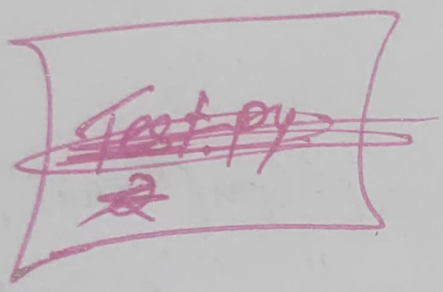
12th

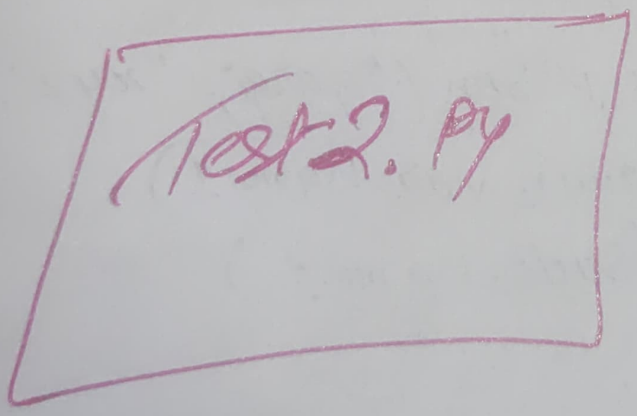
DAY

PYTHON

OOPS

PART-1


test.py
2


Test2.py

project

oops

main.py

test1.py

test2.py

main.py | test1.py | test2.py

```
class person:
    def __init__(sudh, name,
                 surname, emailID, " "):
        sudh.name1 = name
        sudh.surname = surname
          "          "
          "          "

anuj_var = person("anuj", "bhandari",
                  "     "     "
sudh = person("sudhanshu", "kumar",
gargi = person("gargi", "xyz", "
print(anuj_var.name1)
print(sudh.name1)
```

C:\Users\windio.

anuj

sudhanshu

gargi

Explanation :-

Here, instead of using 'self' what I will do is, I will create another 'python' file ~~.txt~~ which is test2.py

Same code copy paste here & instead of using 'self' I will use my name. 'sudh'. 1:16:00

*If we execute it, as a outcome we can see over here. It's working as it is.

*Note :-
whatever 1st variable that we are going to take that will behave as a pointer for any function that we are going to write.

def __init__ (sudh, name1

* we are writing a function, but we are writing end of number of parameters as requirement of the function. itself.

But, here we have to write additional parameter that will behave as a pointer.

that will point toward a class at any point of a time.

* __init__ what is the meaning of (init). ?

with the help of init, we are just try to pass a data. so that if I have to do something.

Code

```
* class person:
    def __init__(sudh, name, surname, email_id,
                 year_of_birth):
        sudh.name1 = name
        sudh.surname = surname
        "
        "
anuj_var = person("anuj", "bhandari",
                                    1994)
sudh = person(" ", " ")
gargi = person("gargi", "xyz",
print(sudh.name1 + sudh.surname)
print(anuj_var.name1)
print(sudh.name1)
print(gargi.name1)
print(type(sudh))
```

(right click)
Run code)
test2

test2 |
C:\Users\win10\anaconda1
Sudhanshu kumar
anuj
Sudhanshu
gargi

let's understand

Question :-

Can you please try to give me a concatenation of 'Name' and 'Sername' for 'Sudhanshu' for 'sudh' variable.

Can you please try to give me a concatenation of name and surname.

To this first of all, I have to access the data, once I will be able to access the data, then I will be able to do a concatenation.

Now, I will be able to access 'sudhanshu' 'name' and 'Sername'.

*If we run this code, As a outcome we can see below - (print(sudh.name1 + sudh.Surname)

Sudhanshu Kumar
anuj
Sudhanshu
gargi

How I got this outcome, simple or how I can access.

* ~~variable~~ Sudh.name1 + Sudh.surname)
  variable variable

These line code, I am able to do my concatenation operation. Obviously, it should be a string (str).

1st thing is access a data.

