/edit?usp=sharing](https://docs.google.com/document/d/1S7siGx3Vk_RftxrURVDJ6lrJo04lxxgdrFSJAqRUvew/edit?usp=sharing)



```
copy line from test.text to
test1.text file
```

<https://docs.google.com/document/d/1j0wx2kWt0EEfAXNBTCQPYezjGzFL-yAdAnbgqxH7Kg/edit?usp=sharing>

## Recursion

```
ReversList using recursion
```

<https://docs.google.com/document/d/1C9kn8N9T-n4scdl8-1NmG-klJpPsZJRdTH5IIAQrjp8/edit?usp=sharing>

```
recursion with fibonacci
serise
```

[https://docs.google.com/document/d/1RL316VVS2jG5itDjZqc1K\\_O8PxEPzogAFa4Ccc6A58Y/edit?usp=sharing](https://docs.google.com/document/d/1RL316VVS2jG5itDjZqc1K_O8PxEPzogAFa4Ccc6A58Y/edit?usp=sharing)

## Palindrome or not

```
palindrop or not
```

[https://docs.google.com/document/d/1V4\\_zco0--IRPqZKoh3n34JYPIGnUqiA7tDMs6S\\_drD0/edit?usp=sharing](https://docs.google.com/document/d/1V4_zco0--IRPqZKoh3n34JYPIGnUqiA7tDMs6S_drD0/edit?usp=sharing)

```
Sting pllindrom
```

[https://docs.google.com/document/d/1Q\\_z54HkJL\\_7Ryi4VeTkLt3KpKOFecIButxCos4Q74MQ/edit?usp=sharing](https://docs.google.com/document/d/1Q_z54HkJL_7Ryi4VeTkLt3KpKOFecIButxCos4Q74MQ/edit?usp=sharing)

## Sort

```
people sort
```

<https://docs.google.com/document/d/1wcKHfpAJNi3GjyTbc6CJ8OHSMvLE-QsqQ7KQJBOjQM8/edit?usp=sharing>

## Find

```
Find Missng Number
```

[https://docs.google.com/document/d/1jdFFUJFSsre\\_NlYvD4pbC49uoBSaHfxWTzSAhNveEMA/edit?usp=sharing](https://docs.google.com/document/d/1jdFFUJFSsre_NlYvD4pbC49uoBSaHfxWTzSAhNveEMA/edit?usp=sharing)

## # Find element from nested lists

[https://docs.google.com/document/d/1ZzS0QLyDt-l3f6WnN6auj1ftM\\_EUPHZd4p0L0sJoJlg/edit?usp=sharing](https://docs.google.com/document/d/1ZzS0QLyDt-l3f6WnN6auj1ftM_EUPHZd4p0L0sJoJlg/edit?usp=sharing)

## # Find even number from list using lambda function

[https://docs.google.com/document/d/1\\_IP4avqTwRd7-JHsEE9ghXVQTqTG93MAe2c1IsjWakk/edit?usp=sharing](https://docs.google.com/document/d/1_IP4avqTwRd7-JHsEE9ghXVQTqTG93MAe2c1IsjWakk/edit?usp=sharing)

## # Find odd number

[https://docs.google.com/document/d/102EjSR\\_oBrrahyLR4UEDZm9KmYzjPjbPWE9XdrcYsm8/edit?usp=sharing](https://docs.google.com/document/d/102EjSR_oBrrahyLR4UEDZm9KmYzjPjbPWE9XdrcYsm8/edit?usp=sharing)

## Program

## # prime number program

[https://docs.google.com/document/d/1Vx9PmCMHCghDWXVVPPhWJ4DOrOgtbn\\_Po4xXzvKmX1rw/edit?usp=sharing](https://docs.google.com/document/d/1Vx9PmCMHCghDWXVVPPhWJ4DOrOgtbn_Po4xXzvKmX1rw/edit?usp=sharing)

## # Armstrong or not

<https://docs.google.com/document/d/1U80Bal86ESgLzKXaZcJ8uYxii0rr8FE6qM5Ss-7KO0A/edit?usp=sharing>

## # Factorial number

[https://docs.google.com/document/d/12pDCriNwk9epDa\\_seRZ60a8jkQtNplYyqamwnvJ5FtU/edit?usp=sharing](https://docs.google.com/document/d/12pDCriNwk9epDa_seRZ60a8jkQtNplYyqamwnvJ5FtU/edit?usp=sharing)

