

python 

Presented by Swetha

What is python?

Python is a high-level, interpreted programming language known for its readability and ease of use.

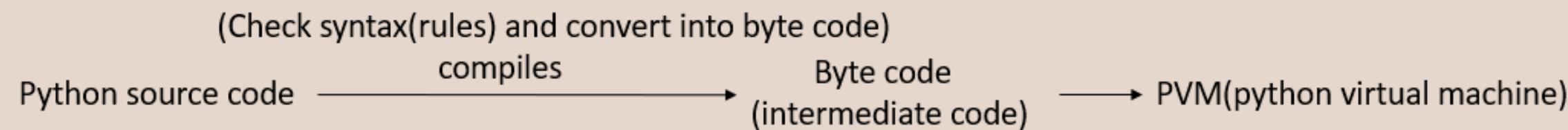
Why python is called easy language?

- 1.Simple and Readable Syntax
- 2.No Need to Declare Data Types
- 3.Large Standard Library
- 4.Interpreted Language

Who invented python?

Python was invented by Guido van Rossum in the late 1980s and was first released in 1991.

General flow of how Python code is executed:-



Conditional Statements:-

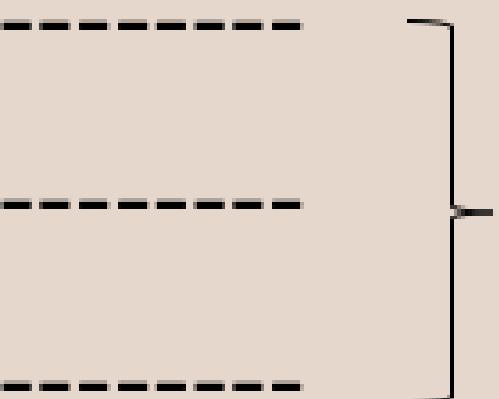
- Conditional statements in Python are used to execute code based on certain conditions.
 - 1.Simple If
 - 2.If-else
 - 3.If-elif-else
 - 4.Match case

1. Simple if:-

Executes a block of code if a condition is true.

Syntax:-

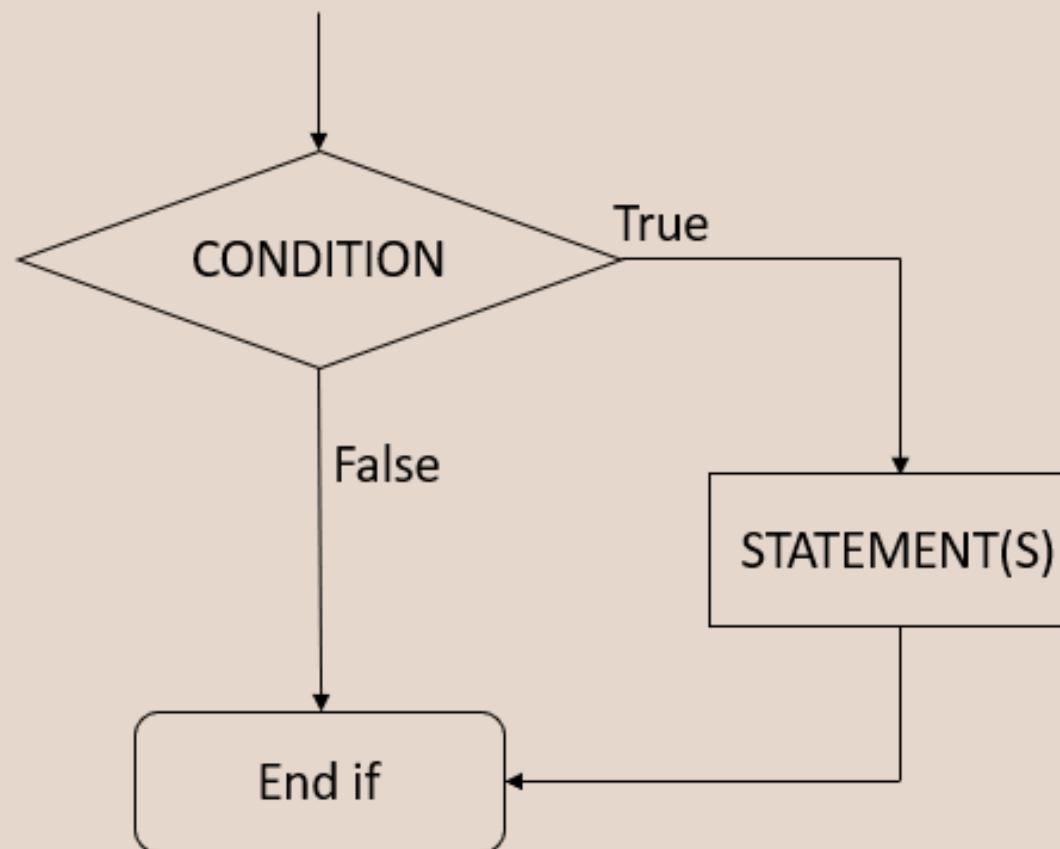
If(test condition):



Indentation block
of if statement

Other statements

Flow chart for simple if:-



Example:-

```
Age=20
if Age>=18:
    print("Welcome to PVR")
print("Thanks,Visit again") #this statement will appear even the age is below 18
```

Welcome to PVR
Thanks,Visit again

2. If-else statement:-

Executes one block of code if a condition is true, and another if it's false.