* Note :- Just for understanding

```
S = "Sudhanshu"
S1 = "Kumar"
print(S+S1)
```

outcome → "SudhanshuKumar"

```
V = "Vijay"
V1 = "gurung"
print(V+V1)
```

→ "Vijaygurung"

So, when we write a simple thing.

```
S = "Sudhanshu"
S1 = " kumar"
print (S+S1)
```

~~As a O~~ we have created basically, variable of a string (str) class.- 'S'
variable class of string - 'S1'.

Then, if we try to call both the variable
Then we will be able to get.
'Sudhanshu kumar'

Similar way, exactly we are trying to do same thing over here.
Someone has already written a 'class' and 'Object'.
This the only region we are able to create it.
But here in code, I am talking about my own custom instances. own custom 'class' custom 'variable' I am trying to create.
I am not talking about 'in build' one.

Code

```
class person:
    def __init__(sudh, name, surname, email, "
                                        " )
    sudh.name1 = name
    sudh.surname = surname
        "
        "
    sudh.year_of_birth = year_of_birth

def age(sudh, current_year): ~~year_of_birth):~~
    return current_year - sudh.year_of_birth
```

year_sudh

substraction sign (-) Negative (-)

```
anuj_var = person("anuj", "
sudh = person("sudhanshu", "kumar",
gargi = person("gargi", "xyz",
```

sudh.age(2022) → use this print(sudh.age(2022))

test.py
C:\Users\ >python

-21402

Let's understand:-

Now, there is a possibility that, I would like to create a generic function. to do something, may try to calculate age of the particular person.

may I would like to create a function ~~cal~~ with the help of that, I will be able to Calculate age of person.

~~Requirm~~ Requirement:- Try to create age of a person.

So, here, I will try to create a age my own custom function.

whenever, we try to create any function inside a class a very first variable will notify as a 'pointer.'

It is try to mention these entire this to the 'classes.'

To Calculate age, I need what To Calculate particular age, I need ~~basi~~ basically a 'current year' & 'year of birth'

'If I am going to do a substraction difference of it, it will give me a 'age'.

```
* def age(sudh, current_years):
      return current_years - sudh.year_of_birth
```

Can we say 'year of Birth' any how we are going to pass. at the time of creation of variable of the classes.

Any how 'year of Birth' we will going to pass.

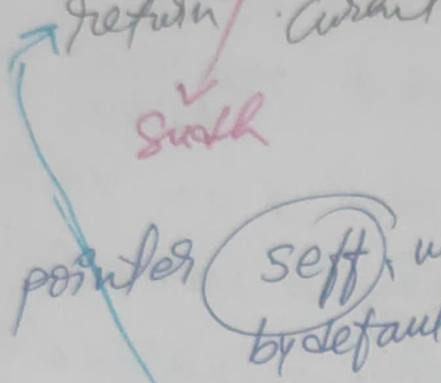
That will be [redacted] available inside a class, class is aware about 'year of birth' for these particular person. any how by default.

If I have to call class variable how, I will be able to [redacted] call a class variable.

Here also same changes made

def age (self, currat years):
    return currat years - self.year_of_birth

self → sudh



My pointer (self), we changed it as 'sudh'. by default but

My pointer 'self' by default, but I changed it made 'sudh'. So, pointer sudh. of years of birth.

So, now these 2 one code function, I have written inside 'class'

class person:

the only difference between a function that, we are writting so far so forth & these function or pointer (self or sudh) helping me out to call any thing from a "classes" & point anything to a class. This is only thing these pointer does.

* Because, I need some reference of Something, which will be allocate something to my class.es.

So, these * current year & ~~yes~~ year of birth class, which I have created I am trying to pass a data by creating a variable So at the time of creation of variable.

I am going to pass a data these is a function.

Now, anything that we have to called from the class whether its a variable that we are able to call or its a function that I would like to access.

Now, I will be access, can I say that I will be access it with the help of 'class variable' & 'objects', I will be able to call it.

But, how I will be able to access 'name'.

Can, I say that (anni_var.name1)
or
(variable.name1)
" : surname
" : year_of_birth.
etc.

This is how, I am able to call it.