## # Fibonacci serise

[https://docs.google.com/document/d/12GGjsdPo1561j-mp1QpEn-A2Vp\\_sSnCvPWRIncVWELA/edit?usp=sharing](https://docs.google.com/document/d/12GGjsdPo1561j-mp1QpEn-A2Vp_sSnCvPWRIncVWELA/edit?usp=sharing)

# Define a function that will take two number and an operator as a argument and evaluate it's result

<https://docs.google.com/document/d/1i5myZIUZNWfnbTGutKstwj42vn8D8zT89kKCL7mEWt8/edit?usp=sharing>

# concatenate tuple

[https://docs.google.com/document/d/1\\_gzquT0xhBl\\_YULthFbDKv9W8woJ6zrq4GI3XwHFP6k/edit?usp=sharing](https://docs.google.com/document/d/1_gzquT0xhBl_YULthFbDKv9W8woJ6zrq4GI3XwHFP6k/edit?usp=sharing)

# multiplication Without operator

<https://docs.google.com/document/d/1qfOPNBA9uleaFvkH80Wp64ABo42toPlj0W-bZolu7ds/edit?usp=sharing>

# addition of the other two objects.

[https://docs.google.com/document/d/1DELcBhwrVZ3L1fidXpR\\_lqgy-IC49fH74WS8HgXtRw0/edit?usp=sharing](https://docs.google.com/document/d/1DELcBhwrVZ3L1fidXpR_lqgy-IC49fH74WS8HgXtRw0/edit?usp=sharing)

# fizzBuzz

<https://docs.google.com/document/d/1awQsESTE-uRywyNsOrFfhQpmOIm2Cy3NvYyAYsjK674/edit?usp=sharing>

# How to combine two data frams in python using pandas

[https://docs.google.com/document/d/1iiboHblpzWRh3nVNt8iYk2ZFw9ACTBhJwqyj8\\_sEVzA/edit?usp=sharing](https://docs.google.com/document/d/1iiboHblpzWRh3nVNt8iYk2ZFw9ACTBhJwqyj8_sEVzA/edit?usp=sharing)

# String

```
'the quick brown fox jumps over the lazy dog'
Count number of 'o'
```

[https://docs.google.com/document/d/1aqPyw59mYUelnUmTgl-\\_iies\\_lX6-6hIKwvvqvqwrG/edit?usp=sharing](https://docs.google.com/document/d/1aqPyw59mYUelnUmTgl-_iies_lX6-6hIKwvvqvqwrG/edit?usp=sharing)

```
"Reverse string data"
data string Reverse
```

[https://docs.google.com/document/d/1gFN-E3aj9lI33rjRhAlshA4qEPkXRVGT2dh8-t6K\\_Q8/edit?usp=sharing](https://docs.google.com/document/d/1gFN-E3aj9lI33rjRhAlshA4qEPkXRVGT2dh8-t6K_Q8/edit?usp=sharing)

```
['12345786', '096265827', 'ADRGJUUIKKMHG',
'12584Ugthj', '12452696k']
['ADRGJUUIKKMHG', '12584Ugthj',
'12452696k']
```

<https://docs.google.com/document/d/13ecJYOW8Fi04j5pLwcWJYGajh-y274xz8HP0YvojbZw/edit?usp=sharing>

```
lowest Repeating character in a string
"gdsfgus dhsfdshf dsfhviusd"
{ 'g': 2, 'd': 5, 's': 6, 'f': 4, 'u': 2, ' ': 2,
'h': 3, 'v': 1, 'i': 1}
s
```

[https://docs.google.com/document/d/1rd7jPAY8i-Ra1sZgxXEdGlogWA\\_ZIOq1RDNFfyHOtmE/edit?usp=sharing](https://docs.google.com/document/d/1rd7jPAY8i-Ra1sZgxXEdGlogWA_ZIOq1RDNFfyHOtmE/edit?usp=sharing)

```
"/ *apple are & found% only @red & green"
apple are found only red green
```

[https://docs.google.com/document/d/1cMC\\_3171JWqxhSMgSSzSM0poW-5TSipALW4E78HuBbl/edit?usp=sharing](https://docs.google.com/document/d/1cMC_3171JWqxhSMgSSzSM0poW-5TSipALW4E78HuBbl/edit?usp=sharing)

```
Modifi String Format:
```

```
Input = I_Am_A_Coder
Output = i.aM.a.cODER
Input = This_Is_A_Good_Morning
Output = tHIS_iS_a_gOOD_mORNING
```

```

Using List
def string_format(s):
 l = []
 temp = s.split('_')
 for i in temp:
 l.append(i[0].lower() + i[1:].upper())
 s = '.'.join(l)
 print(s)

s = "This_Is_A_Good_Morning"
string_format(s)