Syntax:-

```
if(test condition):
```

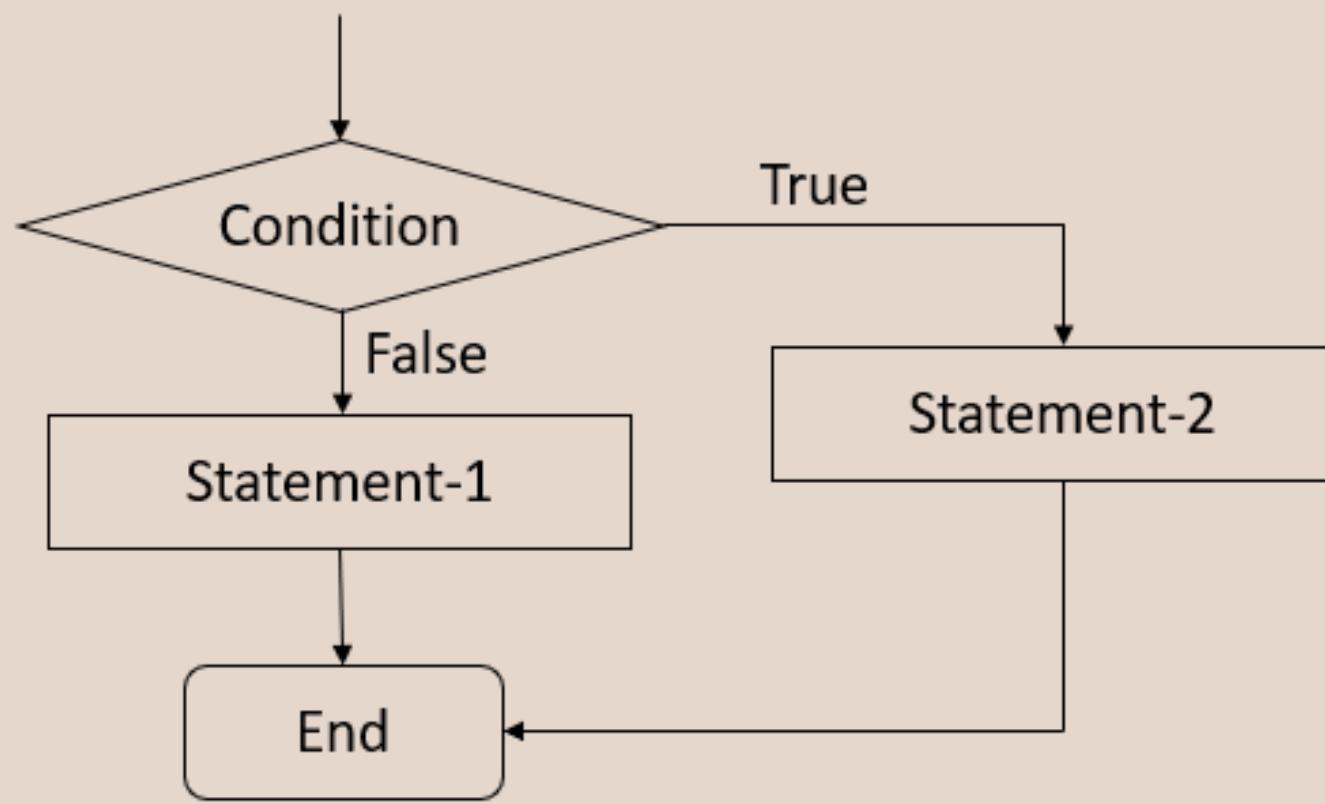
```
    ---  
    --- } Indentation block of  
    ---     if statement
```

```
else:
```

```
    ---  
    --- } indentation block  
    ---     of else statement
```

```
other statements
```

Flow chart of if-else statement:-



Example:-

```
X=3
if X>5:
    print("X is greater than 5")
else:
    print("X is lesser than 5")
print("Thank you")
```

X is lesser than 5

Thank you

3. If-elif-else statement:-

Checks multiple conditions in sequence.

Syntax:-

if(test condition):

----} Indentation block
of if statement

elif:

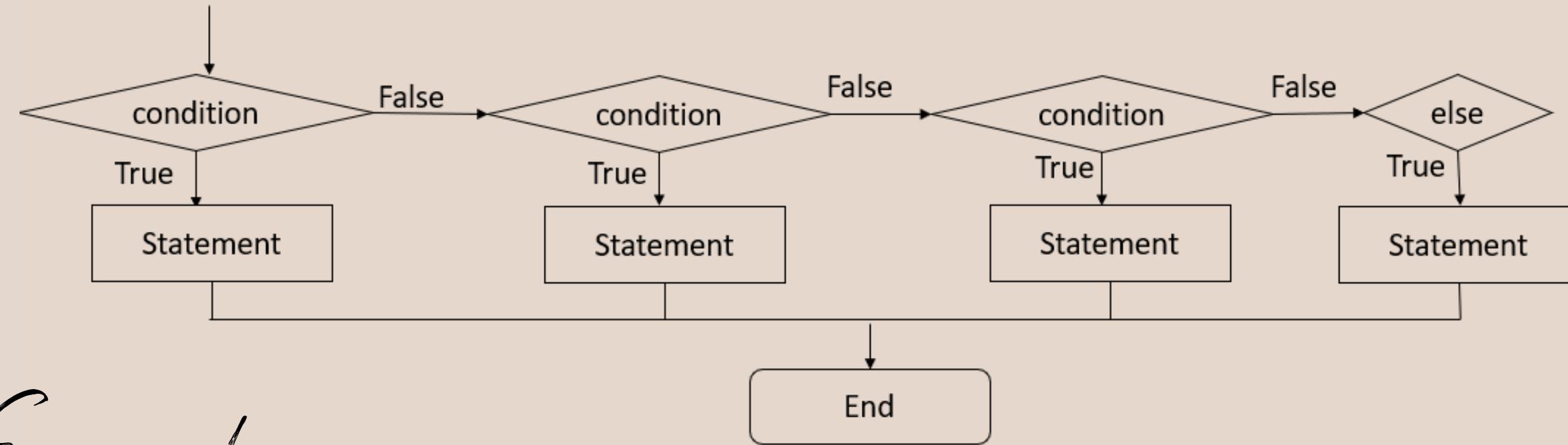
----} Indentation block of
elif statement

else:

----} Indentation block
of else statement

other statement

Flow chart of if-elif-else statement:-



Example:

```
num=int(input("Enter a number: "))
if num>100:
    print("The number is greater than 100.")
elif num>50:
    print("The number is greater than 50 but less than or equal to 100.")
elif num>10:
    print("The number is greater than 10 but less than or equal to 50.")
else:
    print("The number is 10 or less.")
```

```
Enter a number: 15
The number is greater than 10 but less than or equal to 50.
```

4. Match case statement:-

It is a way to handle multiple conditions.