Similar way,

If we have to call even a function how we will be able to call a function.

Let's suppose,
I am trying to call 'Age'!
Age for whom we would like to call.

24:35

for
* Anni,
* Sudhanshu or
* gargi

For which person, would like to get or understand the "Age"!
I can get the Age for all three (3) or get a Age for anyone.

* ① Anuj_var = person ("anuj", "bhandari", ,,,)
② Sudh = person ("sudhanshu", "kumar", ,,,,)
③ gargi = person ("gargi", "xyz", ,,,,)

~~Sudh.age(2022)~~
~~print(sudh.age(2022))~~ or ④ print(sudh.age(2022))

test2.py
C:\\\

just for understanding

-21402

use this code & Run the code test 2 right click

Now, here, I have to call a 'Age', let's suppose for 'Sudh' we would like to call a 'Age'
what we will do.

Sudh.age
↓
variable

So, 'Age' wise looking for a parameter, it is asking that, Can you please try to provide current year.

So, the current year is what it's a 2022, that I am going to pass.

Now once we are going to call these function. It is going to pass current year is 2022.

(Sudh.age(2022) will get from where. From here, it will get that.

* def .age(sudh, current_year):
    return .current_year - sudh.year_of_birth

* Because, Anyhow, I am trying to pass year_of_Birth. whatever 'class variable' if I have to call within our class, I have to use a pointer. whatever name of a pointer will going to define, by default we will be able to find out 'self' everywhere.

If we execute this code, As a outcome we can see, I am able to get the result.

outcome - -21402

* Code

```
class person:
    def __init__(self, name, ...):
        ...
    def age(self, current_year):
        ...

anuj_var = person(
sudh = person
gargi = person
print(sudh.age(2022))
print(anuj_var)
```

print(anuj_var.age(2022))

Now change here. 'sudh' Make it 'Anuj' then we will get Anuj Age.

test 2

C:\Users\Win10

28

Let's understand

Here,

Let's Suppose, I want to know the Age of 'Anuj'.

So, below I will make change.

```
* anuj_var = person("anuj", "                    )
  Sudh. = person ("sudhanshu"  "  . . .  )
  gargi = person ("gargi", "xyz", . . . . . )
  print(anuj_var. age(2022))
```

tgt2:
C:\Users

28

So, here in the last line code, we made change is that, from.

* print (sudh. age(2022)) TO

* print (anuj_var. age(2022))

So, that is why we get the age of 'Anuj'.

So, it is trying to do what.
It is trying to pass a 'current year' values. over here.

* def age(Sudh, current_year):
    return current_year - sudh.year_of_birth

Then it is trying to [redacted]
Substract (-) Age or year of birth that I have passed.

* return current_year - sudh.year_of_birth

This is Symbol of (-) substraction & here I am doing substraction.

Here, what I am saying is,
How I am able to access 'Age'.
I am able to Access 'Age' with the help of these 'class variable.'

Note:-

For understanding only - example:-

let's suppose I have created

S = "sudh"

Can you please convert this into a 'Upper' case.

we will call S.upper() string variable

Def(.) Upper()

S = "sudh"

S.upper()

This method we have already done,

This is what we do.

we trying to do, is, we have declear a class variable of a "string" then from that class.

String class, with the help of that variable we are trying to call a function. This is we have learn in our 'String' class.

Similar way for others also.

'List'
'integer'
'Tuples', dictionary, 'set' etc.

A for all these must be some function which is available.

Here also we have done same thing.

Code:

Code

```
class person:

    def __init__(sudh, name, surname,
                 sudh.name1 = name
                 sudh.surname = surname
                 "
                 "
                 sudh.year_of_birth = year_of_birth

    def age(sudh, current_year):
        return current_year - sudh.year_of_birth

anuj_var = person("anuj", "bhandari")
sudh = person("Sudhanshu", "Kumar", "sudh@gmail")
gargi = person("gargi", "xyz", "  ", 23424)
print(anuj_var.age(2022))
```

test 2:-

Error outcome

```
RUN: !C! users | win10 | =
anuj_var = person("anuj", "bhandari")
TYPEERROR: __init__() missing 2 required
    positional arguments: 'email2' & 'year of
                                    birth'
```

Let's understand:

Let's suppose, we are going to pass maybe 4 or it is looking for 4 inputs.

