Manual_Assignment_3
Testing On Live Application
Submitted by Kuldeep Parmar

1. What is RDBMS

Relational Database Management System (RDBMS) is a more advanced version of a DBMS system that allows access to data in a more efficient way. It is used to store or manage only the data that are in the form of tables

The software used to store, manage, query, and retrieve data stored in a relational database is called a relational database management system (RDBMS)

The RDBMS provides an interface between users and applications and the database, as well as administrative functions for managing data storage, access, and performance

2. What is SQL

SQL stands for Structured Query Language SQL is a standard language for storing, manipulating and retrieving data in databases

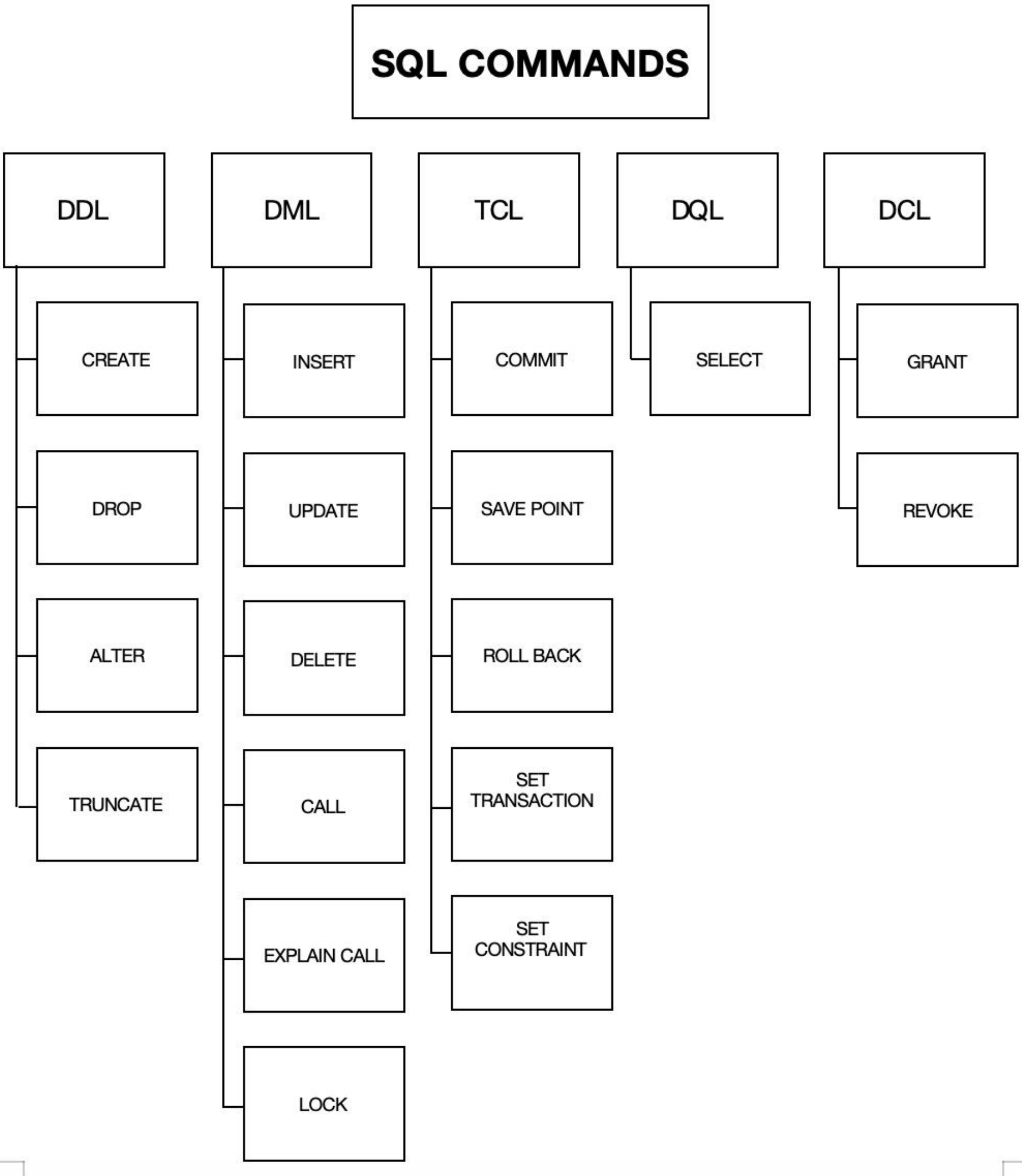
SQL allows you to access and manipulate the databases