Syntax:-

match variable:

 case pattern1:

 ---- } Indentation block
 ---- for case 1

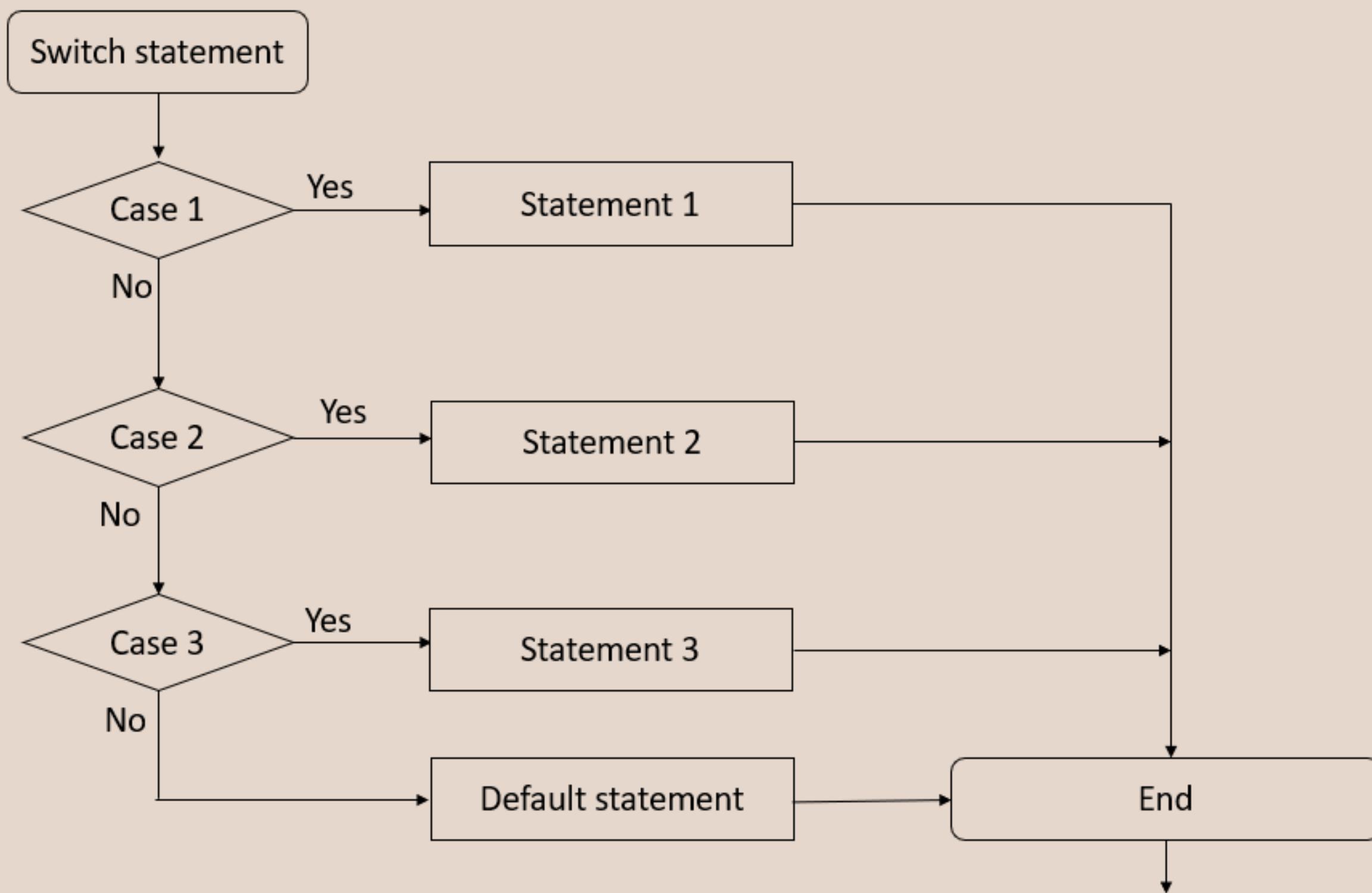
 case pattern2:

 ---- } Indentation block
 ---- for case 2

 case pattern _:

 ---- } Indentation block
 ---- for case 3

Flow chart of match case statement:-



Example:-

```
print("choose any one option from the menu:")
print("1.addition")
print("2.subtraction")
print("3.multiplication")
number=int(input("enter your choice:"))
a=int(input("enter first number:"))
b=int(input("enter second number:"))
match number:
    case 1:
        c=a+b
        print(f"the sum of {a} and {b} is {c}")
    case 2:
        c=a-b
        print(f"the sub of {a} and {b} is {c}")
    case 3:
        c=a*b
        print(f"the mul of {a} and {b} is {c}")
    case _:
        print("Invalid input!")
```

```
choose any one option from the menu:
1.addition
2.subtraction
3.multiplication
enter your choice: 3
enter first number: 4
enter second number: 4
the mul of 4 and 4 is 16
```

Nested if statements:-

An if statement inside another if statement.

Syntax:-

if(condition1):

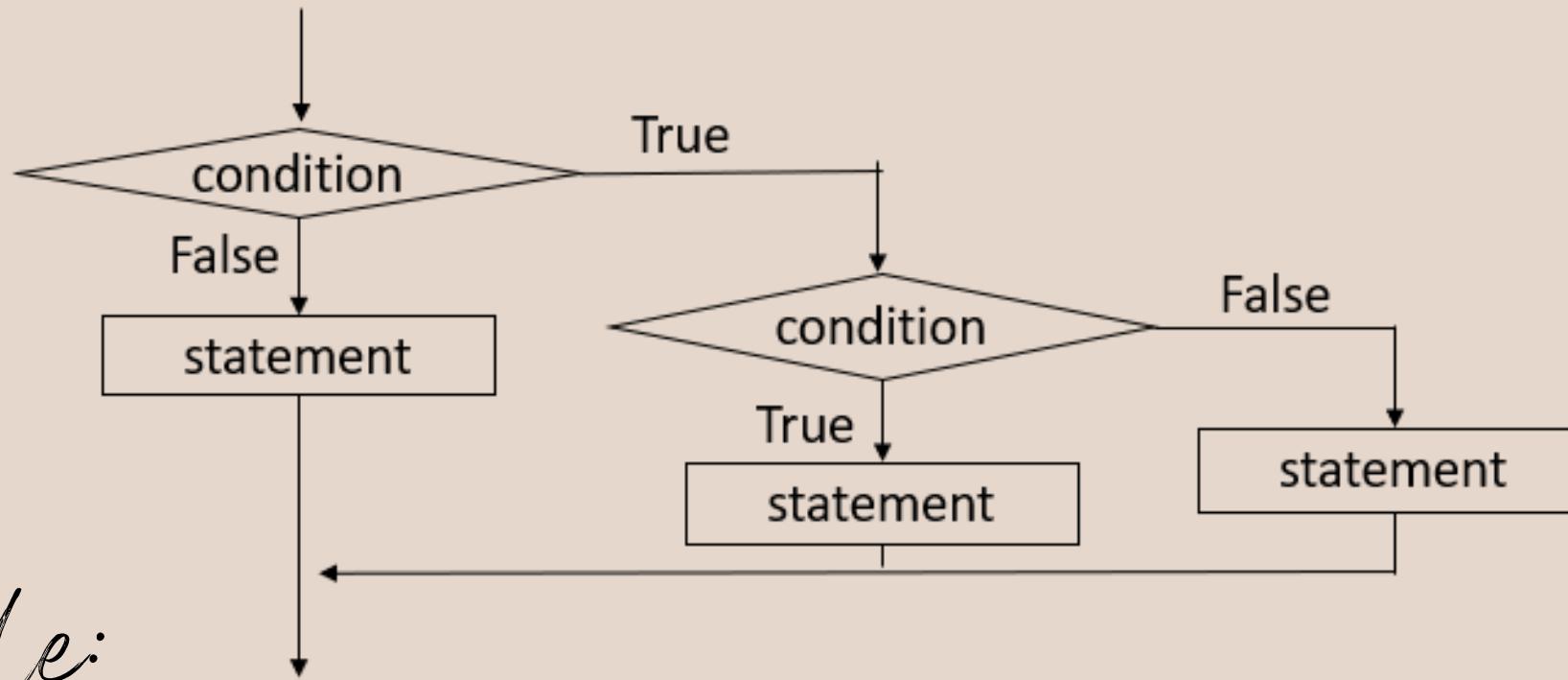
if(condition2):

