

Banking →

functionality

E-commerce

login method () {

}

Add user 15-20
username - 6-25 Login Excel sheet

Test Date] spec.

loop; loop

⊕ 7-8 TCs

method

7-8 time
execute by using

V	V
V	I
I	I
I	V
V	B
B	V
B	B

→ Excel sheet ? - Test Data - Manual Test Data

→ DB

→ properties.
k-v

• java

Constants

10 Rows 5 col

JAVA

columns

→ script

login()

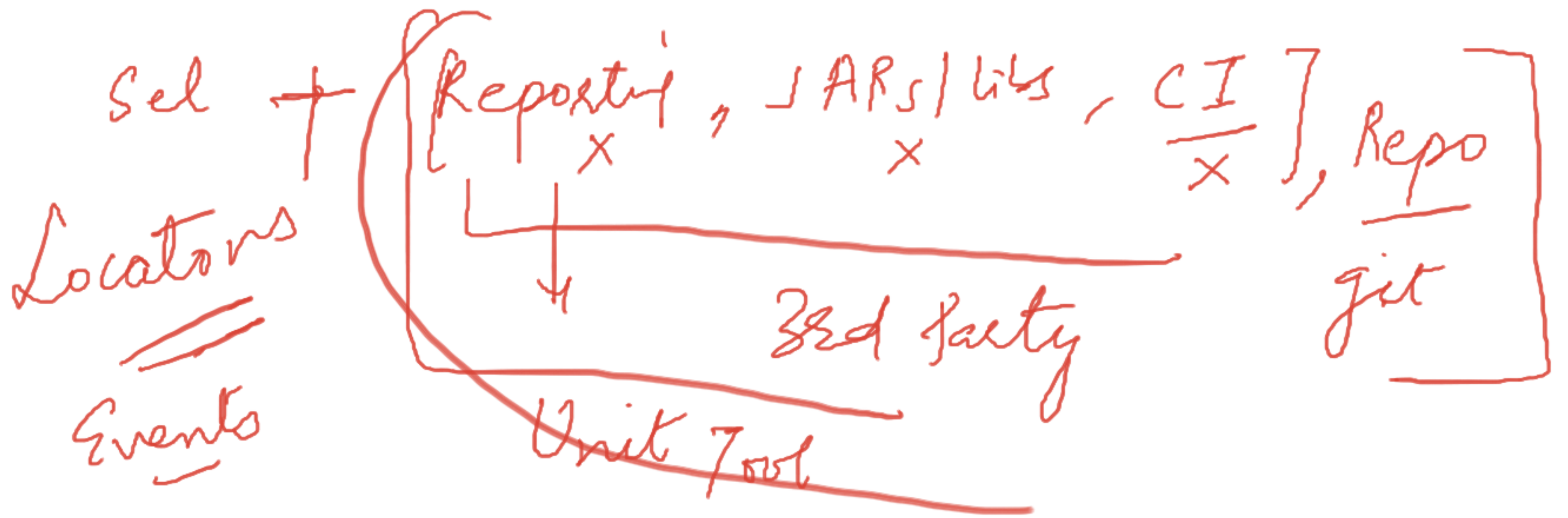
Row 2

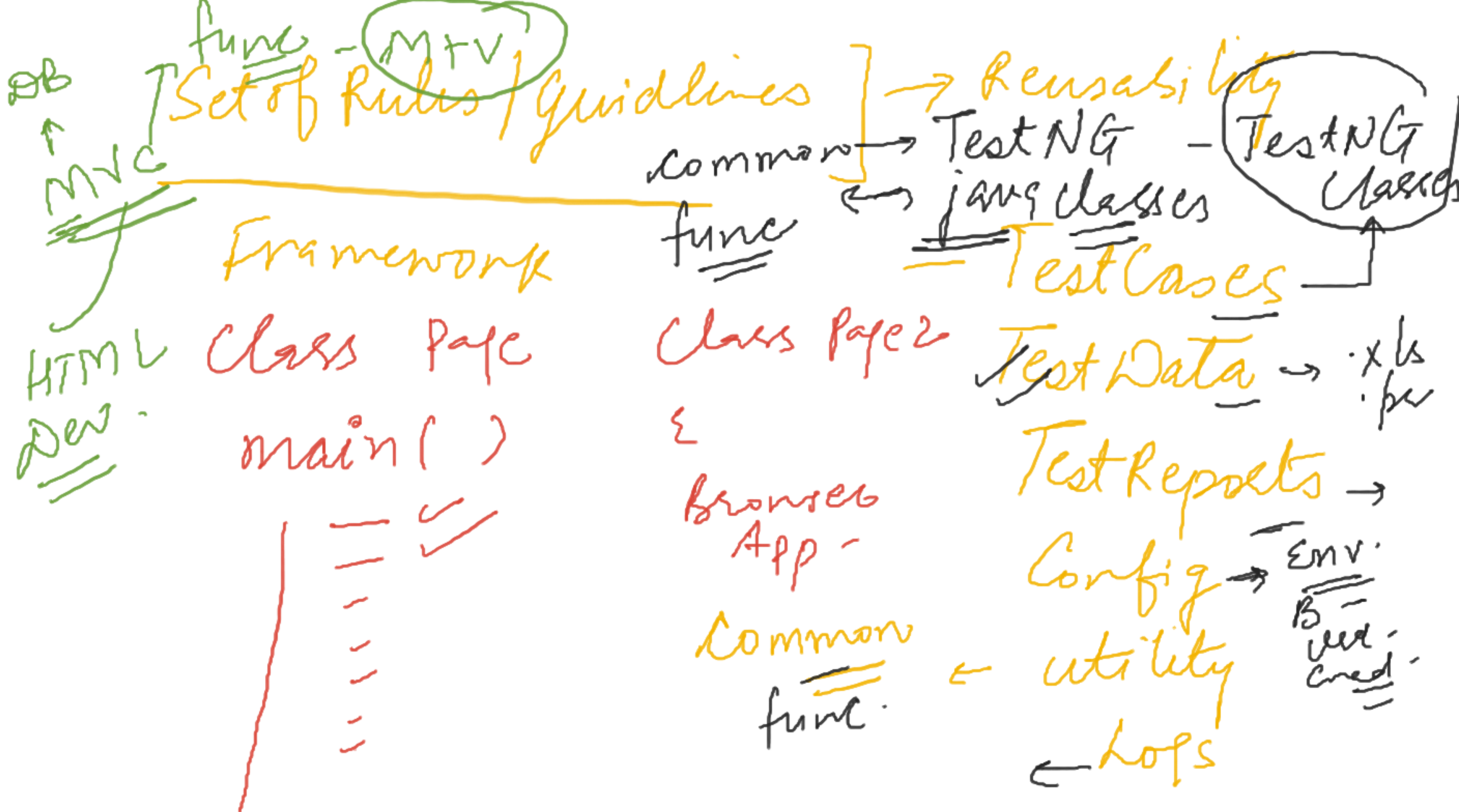
Row 3

3

Get + JAVA

↓ ↳ 5 days





Framework :- Data Driven framework, X

✓✓
Unit Testing
framework
(it-else
main)

(2) keyword Driven framework

(3) DDF + KDF = Hybrid

(4) Modular Framework / POM

Page Object Model Design
pattern

DDF + POM
KDF

→ Hybrid

(5) { BDD } ✓✓ → JUnit

@Test → class
public void m1() {

System.out.println("Hello");

TC 3

@Override
↓

When to start Automation? →

- Stable build
- Quality inc / cost dec
- Complex ^{func} Time save platform across various
- Test Coverage increase
- func → TestData] Parameterization
execute the same TC using various Test Data.
100 Rows

When not to Automatic?

- TCs which designed new → not executed manually at least once.
- requirement changes frequently x
- adhoc - basis →
- don't have enough time
- expected result is not there. Catches.
- usability → , $\{O+P, \text{Authorization code}\}$

Selenium

QTP

Functionige

Safal

Coding

Open Source

No coding

A + AI

Automation

Web App

Mobile

Multi platform
Browsers

Desktop App + Mobile Test

paid

Basic script

APT Testing → Manually
Auto matw

Mobile Automation: -

✓
← Ruby | JAVA
↓
Cucumber ✓
↓
BDD Arch.

[Appium] → A+IOS
↓
Selendroid
↓
Android

Selenium component -
Sel IDF

[Chrome -] update / Eclipse / JAVA
FF -] 1.8/15

<!DOCTYPE HTML> Browser HTML
docum

<html> → startig

↓ Tag

<head> <title> MyPage </title>

</head>

</html> closing

FF|
Chrome]

? DOM (Document object
Model)
Document (root)