The use of SQL is in: MySQL, SQL Server, MS Access, Oracle, Sybase, Informix, Postgres, and other database systems

3. Write SQL Commands

SQL commands are mainly categorized into five categories:

DDL - Data Definition Language
DML - Data Manipulation Language
TCL - Transaction Control Language
DQL - Data Query Language
DCL - Data Control Language



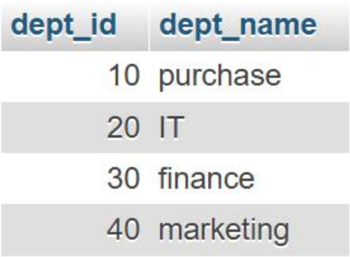
4. What is join?

A JOIN is a clause used in RDBMS to combine rows from two or more tables, based on a related column between them

The JOIN keyword merges two or more tables and creates a temporary image of the merged table, then according to the conditions provided, it extracts the required data from the image table, and once data is fetched, the temporary image of the merged tables is dumped

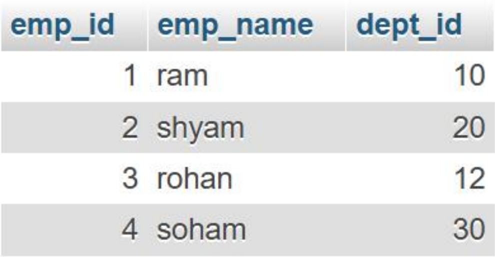
5. Write types of JOIN

departments



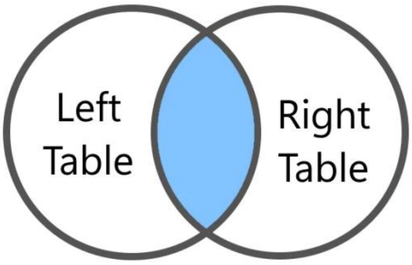
dept_id	dept_name
10	purchase
20	IT
30	finance
40	marketing

employees



emp_id	emp_name	dept_id
1	ram	10
2	shyam	20
3	rohan	12
4	soham	30

(INNER) JOIN: A JOIN clause which returns records that have matching values in both tables

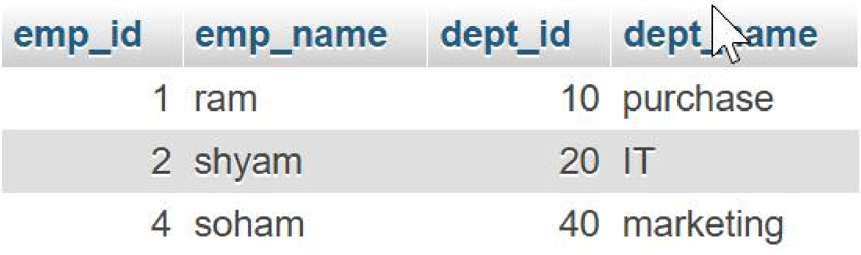


```
SELECT e.emp_id, e.emp_name, e.dept_id, d.dept_name
FROM employees e
JOIN departments d
ON e.dept_id = d.dept_id;
```

OR

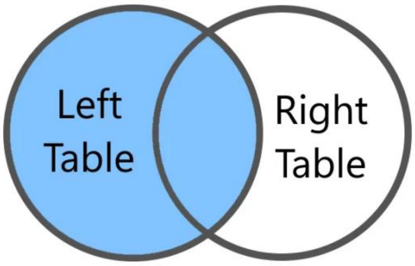
```
SELECT e.emp_id, e.emp_name, e.dept_id, d.dept_name
FROM employees e
INNER JOIN departments d
ON e.dept_id = d.dept_id;
```

Result:



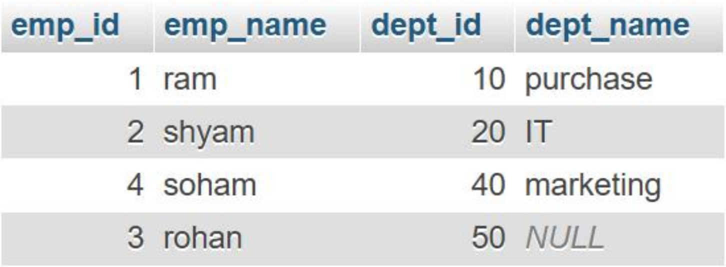
emp_id	emp_name	dept_id	dept_name
1	ram	10	purchase
2	shyam	20	IT
4	soham	40	marketing

LEFT (OUTER) JOIN: A JOIN clause which returns all records from the left table, and the matched records from the right table



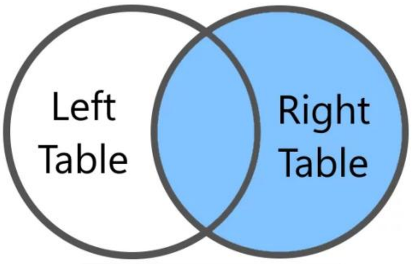
```
SELECT e.emp_id, e.emp_name, e.dept_id, d.dept_name
FROM employees e
LEFT OUTER JOIN departments d ON e.dept_id =
d.dept_id;
```

Result:



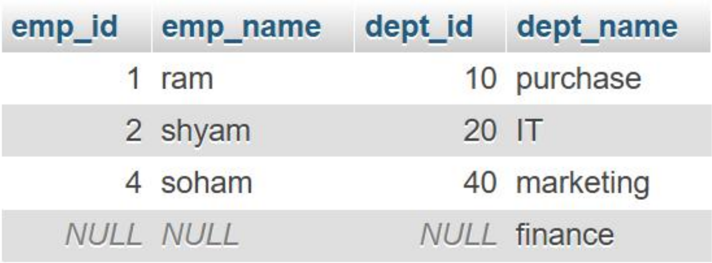
emp_id	emp_name	dept_id	dept_name
1	ram	10	purchase
2	shyam	20	IT
4	soham	40	marketing
3	rohan	50	*NULL*

RIGHT (OUTER) JOIN: A JOIN clause which returns all records from the right table, and the matched records from the left table



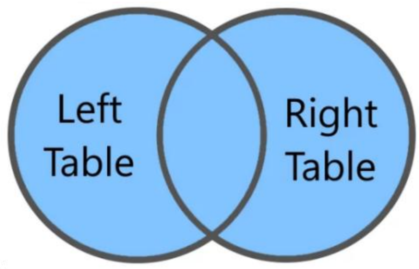
```
SELECT e.emp_id, e.emp_name, e.dept_id, d.dept_name
FROM employees e
RIGHT OUTER JOIN departments d ON e.dept_id =
d.dept_id;
```

Result:



emp_id	emp_name	dept_id	dept_name
1	ram	10	purchase
2	shyam	20	IT
4	soham	40	marketing
NULL	*NULL*	*NULL*	finance

FULL (OUTER) JOIN: A JOIN clause which returns all records when there is a match in either left or right table

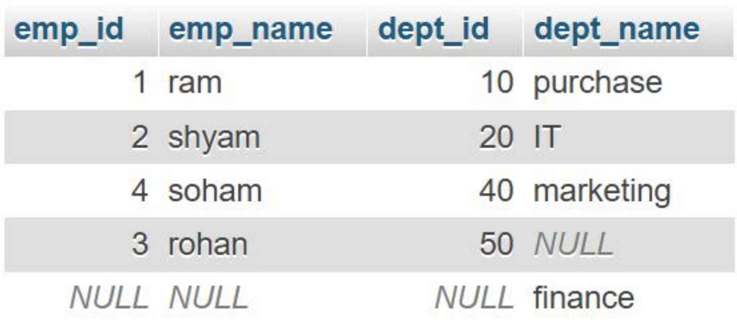


```
SELECT e.emp_id, e.emp_name, e.dept_id, d.dept_name
FROM employees e
FULL OUTER JOIN departments d ON e.dept_id =
d.dept_id;
```