else:

else:



Flow chart for nested if statement:-



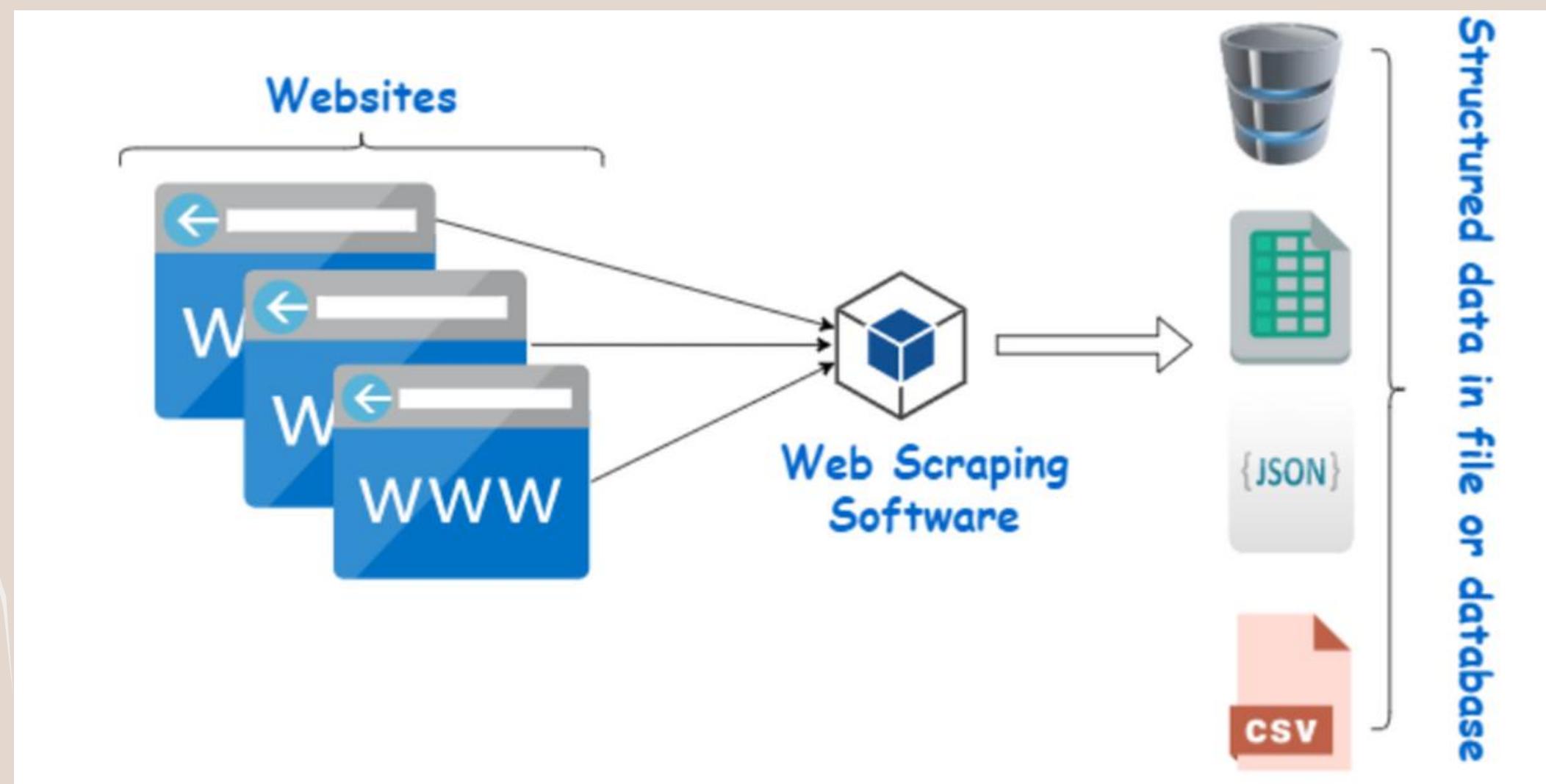
Example:

```
num = int(input("Enter a number: "))
if num > 0:
    print("The number is positive")
    if num % 2 == 0:
        print("It is also even")
    else:
        print("But it is odd")
else:
    print("The number is negative")
```

```
Enter a number: 3
The number is positive
But it is odd
```

Web scraping:-

It is the process
of extracting data
from the website.