```

```

Compress String
aabbccddeeee = a2b2c2d2e4

```

```

def compress(s):
 n = len(s)
 new_str = ''
 count = 1
 for i in range(n-1):
 if s[i] == s[i+1]:
 count += 1
 else:
 new_str += (s[i] + str(count))
 count = 1
 new_str += (s[n-1] + str(count))
 return new_str

s = 'aabbccddeerrrrr'
print(compress(s))

```

```

input - 'hello, world, sahu'
output - 'sahu, world, hello'

```

```

str1 = 'hello,world,sahu'
s = str1.split(',')
revStr = s[::-1]
joinRevStr = ','.join(revStr)
print(joinRevStr)

printOutPut
str2 = 'helloworldsahu'

```

```
print("str2[5::]", str2[5::])
print("str2[:5:]", str2[:5:])
print("str2[::-1]", str2[::-1])
```

```
string='abcdef'
fibiSeries = [1,1,2,3,5,8]
output- 20a1b1c2d3e5f8
```

```
s1 = 'abcdef'
fibo = [1,1,2,3,5,8]
sum = 0
for i in fibo:
 sum += i
new_s = str(sum)
n = len(s1)
for i in range(n):
 new_s = new_s + str(s1[i]) + str(fibo[i])
print(new_s)
```

-----

-----

-----

-----

-----

-----

-----

## Decorator

### # Decorator example

<https://docs.google.com/document/d/1YQQ3C9tXWCpUZSC42Z5xyX4vWYwI3vxGvOuj-yEZkw8/edit?usp=sharing>

## List

### # List comprehension

<https://docs.google.com/document/d/1s7fTnN9gkYJiXmMnUiN370AfOOOVJPrclQtbWge-4m4/edit?usp=sharing>

## Dictionary

### # Dict Comprehension

[https://docs.google.com/document/d/1PBGqi\\_Hze96y8cr5PA4v2MJDBkoF6YpVqPzEQfnWC1g/edit?usp=sharing](https://docs.google.com/document/d/1PBGqi_Hze96y8cr5PA4v2MJDBkoF6YpVqPzEQfnWC1g/edit?usp=sharing)

## Generator And Iterator:

### # Generator Example

<https://docs.google.com/document/d/1Oo4BPZNjvYLLDR5fwxgjQkYDC4pYrdL0lIXmheGgyE4/edit?usp=sharing>

### # Iterator Example

<https://docs.google.com/document/d/1BUy3X2Mzxn51mQk1QnK40M7g34RvbPYwm2WW0rAg8WA/edit?usp=sharing>

## Class & Object:

### # Class, object and Constructor

<https://docs.google.com/document/d/1Ral2uhBuAiXNZr5ngQGApoJvbcS27zO5qf2FZUa-tM/edit?usp=sharing>

### # Abstract class and Abstract method

<https://docs.google.com/document/d/1aQHmmEpsgNP8vpTy0v-YC5qhkZ5EA3SW5atnYBhemRU/edit?usp=sharing>

## Inheritance:

### # Multiple Inheritance

[https://docs.google.com/document/d/1oYNn7Aw9z08Q5i5\\_5zexzWfSWiJqYD6-fKcQhwlMU5o/edit?usp=sharing](https://docs.google.com/document/d/1oYNn7Aw9z08Q5i5_5zexzWfSWiJqYD6-fKcQhwlMU5o/edit?usp=sharing)

### # MRO(Multiple Inheritance)

[https://docs.google.com/document/d/1Urfdl3U0uLpEFDUS1QaSzkAWwOZRw\\_M2KRoeojfL2w/edit?usp=sharing](https://docs.google.com/document/d/1Urfdl3U0uLpEFDUS1QaSzkAWwOZRw_M2KRoeojfL2w/edit?usp=sharing)

## \*args AND \*\*kwargs:

### # \*args Example && \*\*kwargs Example

[https://docs.google.com/document/d/1E0Io9Dgb\\_ksnIfFiliIbLRDQDaI8T8JGB3Hz5BRdACw/edit?usp=sharing](https://docs.google.com/document/d/1E0Io9Dgb_ksnIfFiliIbLRDQDaI8T8JGB3Hz5BRdACw/edit?usp=sharing)

## Exception:

### # Normal Exception

<https://docs.google.com/document/d/1Va0HyRN67eUqvZvbOPMAeVfweK-MVFjRgHADubYyLeA/edit?usp=sharing>