I am going to pass may be 5 or 2 or 3 what will happened that case, what happend. in term of function.

It will give an Error; it will complain, you told that you will going to 2 variable, but you have passed 3 or 1.

Again it is an 'function', I have written. 'def'.

def __init__(

def keyword been used for function declaration.

Although, I am using a inbuilt function over here.

I am trying to over write it.

But still its a function.

what change, we made here is

in this line code we made change

* anuj_var = person ("anuj", "bhandari",
"anuj@gmail.com", 1994)

TO

* anuj_var = person ("anuj", "bhandari")

So, I have passed less data, name & Title only

as outcome, If we execute this code

It will complain or Error.

test 2:-

Cell window to

Fill "C:/users -
anuj_var = person ("anuj" "bhandari")
TypeError: __init__() missing 2 required
positional arguments: 'emailID' &
"year_of_birth"

Missing & required mail.

we have missed that you told me you will provide email ID & date of birth you have not provided.

So, as a outcome it complaining that 2 things missing, which is

Email ID & Date of Birth

* So, whenever we are trying to pass a data so, yes as per the function declaration, we are suppose to pass a data.

Code

```
class person:
    def __init__(sudh, name, ,. . . . . ):
        sudh.name1 = name
        sudh.surname = surname
        sudh.emailID = emailId
        sudh.year_of_birth = year_of_birth   X

    def age(sudh, current_year):
        return current_year - sudh.year_of_birth

anuj_var = person("anuj", "bhandari", "anuj@gmail.com",
                  1994)
sudh = person("sudhanshu", "kumar", "sudhansh@gmail.com",
              23424)
gargi = person( ,  ,  ,  , . . .  234242)
print(anuj_var.age(2022))
```

test 2:

c:// window 10 :

,, . . . outcome same

Let's understand:-

Let's Suppose,

Here, I am going to remove a 1 variable from these out of total 4 variables.

* def __init__ (self, name, ...):
  1. self.name1 = name
  2. self.surname = surname
  3. self.emailid = emailid
  4. self.year_of_birth = year_of_birth ✗

If I remove out of these 4 variable last one remove and only use above 3 variables Is it going to work?

* def __init__ (self, name, ...):
  1. self.name1 = name
  2. self.surname = surname
  3. self.emailID = emailid

Rest below code don't change anything.

As a outcome, it will not complain anything, its completely fine, we will get our outcome. Simple, instead of '4 variable', I am only using '3 variables' that is fine. My outcome will same.

Code :-

Oops project
- main.py
- Test1.py
- Test2.py
- Test3.py

main.py | Test1.py | Test2.py | Test3.py

```
class person :
* def __init__ (sudh, name, surname, email_id, year_of_birth):
    sudh.name1 = name
    sudh.surname = surname
    sudh.emailId = emailID
    sudh.year_of_birth = year_of_birth
*  def __init__ (sudh, name, surname):
    sudh.name1 = name
    sudh.surname = name

def age(sudh, current_year):
    return " " " " " " " "

anuj_var = Person("anuj", "bhandari", anuj@gmail.com, 1994)
sudh = person(" " " " " " 23424)
gargi = person(" " " " 234242)
print(anuj_var.age(2022))
```

variables

variables

Test 3
C:\ window 10\

File:
anuj_var = person("anuj", "bhandari@gmail.com",
TypeError: __init__() take 3 positional arguments but 5 were given.

let's understand :-

let's suppose,
I am going to use or create 2 'init' inside these 'init', I ~~am~~ am trying to pass only 2 data or 2 variables only, in 2nd 'init'.

* what change here in this code,
I have add 1 more (__init__) where, I have kept only 2 variable,
1st 'init' 4 variables
2nd 'init' 2 variables

If, I am going to write 2 (__init__) then, my code [redacted] out of these 2 'init' which one it will consider.
Both (__init__) or only one. ?