Importing libraries, package and module:-

```
: #web scraping essentials
import requests
from bs4 import BeautifulSoup #this plays is major role in web scraping
import re
import time
import pandas as pd #used for data analysis =CONVERTING RAW DATA INTO DATA SET
```

Website:-

Url=<https://www.bikewale.com/royalenfield-bikes/>

```
url="https://www.bikewale.com/royalenfield-bikes/"
response=requests.get(url)
soup=BeautifulSoup(response.text,'html.parser')
```

To extract all bikes name and prices:-

```
# Find all bike name tags
bike_name_tags=soup.find_all('h3', class_='o-j4 o-jq o-hM o-c4')
price_tags = soup.find_all('span', class_='o-jJ o-jr o-j3 o-kJ')
print("Bike Name and Price:")
for name_tag, price_tag in zip(bike_name_tags, price_tags):
    bike_name=name_tag.get_text(strip=True)
    price=price_tag.get_text(strip=True)
    print(f"bike_name:{bike_name}\t\tPrice:{price}")
```

Output:-

Bike Name and Price:	
bike_name:Royal Enfield Hunter 350	Price:₹ 1,50,000
bike_name:Royal Enfield Classic 350	Price:₹ 1,95,133
bike_name:Royal Enfield Continental GT 650	Price:₹ 3,22,420
bike_name:Royal Enfield Bullet 350	Price:₹ 1,74,779
bike_name:Royal Enfield Guerrilla 450	Price:₹ 2,39,000
bike_name:Royal Enfield Meteor 350	Price:₹ 2,07,777
bike_name:Royal Enfield Himalayan 450	Price:₹ 2,85,000
bike_name:Royal Enfield Hunter 350 [2024]	Price:₹ 1,49,900
bike_name:Royal Enfield Shotgun 650	Price:₹ 3,63,541
bike_name:Royal Enfield Super Meteor 650	Price:₹ 3,68,077
bike_name:Royal Enfield Classic 650	Price:₹ 3,36,612
bike_name:Royal Enfield Interceptor 650	Price:₹ 3,06,243
bike_name:Royal Enfield Goan Classic 350	Price:₹ 2,35,000
bike_name:Royal Enfield Scram 440	Price:₹ 2,07,598
bike_name:Royal Enfield Bear 650	Price:₹ 3,42,875

Royal Enfield Himalayan 450 >

h3.o-j4.o-jq.o-hM.o-c4 223.83 × 19.33

Color	#484848
Font	16px LatoGoogle, sans-serif
Padding	0px 8px 0px 0px

ACCESSIBILITY

Name	Royal Enfield Himalayan 450
Role	heading
Keyboard-focusable	<input checked="" type="checkbox"/>

span.o-jJ.o-jr.o-j3.o-kJ 70.19 × 18.67

Color	#333333
Font	15px LatoGoogle, sans-serif

ACCESSIBILITY

Contrast	Aa 12.63 <input checked="" type="checkbox"/>
Name	
Role	generic
Keyboard-focusable	<input checked="" type="checkbox"/>

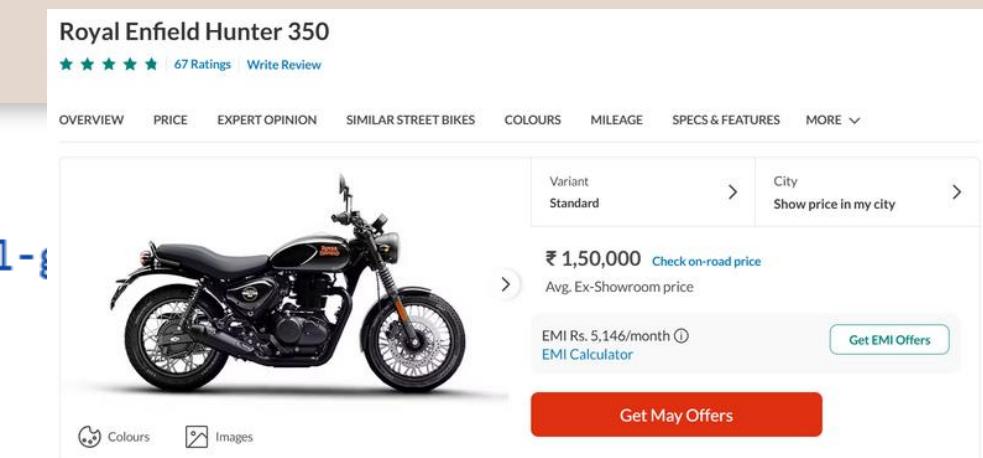
₹ 2,85,000

To extract all the hyper links:-

```
links=soup.find_all('a')
for link in links:
    href=link.get('href')
    text=link.text.strip() #strip()=it removes all the spaces
    if href:
        print(f"{text or '[no text]'}----->{href}")
```

Output:

```
Royal Enfield Hunter 350----->https://www.bikewale.com//royalenfield-bikes/hunter-350/
Royal Enfield Classic 350----->https://www.bikewale.com//royalenfield-bikes/classic-350/
Royal Enfield Continental GT 650----->https://www.bikewale.com//royalenfield-bikes/continental-gt-650/
Royal Enfield Bullet 350----->https://www.bikewale.com//royalenfield-bikes/bullet/
Royal Enfield Guerrilla 450----->https://www.bikewale.com//royalenfield-bikes/guerrilla-450/
Royal Enfield Meteor 350----->https://www.bikewale.com//royalenfield-bikes/meteor-350/
Royal Enfield Himalayan 450----->https://www.bikewale.com//royalenfield-bikes/himalayan/
Royal Enfield Hunter 350 [2024]----->https://www.bikewale.com//royalenfield-bikes/hunter-350-2024/
Royal Enfield Shotgun 650----->https://www.bikewale.com//royalenfield-bikes/shotgun-650/
Royal Enfield Super Meteor 650----->https://www.bikewale.com//royalenfield-bikes/super-meteor-650/
Royal Enfield Classic 650----->https://www.bikewale.com//royalenfield-bikes/classic-650/
Royal Enfield Interceptor 650----->https://www.bikewale.com//royalenfield-bikes/interceptor/
Royal Enfield Goan Classic 350----->https://www.bikewale.com//royalenfield-bikes/goan-classic-350/
Royal Enfield Scram 440----->https://www.bikewale.com//royalenfield-bikes/scram-440/
Royal Enfield Bear 650----->https://www.bikewale.com//royalenfield-bikes/bear-650/
Royal Enfield Himalayan 750----->https://www.bikewale.com//royalenfield-bikes/himalayan-750/
Royal Enfield Scrambler 450----->https://www.bikewale.com//royalenfield-bikes/scrambler-450/
Royal Enfield Flying Flea C6----->https://www.bikewale.com//royalenfield-bikes/flying-flea/
```



Extracting images from class="data-src":-

```
#extract images from class=data-src
images=soup.find_all('img')
for img in images:
    src=img.get('data-src')
    print(src)
```

```
| Bear 650" title="Royal Enfield Bear 650" data-src="https://imgd.aeplcdn.com/227x128/n/cw/ec/138795/bear-650-right-front
```