### # User define exception with list operation

[https://docs.google.com/document/d/1kbgvSrYZsoVslpslgXjL\\_DGDjH0Bcly-g4ashUS3A8/edit?usp=sharing](https://docs.google.com/document/d/1kbgvSrYZsoVslpslgXjL_DGDjH0Bcly-g4ashUS3A8/edit?usp=sharing)

### # map, filter, reduce

[https://docs.google.com/document/d/1HOMS\\_PsOw67N7oAQ6IYbflWEHnpYxBLAv2XJ2OQRbY/edit?usp=sharing](https://docs.google.com/document/d/1HOMS_PsOw67N7oAQ6IYbflWEHnpYxBLAv2XJ2OQRbY/edit?usp=sharing)

### # monkey patching concept



[https://docs.google.com/document/d/1D9\\_NDCfWklqZVN51HfnL1pseIUoM2AoK\\_RuPisMA7G8/edit?usp=sharing](https://docs.google.com/document/d/1D9_NDCfWklqZVN51HfnL1pseIUoM2AoK_RuPisMA7G8/edit?usp=sharing)

## # Call by value & Call by ref

<https://docs.google.com/document/d/1dkEmXbB-KjtMv-Sm4t2wMKzR3jXYA-o4nS0LnuPo7PM/edit?usp=sharing>

## # Deep Copy & Shallo Copy

<https://docs.google.com/document/d/1WkGZ9zGbqGmYKV94786z3QQGjVkSupjqmhmYWDWYoZY/edit?usp=sharing>

## # Instance Method & Class Method & static Method

[https://docs.google.com/document/d/1fs3qLlcjqxkWanN3HOD8oPb8q3onw\\_CX8JWricO2BF4/edit?usp=sharing](https://docs.google.com/document/d/1fs3qLlcjqxkWanN3HOD8oPb8q3onw_CX8JWricO2BF4/edit?usp=sharing)

## # Operator Overloading

[https://docs.google.com/document/d/1CuX3ttYH8xOie5jZgryyAiXPcXRW\\_9Tr-PfDgoyLvzo/edit?usp=sharing](https://docs.google.com/document/d/1CuX3ttYH8xOie5jZgryyAiXPcXRW_9Tr-PfDgoyLvzo/edit?usp=sharing)

## Array

```
let animalArray = ['dog','cat',10]
```

```
// Add data in array
```

```
animalArray.push("Fox");
for(let i=0; i<animalArray.length; i++){
 console.log(animalArray[i]);
}
```

## // for...of With array

```
let arr=[1,2,3];
for(let val of arr){
 console.log(val);
}
```

## // for...in with object

```
const person = {
 name: 'puspendu',
 age: 26
};
for(let key in person){
 console.log(person[key]);
}
```

## // forEach with array and object

```
let arr=[1,2,3];
arr.forEach(function(item){
 console.log(item);
});
```

## Object

```
const objPerson = {
 name: 'puspendu',
 age: 26
};
Object.values(objPerson).forEach(value
=> {
 console.log(value);
})
```

## // Named Function

```
function sum(a,b){
 return a+b;
};
console.log(sum(10,20));
```

## // Anonymous

```
let sum = function(a,b){
 return a+b;
};
console.log(sum(10,20));
```

## // Callback

```
function add(a,b){
 return a+b;
};
function mul(a,b){
 return a*b;
};
function calculate(x,y,operation){
 return operation(x,y);
}
console.log(calculate(10,20,add))
console.log(calculate(10,20,mul))
```

## // Arrow function

```
let sum = (a,b) => a+b;
console.log(sum(10,20))
```

## // HigherOrder

```
function hof(func){
 func();
}
hof(sayHello);

function sayHello(){
 console.log("Hello");
}
```

```
// event handling
click,mouseover,ke
ydown,keyup,submit
,focus,blur,change
,load
addEventListener('click', handler)
```

```
// promises---> pending, resolved, or rejected
const myPro = new Promise((resolve, reject)=>{
 // perform asynchronous operation
 // if successfull call resolve
 // if failed call reject
 const randomNum = Math.floor(Math.random()*10);
 if(randomNum<5){
 resolve(`Success! Random number: ${randomNum}`)
 }else{
 reject(`Error! Random number: ${randomNum}`)
 }
});
```

```
myPro
 .then((result) => {
 console.log(result);
 })
 .catch((error) => {
 console.log(error);
 })
```

## Array Destructuring

Extracting multiple values and stored in objects and arrays.

```
var a, b;
[a, b] = [5, 10];
console.log(a); // 5
console.log(b); // 10
```

UseContext(listed of used prop drilling)