If we Execute Code,
As a outcome we are getting an Error, or TYPEERROR
* __init__ take 3 argument, but 5 were given

* This __init__ taking only

* Here, the '__init__' we have created. This __init__ taking how many parameter? Actually '5' including pointer, which will be by default.

* def __init__(sudh, name, surname, email Id, year_of_birth):
  1 2 3 4 5

* def __init__(sudh, name, surname):
  1 2 3
  sudh.name1 = name
  sudh.surname = surname } variables

Here, it taking 3. Its takes only 3 position argument.

1 pointer 2 variables

why you have given me 5, why you are assigning 5 over here.

* So, which (__init__) it is going to Consider,

Can we say that it will Consider

2nd one.

2nd option (__init__)

It is just trying to over right the 1st one 1st (__init__).

Next ~~Code~~ line code correct one Next page Continue...

Code

oops project

main.py | Test.1 | Test.2 | Test.3

```
class person:
    def __init__(sudh, name, "
                 "  year_of_birth):
        1 sudh.name1 = name
        2   "
        3   "
        4   "
    def __init__(sudh, name, surname):
        sudh.name1 = name
        sudh.surname = surname

    def age(sudh, current_year):
        return current_year

anuj_var = person("anuj", "bhandari")
sudh = person("sudhanshu", "kumar")
gargi = person("gargi", "xyz")
print(anuj_var.age(2022))
```

test 3.py
C:\Users\win10\

2022 ← outcome.

let's understand ↓

Here, In these code, we have made allot of changes.
what change, we have made over here.
we have removed.

*1st return current_year - Sudh.year_of_birth
(remove it because there is no year of birth so.)

Now,
return.current_year.

*2nd
anuj_var = person("anuj", "bhandari")
Sudh = person("sudhanshu", "kumar")
gargi = person("gargi", "xyz")
print(anuj_var.age(2022))

(Here we have removed; gmail & years)

Note:
we have remove 2 places 2 things.

1st place only these one
Sudh.year_of_birth.

2nd place we have removed. gmail Id & years..

Now only Name & Titles.

Now,
As a outcome we can see that, we are getting.

It is taking 2nd option of 2nd(__init__).

Note:-
If we are going to write multiple -(__init__). means 'multiple constructor' to pass a data.
It is always going to take last one or latest one.
Both are same.

Again, it is not mendatory to pass a data.
May be I have just created a 'class' end up writing a couple of methods couple of function over there.
without passing a data to my classes. that is completely fine.

* But, If we would like to pass data at time of creation of an object, only in that case (init) is required.

Project
Opps

main.py
Test 1.py
Test 2.py
Test 3.py
Test 4.py

Code

main.py | test 1 | test 2 | test 3 | Test 4.py

```
class person:
    def age(self, current_year, year_of_birth):
        return current_year - year_of_birth
    def email_id_input(self, email_id):
        print("take and mail id from form a person and print it", email_id)
    def ask_name(self):
        name = input("tell me your name")
        return name
    def ask_dob(self):
        dob = input("tell me your date of birth")
        return dob

sudh = person()
anuj = person()
gargi = person()
krish = person()
hitesh = (person()
```

Note:-

This is Half Code don't run it next page actual explain what we are looking information.

Let's understand — test4.py

Here, I have created "class person".
Inside that, I am not going to create any kind of (__init__).

* def age(self, ...):
    return current_year - year_of_birth

So, I have created a class inside that, I have created one function.

* I can create another function, I would like to take input from the emailID.

def email_id_input(, . .):
    print(, ,
    }

* Now, another, I can write a function

def ask_name(self):
    ''
    ''

* Now,
here again another function,
user input, Input is a function.

* def ask_dob(self): —— dob (date of birth)
"
"
"

So, Here in code, I have just created a 'person class:'.
I have to take some short of a input from a 'user' or may be from calculation, I have to do for all the "users." whoever 'user' who is going to register with me.
I just have to ask to all the users.

* I have not created any (__init__)

over here, I am not looking for any kind of input from user at the time of creation of variable.

Now, how I will be able to create a variable for a particular person.

* Sudh =
* anuj =
* gargi =
* Krish =
* Hitesh =

let's suppose,
I have a person, 5 person over here.