Output:-

```
None
None
None
None
https://imgd.aeplcdn.com/227x128/n/cw/ec/127499/bullet-right-side-view-13.jpeg?isig=0&q=80
https://imgd.aeplcdn.com/227x128/n/cw/ec/138829/guerrilla-450-right-side-view.jpeg?isig=0&q=80
https://imgd.aeplcdn.com/227x128/n/cw/ec/51245/meteor-350-right-side-view-10.png?isig=0&q=80
https://imgd.aeplcdn.com/227x128/n/cw/ec/110431/himalayan-right-side-view-6.png?isig=0&q=80
https://imgd.aeplcdn.com/227x128/n/cw/ec/124013/hunter-350-right-side-view-5.png?isig=0&q=80
https://imgd.aeplcdn.com/227x128/n/cw/ec/165367/shotgun-650-right-side-view-4.png?isig=0&q=80
https://imgd.aeplcdn.com/227x128/n/cw/ec/108885/super-meteor-650-right-side-view-6.png?isig=0&q=80
https://imgd.aeplcdn.com/227x128/n/cw/ec/138841/classic-650-right-side-view.jpeg?isig=0&q=80
https://imgd.aeplcdn.com/227x128/n/cw/ec/49657/interceptor-right-side-view-13.png?isig=0&q=80
https://imgd.aeplcdn.com/227x128/n/cw/ec/151461/goan-classic-350-right-side-view-2.jpeg?isig=0&q=80
https://imgd.aeplcdn.com/227x128/n/cw/ec/192235/scram-440-right-side-view-3.png?isig=0&q=80
https://imgd.aeplcdn.com/227x128/n/cw/ec/138795/bear-650-right-front-three-quarter.jpeg?isig=0&q=80
https://imgd.aeplcdn.com/227x128/n/cw/ec/106133/himalayan-650-right-rear-three-quarter.jpeg?isig=0&q=80
https://imgd.aeplcdn.com/227x128/n/cw/ec/138831/scrambler-450-right-side-view.jpeg?isig=0&q=80
https://imgd.aeplcdn.com/227x128/n/cw/ec/181825/flying-flea-c6-right-side-view-2.jpeg?isig=0&q=80
https://imgd.aeplcdn.com/227x128/n/cw/ec/138839/bullet-650-right-front-three-quarter.jpeg?isig=0&q=80
https://imgd.aeplcdn.com/227x128/n/cw/ec/138835/continental-gt-150-right-front-three-quarter.jpeg?isig=0&q=80
```

Extracting images from class="src" :-

```
images = soup.find_all('img')
for img in images:
    src = img.get('src')
    print(src)
```

```
<div style="top:-11px;left:-11px" class="o-an o-D"></div>
▼<div class="o-hl o-C o-o PhYMAu">
  
</div>
</div>
▼<div class="o-fb o-aR o-C">
```

Output:-

Extracting all images:-

```
images=soup.find_all('img')
for img in images:
    src=img.get('data-src') or img.get('src')
    print(src)
```

Output:-



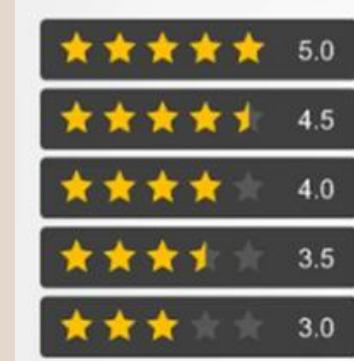
<https://www.facebook.com/tr?id=904915516356416&ev=PageView&noscript=1>
<https://imgd.aeplcdn.com/227x128/n/cw/ec/201293/hunter-350-2025-right-side-view-2.jpeg?isig=0&q=80>
<https://imgd.aeplcdn.com/227x128/n/cw/ec/183389/classic-350-right-side-view-62.jpeg?isig=0&q=80>
<https://imgd.aeplcdn.com/227x128/n/cw/ec/49656/continental-gt-right-side-view-12.png?isig=0&q=80>
<https://imgd.aeplcdn.com/227x128/n/cw/ec/127499/bullet-right-side-view-13.jpeg?isig=0&q=80>
<https://imgd.aeplcdn.com/227x128/n/cw/ec/138829/guerrilla-450-right-side-view.jpeg?isig=0&q=80>
<https://imgd.aeplcdn.com/227x128/n/cw/ec/51245/meteor-350-right-side-view-10.png?isig=0&q=80>
<https://imgd.aeplcdn.com/227x128/n/cw/ec/110431/himalayan-right-side-view-6.png?isig=0&q=80>
<https://imgd.aeplcdn.com/227x128/n/cw/ec/124013/hunter-350-right-side-view-5.png?isig=0&q=80>
<https://imgd.aeplcdn.com/227x128/n/cw/ec/165367/shotgun-650-right-side-view-4.png?isig=0&q=80>
<https://imgd.aeplcdn.com/227x128/n/cw/ec/108885/super-meteor-650-right-side-view-6.png?isig=0&q=80>
<https://imgd.aeplcdn.com/227x128/n/cw/ec/138841/classic-650-right-side-view.jpeg?isig=0&q=80>
<https://imgd.aeplcdn.com/227x128/n/cw/ec/49657/interceptor-right-side-view-13.png?isig=0&q=80>
<https://imgd.aeplcdn.com/227x128/n/cw/ec/151461/goan-classic-350-right-side-view-2.jpeg?isig=0&q=80>
<https://imgd.aeplcdn.com/227x128/n/cw/ec/192235/scram-440-right-side-view-3.png?isig=0&q=80>
<https://imgd.aeplcdn.com/227x128/n/cw/ec/138795/bear-650-right-front-three-quarter.jpeg?isig=0&q=80>
<https://imgd.aeplcdn.com/227x128/n/cw/ec/106133/himalayan-650-right-rear-three-quarter.jpeg?isig=0&q=80>
<https://imgd.aeplcdn.com/227x128/n/cw/ec/138831/scrambler-450-right-side-view.jpeg?isig=0&q=80>
<https://imgd.aeplcdn.com/227x128/n/cw/ec/181825/flying-flea-c6-right-side-view-2.jpeg?isig=0&q=80>
<https://imgd.aeplcdn.com/227x128/n/cw/ec/138839/bullet-650-right-front-three-quarter.jpeg?isig=0&q=80>
<https://imgd.aeplcdn.com/227x128/n/cw/ec/138835/continental-gt-450-right-front-three-quarter.jpeg?isig=0&q=80>
<https://imgd.aeplcdn.com/227x128/n/cw/ec/146033/himalayan-electric-right-front-three-quarter.png?isig=0&q=80>
<https://imgd.aeplcdn.com/227x128/n/cw/ec/138825/himalayan-raid-450-right-side-view.jpeg?isig=0&q=80>