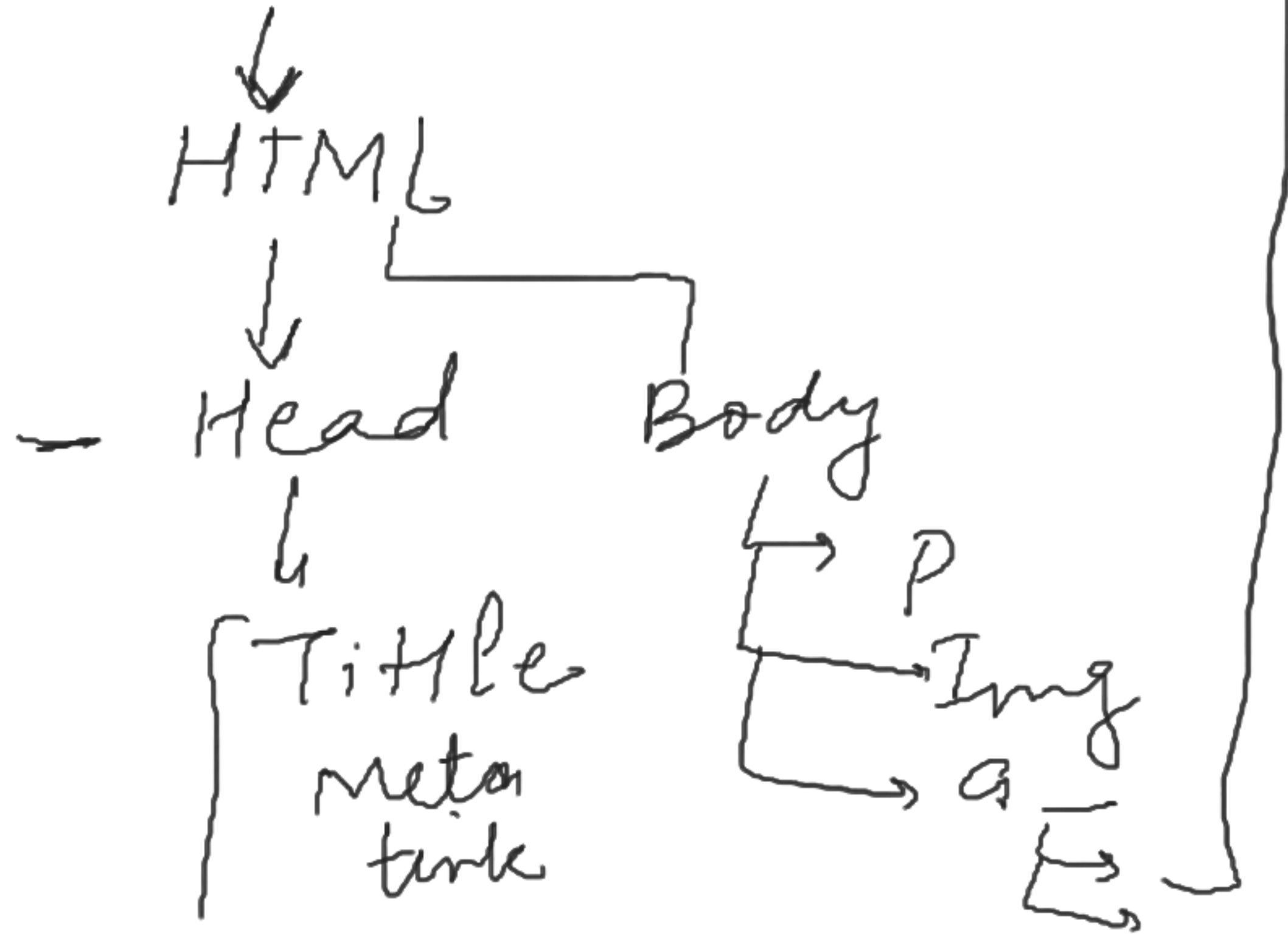


Image:

~~<img~~

~~src = "path of Image">~~

Web
Element

Tag Selector

HTML

Element

attribute attr-
= value

tag name

attr value

**<input
name="txtUsername"
id="txtUsername"
type="text">**

1. attr. name

2.

3

a name

a v

Form Elements :- ^{input}TextFields - type = text
PasswordField | type = password

- ✓ Radio Buttons - input | type = 'radio'
- ✓ ☐ Check Boxes - input | type = 'checkbox'

Drop Down → → select → option

Message: → textarea



Buttons :- Submit ^{Tag} `<input type='submit'`
`<button>` Data Server] DB
Store

Image → ~~Form~~ `<input type='image'`
as a
Submit
button `<input type = 'button'`

links → link

link → color
Bg color
font
underline
tag name

attribute
attr-
val

a {
text-decoration:
= none;
}

CSS
a: hover {
text-decoration:
underline;
}

attr {
 name = attr.value
 id = z
 class = z
 tagname
}

Static value
also = 0-4 X

<table>

<tr>

<td>

1 </td>

<td>

2 </td>

</tr>

</table>

1	2
---	---