* Sudh
* anuj
* gargi
* Krish
* Hitesh

I have to take a input, of.

Question :-

(1) Can you please tell me Hitesh email Id?
(2) Can you please " " Krish date of birth?
(3) Can you ~~tell me~~ please " " Sudh age or something?

So, these are few questions, if some one going to ask.

Likewise, I have so many users. If I have to take these inputs & everything depend upon my con'context, which I am looking for.

So, ~~first~~ 1st of all I have to create a variables. a class variable.

I have to call age & sometime email ID, date of birth, .. these are thing which I am suppose to call.

Here, we are not using any 'init'.

So, whatever function, we are going to create, we have to take 1st parameter **by default**, we can use '**self**' & we can use own name or anything.

* 'self' is not a reserved keyword inside python.
Just for reference, we can use any keyword.

```
* sudh = person()
* anuj = person()
* gargi = person()
* krish = person()
* Hitesh = person()
```

Here, I have to have created variable of what 'person class'.

sudh = person() variable creation done.

Is our class looking for some input at the time of variable creation. This time.?

No, right.

But, in our previous ~~class~~ example code we have created a 'constructor'.

Responsibility of 'constructor' is a one again method by default.

* Responsibility of 'Constructor' is to take a input from the user.

But, here, in this code, I have not created anything like that.

Sudh = person()

"

"

"

"

Hitesh = person()

So, 'Sudh' become 'person variable.' like for all ami ..... Hitesh.

Now, variable of these person class I have created.

By time I am going to create these variable, can I say that, now all the function which I have written inside these class will be accessable. by all the variables.

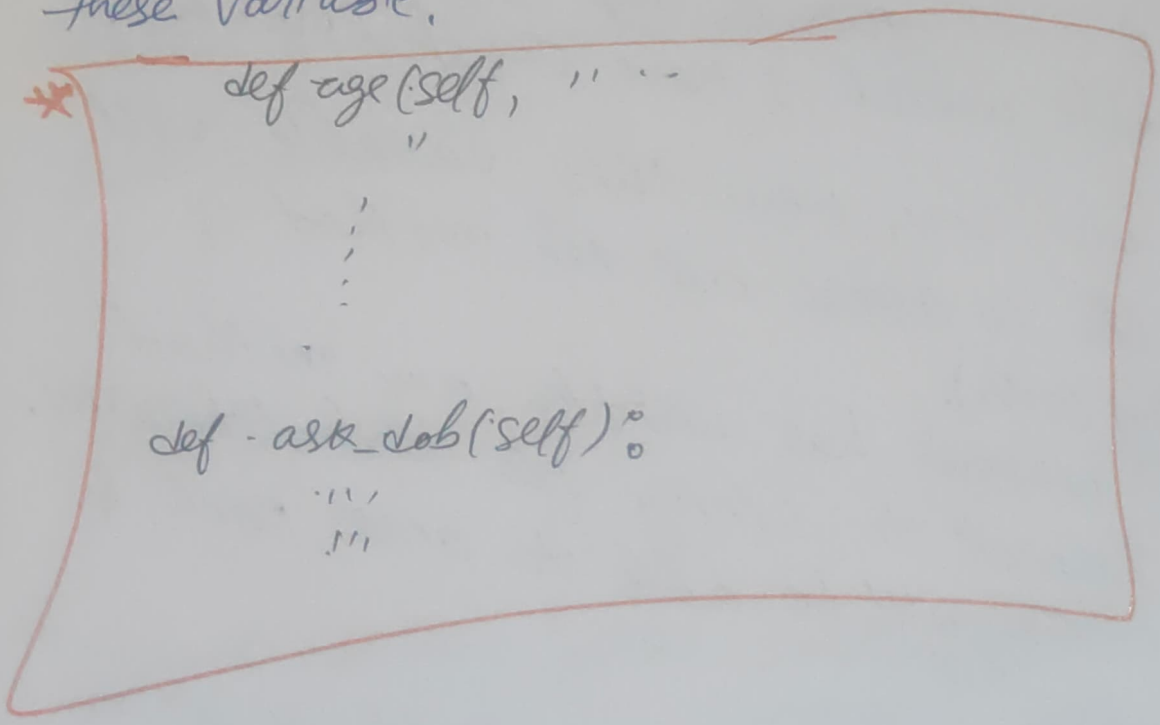
* s = "sudh"
s.upper()

let's suppose,
this is my string (str).
variable which I have created over here.
Can we say, now this variable will be able to access all the method of string (str).
whoever has written any method with respect to string for that libraries.
They will be able to access each & all
every thing.