Extracting rating:-

```
print("extracting rating:")
ratings=soup.find_all('p', class_='o-iY o-cx o-jq o-d o-j o-jJ')
for i, rating in zip(range(1,16), ratings):
    rating=rating.get_text(strip=True)
    print(f"bike{i}-{rating}")
```

Output:-

```
extracting rating:
bike1-4.9/5
bike2-4.7/5
bike3-4.8/5
bike4-4.7/5
bike5-4.7/5
bike6-4.6/5
bike7-4.7/5
bike8-4.7/5
bike9-4.7/5
bike10-4.8/5
bike11-4.6/5
bike12-4.6/5
bike13-4.8/5
bike14-4.2/5
bike15-4.6/5
```



A screenshot of a web page showing a rating of 4.9/5 with 63 Ratings. The rating is displayed in a green box with a star icon. A tooltip shows the CSS selector p.o-iY.o-cx.o-jq.o-d.o-j.o-jJ. Below the rating, there is a detailed view of the element's properties:

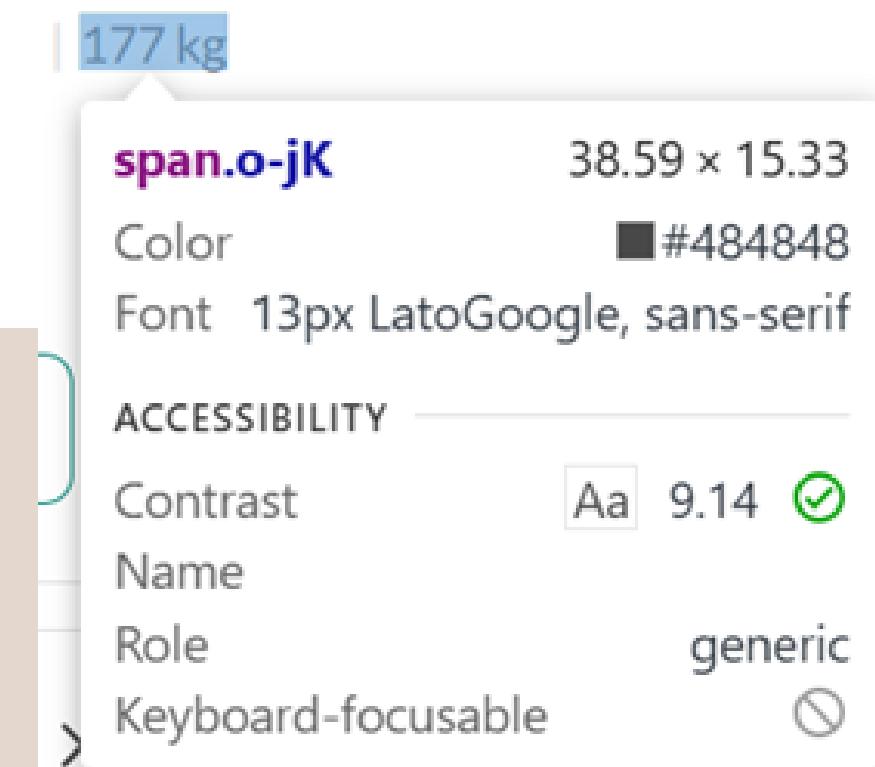
Color	#00857A
Font	13px LatoGoogle, sans-serif
Padding	0px 8px 0px 2px
ACCESSIBILITY	
Name	
Role	paragraph
Keyboard-focusable	(unchecked)

Extracting weight of bike:-

```
kgs=soup.find_all('span',class_='o-iZ o-jf')  
for kg in kgs:  
    weight=kg.get_text(strip=True)  
    if "kg" in weight:  
        print(weight)
```

Output:-

```
177 kg  
195 kg  
211 kg  
195 kg  
185 kg  
191 kg  
196 kg  
177 kg  
240 kg  
241 kg  
243 kg  
213 kg  
197 kg  
196 kg  
216 kg
```



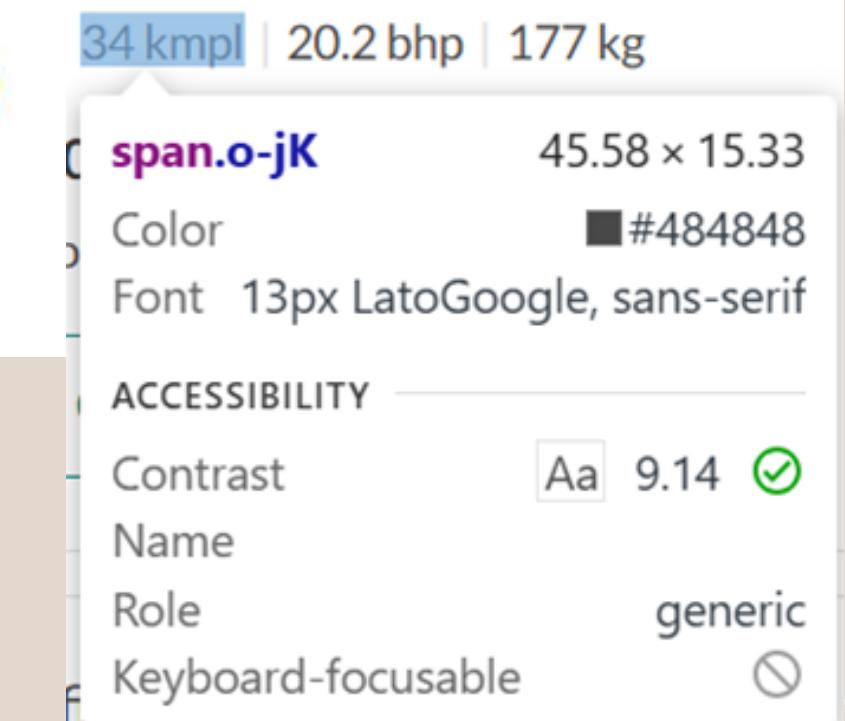
Extracting kmpl:-



```
kmpls=soup.find_all('span',class_='o-iZ o-jf')
for kmpl in kmpls:
    km_per_1=kmpl.get_text(strip=True)
    if "kmpl" in km_per_1:
        print(km_per_1)
```

Output:-

```
34 kmpl|
34 kmpl|
23.5 kmpl|
36 kmpl|
30 kmpl|
33 kmpl|
30 kmpl|
35 kmpl|
22 kmpl|
25 kmpl|
31 kmpl|
24 kmpl|
33 kmpl|
30 kmpl|
25 kmpl|
```