* So, the "sudha" is a variable of person technically its a object.
'sudh'
"anni"
"
"Hitesh"
Everyone become an object.
variable of person classes.
If its a variable of person classes.
So, the way we are access all the string.

string method, by using a string variable similar way.

we will be able to access all the method of these class by using these variable.



Code

OOPS
pixels
oops
main.py
Test1.py
Test2.py
Test3.py
Test4.py

main.py | Test1 | Test2 | Test3 | Test4

~~class pes~~
class person:
    def age(self, current_years, years_of_birth):
        return .....

    def email_id_input(self, email_id):
        ''
        ''

    def ask_name(self):
        ''
        ''

    def ask_dob(self):
        ''
        return dob

sudh = person()
anuj = person()
gargi = person()
krish = person()
hitesh = person()

sudh.email_id_input("sudhanshu@ineuron.ai") (1)
print(sudh.ask_name()) (2)

Test 4
C:\Users\ashto

Take and email id from a person & ~~print it~~

→ Sudhanshu@ineuron.ai

tell me your name Sudhanshu kumar

Let's understand :-

Now
Some one asked me, Question.
Can you please tell me "email ID" of
a person → 'sudh'

So, simple, I will do.

I will add one line code in ~~the~~ for these.

(A) * sudh.email_id = input("sudhanshu@ineuron.ai")

If we execute these code as a outcome we can see.

C:\Users\person 10

take if ~~a~~ mail id form a person ...

* sudhanshu@ineuron.ai

out come

Email Id we are able to get.

2nd part (B)

1:41:20

Question :-
Some one asking Can you please
asked a name a full name of 'sudh' ?

Yes,

[redacted]

* sudh.email_id_input("sudhanshu@ineuron.ai")
* sudh.ask_name()

If we exelute by putting this one line piece of code. [redacted].

[redacted] As a outcome we can see.

- test.py/
C:\\user\windows10

take and mail id form a person & print it
sudhanshu@ineuron.ai

tell me your name Sudhanshu Kumar

Code

main.py / test1 / Test2 / Test3 / Test4

oops
projects
oops
main.py
Test1.py
Test2.py
Test3.py
Test4.py

```
Class person:
    def age (self, Current_year, ....)
        return ...
    def email_id_input (self, .... )
        ''
        ''
    def ask_name (self):
        ''
        ''
    def ask_dob (self):
        ''
        ''
        return dob

Sudh = person()
anuj = person()
    ''
    ''
    ''
Hitesh = person()

Sudh.email_id_input("Sudhsh@gmail.com")
print(sudh.ask_name())
print(hitesh.ask_dob())
```

C:\Users. with 10
take & mail id form a . . . Sudhanshu @gmail.com
tell me your name 'Sudhanshu kumar'
Sudhanshu kumar tell me your date of birth 3453453
3453453

Here we can give 2 / more

Let's understand

Q: Now Someone is looking for Date-of-Birth.

ASK date_of_Birth of "Hitesh".

* print(hitesh.ask_dob())

outCome -> 345 3453 (I can give like this also) 27 to 1995.

In that case what I will do, I will call first 'Hitesh' variable.

because all of them are able to access all the 4 variables of the 4 methods.

ask Date of Birth; I of 4 putting this number as Date of Birth 'print it.'

print(hitesh.ask_dob())

* Sudh.email_id_input("sudhanshu@ineuron.ai")
* print(sudh.ask_name())

* print(hitesh.ask_dob())

If execute it

Asking for name, so put 'Sudhanshu Kumar' & then asking for date of birth. 'Hitesh' date of birth asking from Hitesh variable.