Appending the each output:-

```
all_bike=[]
bike_name_tags=soup.find_all('h3', class_='o-j4 o-jq o-hM o-c4')
price_tags = soup.find_all('span', class_='o-jJ o-jr o-j3 o-kJ')
ratings=soup.find_all('p', class_='o-iY o-cx o-jq o-d o-j o-JJ')
images=soup.find_all('img')[1:] #skip first image
kgs=soup.find_all('span',class_='o-iZ o-jf')
weights=[]

for kg in kgs:
    weight=kg.get_text(strip=True)
    if "kg" in weight:
        weights.append(weight)

for name_tag,price_tag,rating,img,weight in zip(bike_name_tags, price_tags,ratings,images,weights):
    bike_name=name_tag.get_text(strip=True)
    price=price_tag.get_text(strip=True)
    ratings=rating.get_text(strip=True)
    src=img.get('data-src') or img.get('src')
    all_bike.append({
        'title':bike_name,
        'price':price,
        "rating":ratings,
        "src":src,
        "weight":weight
    })
print(len(all_bike))
```

Checking output:-

```
for bike in all_bike:  
    print(f"BIKE NAME= {bike['title']}\\n{bike['price']}\\n{bike['rating']}\\n{bike['src']}\\n{bike['weight']}")  
  
BIKE NAME= Royal Enfield Hunter 350  
₹ 1,50,000  
4.9/5  
https://imgd.aeplcdn.com/227x128/n/cw/ec/201293/hunter-350-2025-right-side-view-2.jpeg?isig=0&q=80  
177 kg  
BIKE NAME= Royal Enfield Classic 350  
₹ 1,95,133  
4.7/5  
https://imgd.aeplcdn.com/227x128/n/cw/ec/183389/classic-350-right-side-view-62.jpeg?isig=0&q=80  
195 kg  
BIKE NAME= Royal Enfield Continental GT 650  
₹ 3,22,420  
4.8/5  
https://imgd.aeplcdn.com/227x128/n/cw/ec/49656/continental-gt-right-side-view-12.png?isig=0&q=80  
211 kg  
BIKE NAME= Royal Enfield Bullet 350  
₹ 1,74,779  
4.7/5  
https://imgd.aeplcdn.com/227x128/n/cw/ec/127499/bullet-right-side-view-13.jpeg?isig=0&q=80  
195 kg  
BIKE NAME= Royal Enfield Guerrilla 450  
₹ 2,39,000  
4.7/5  
https://imgd.aeplcdn.com/227x128/n/cw/ec/138829/guerrilla-450-right-side-view.jpeg?isig=0&q=80  
185 kg
```

Saving file to csv:-

Output in excel:-

```
import pandas as pd
df=pd.DataFrame(all_bike)
df.head()
df.to_csv("bikes_data.csv",index=False)
print("data set saved successfully")
```

data set saved successfully

	A	B	C	D	E
1	title	price	rating	src	weight
2	Royal Enfield Hunter 350	₹ 1,50,000	4.9/5	https://imgd.aeplcdn.com/227x128/n/cw/ec/201293/hunter-350-2025-right-side-view-2.jpeg?isig=0&q=80	177 kg
3	Royal Enfield Classic 350	₹ 1,95,133	4.7/5	https://imgd.aeplcdn.com/227x128/n/cw/ec/183389/classic-350-right-side-view-62.jpeg?isig=0&q=80	195 kg
4	Royal Enfield Continental GT 650	₹ 3,22,420	4.8/5	https://imgd.aeplcdn.com/227x128/n/cw/ec/49656/continental-gt-right-side-view-12.png?isig=0&q=80	211 kg
5	Royal Enfield Bullet 350	₹ 1,74,779	4.7/5	https://imgd.aeplcdn.com/227x128/n/cw/ec/127499/bullet-right-side-view-13.jpeg?isig=0&q=80	195 kg
6	Royal Enfield Guerrilla 450	₹ 2,39,000	4.7/5	https://imgd.aeplcdn.com/227x128/n/cw/ec/138829/guerrilla-450-right-side-view.jpeg?isig=0&q=80	185 kg
7	Royal Enfield Meteor 350	₹ 2,07,777	4.6/5	https://imgd.aeplcdn.com/227x128/n/cw/ec/51245/meteor-350-right-side-view-10.png?isig=0&q=80	191 kg
8	Royal Enfield Himalayan 450	₹ 2,85,000	4.7/5	https://imgd.aeplcdn.com/227x128/n/cw/ec/110431/himalayan-right-side-view-6.png?isig=0&q=80	196 kg
9	Royal Enfield Hunter 350 [2024]	₹ 1,49,900	4.7/5	https://imgd.aeplcdn.com/227x128/n/cw/ec/124013/hunter-350-right-side-view-5.png?isig=0&q=80	177 kg
10	Royal Enfield Shotgun 650	₹ 3,63,541	4.7/5	https://imgd.aeplcdn.com/227x128/n/cw/ec/165367/shotgun-650-right-side-view-4.png?isig=0&q=80	240 kg
11	Royal Enfield Super Meteor 650	₹ 3,68,077	4.8/5	https://imgd.aeplcdn.com/227x128/n/cw/ec/108885/super-meteor-650-right-side-view-6.png?isig=0&q=80	241 kg
12	Royal Enfield Classic 650	₹ 3,36,612	4.6/5	https://imgd.aeplcdn.com/227x128/n/cw/ec/138841/classic-650-right-side-view.jpeg?isig=0&q=80	243 kg
13	Royal Enfield Interceptor 650	₹ 3,06,243	4.6/5	https://imgd.aeplcdn.com/227x128/n/cw/ec/49657/interceptor-right-side-view-13.png?isig=0&q=80	213 kg
14	Royal Enfield Goan Classic 350	₹ 2,35,000	4.8/5	https://imgd.aeplcdn.com/227x128/n/cw/ec/151461/goan-classic-350-right-side-view-2.jpeg?isig=0&q=80	197 kg
15	Royal Enfield Scram 440	₹ 2,07,598	4.2/5	https://imgd.aeplcdn.com/227x128/n/cw/ec/192235/scram-440-right-side-view-3.png?isig=0&q=80	196 kg
16	Royal Enfield Bear 650	₹ 3,42,875	4.6/5	https://imgd.aeplcdn.com/227x128/n/cw/ec/138795/bear-650-right-front-three-quarter.jpeg?isig=0&q=80	216 kg
17					

Thank You So Much

Prathyush jha | I.T. Vedant