(3453453) - we have provided over here & can put like this also - (27101995). this is fine.

As a outcome we can see over here is (3453453) & am able to get.

So, Here . sir is trying to show a reuseability of a Code.

I have just written ~~4 for~~ 4 function. inside particular classes.

```
* def age(self,
      return

* def email_id_input(self.
      print("                )

* def ask_name(self):
      .name. =
      return

* def ask_dob(self):
      dob =
      return
```

let's suppose, I have 1 millions people, Can I say that for all 1 million people I will be able to use all of these 4 function. I don't have to rewrite again & again. Now its become generic Code.

Now, we have create these 'class' over here. inside these 4 function.

Now, I will just keep adding a function as much as possible.

[redacted]

Can I say that for person, I [redacted] will end up writing all kind of functions. Every kind of function, I will end up writing.

Now next user whoever has to use basically any of these function.

They don't have to write it, its a ready made available they just have to call. and they will be able to use it.

* So, I am able to increase the scability of a code. [redacted]
* I am able to increase the ~~use~~ reusability of a code at the same point of a time, by using 'classes'.
* difference ~~bete~~ between 'class' & 'object'.

So, far so forth here we are writing a function but in a structure way.

we have written one function one file somewhere. Another file somewhere, this is what we are doing.

Can we say that, now we have started collecting a similar kind of a function, function which belong to a person.

* Now, if I will create another 'class' where I will write all the function related to 'car'.

'class person' or 'class car',

this will my choice whether I will use 'person class' or 'car class'.

Just to create an structure we have adopted a [redacted] OOPS concept.

[redacted]

Theory wise logically we are not writing anything.

Here what we are trying is, we trying to structured over here, so that I will be able to 'package or Bundle' a things.

So that I will be able to maintain a code, because number of line of code increase, it hard to maintain.

* let's suppose,
we have written thousand line of code & question is that can you please find out [Test 4] function that we have created in our classes.
It will [redacted] 'puzzle'.

* what if going forward, we will keep all the 'list', dictionary, Tuples etc every function in seperate seperate class.

* 'Tuples' related in a different 'class'.
* 'Dictionary' releated in a different 'class'.
* 'List' " " " "

In that case, Can we say.
Question is Can you please call these function.
In that case, we will click 'List' class & find code.
Similar way others.
That the advantage, we will get if we write code these structured way.

Advantage

Module
Structure
Robustness
Easy to find
Easy to develop.
Easy to emplement

Because, As number of line of code will increase it hard to maintain so.

Q:- why oops concept is important?

If we are developing a project.

- For each person 1 task
- If one is work on 'Tuples' another is working on 'Dictionary', 'set', 'List' etc. This way seperate seperate & later, we Combined all the code in one place & we merged all the code this is how project. Complete.

This is how we develop any kind of project & any kind of product. This is [redacted] where Concept of oops Come to a picture.

TASK - For This Class

1.57

Q: TASK 1 :- In a past whatever questions we have discussed with respect to 'List', 'tuples', 'dictionary', 'set', 'string' ~~and~~ try to create a ~~sepr~~ separate 'class' for each and everything and restructure your code for all the segment and submit.

~~Q: Task 2 :-~~

1: Instruction Number 1 :-
Always use exception Handling.

2: Instruction Number 2 :-
~~Always~~ use logging as well.

3. Reformat your code and submit it to me before tomorrow's class.

Email Id :- sudhanshu@ineuron.ai
sunny.savita@ineuron.ai

I am only looking for your seperate python file in your Github link.

All people done there task & Submitted. to see those task follow below link. of (Github).

1. Anil padiyar :-

https://[redacted] github.com/anilpadiyar/Assignment/blob/main/oop1_Task.py[redacted]

2. Rohit :-

https://[redacted] github.com/BorkiSamRohit/task6_02-07-2022/blob/main/stringTasks.py

3. Uday :-

https://github.com/udayavel/FSDS/blob/main/ds_class.py

4. Piyush :-

https://github.com/piyush-123/Tasks/blob/main/task_02Jul2022.py