OR

```
SELECT e.emp_id, e.emp_name, e.dept_id, d.dept_name
FROM employees e
LEFT OUTER JOIN departments d
ON e.dept_id = d.dept_id
UNION
SELECT e.emp_id, e.emp_name, e.dept_id, d.dept_name
FROM employees e
RIGHT OUTER JOIN departments d
ON e.dept_id = d.dept_id;
```

Result:



emp_id	emp_name	dept_id	dept_name
1	ram	10	purchase
2	shyam	20	IT
4	soham	40	marketing
3	rohan	50	*NULL*
NULL	*NULL*	*NULL*	finance

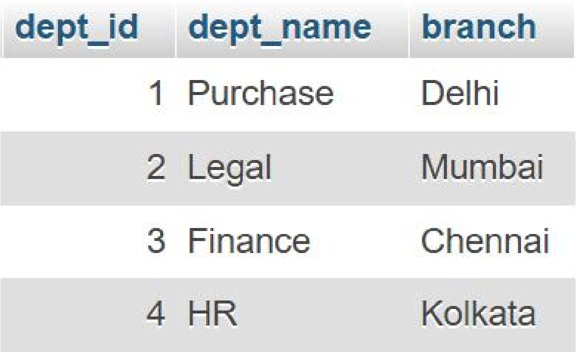
6. Howmany types of constraints in SQL. Describe them

The following types of constraints are commonly used in SQL:

NOT NULL: Ensures that a column cannot have a NULL value

```
CREATE TABLE department
(
   dept_id INT NOT NULL,
   dept_name VARCHAR (20) NOT NULL,
   branch VARCHAR (20) NOT NULL
);

INSERT INTO department VALUES (1,'Purchase','Delhi');
INSERT INTO department VALUES (2,'Legal','Mumbai');
INSERT INTO department VALUES (3,'Finance','Chennai');
INSERT INTO department VALUES (4,'HR','Kolkata');
```

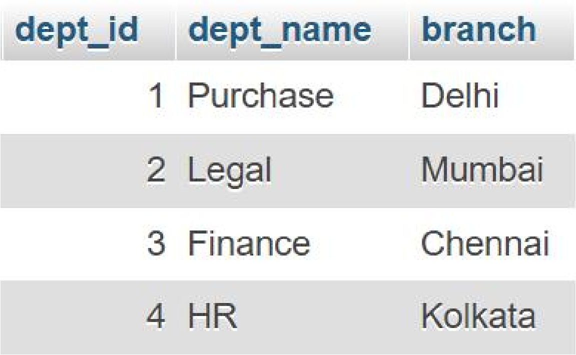


dept_id	dept_name	branch
1	Purchase	Delhi
2	Legal	Mumbai
3	Finance	Chennai
4	HR	Kolkata

UNIQUE: Ensures that all values in a column are different

```
CREATE TABLE department
(
   dept_id INT NOT NULL UNIQUE,
   dept_name VARCHAR (20) NOT NULL,
    branch VARCHAR (20) NOT NULL
);

INSERT INTO department VALUES (1,'Purchase','Delhi');
INSERT INTO department VALUES (2,'Legal','Mumbai');
INSERT INTO department VALUES (3,'Finance','Chennai');
INSERT INTO department VALUES (4,'HR','Kolkata');
```

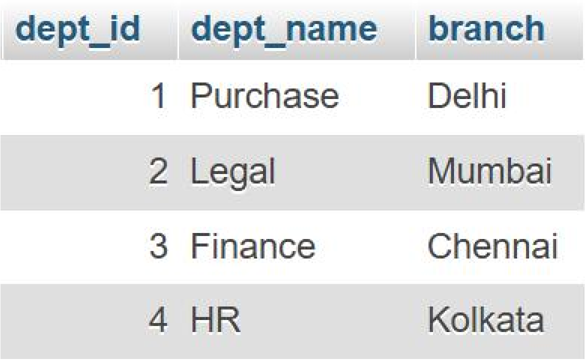


dept_id	dept_name	branch
1	Purchase	Delhi
2	Legal	Mumbai
3	Finance	Chennai
4	HR	Kolkata

PRIMARY KEY: A combination of a NOT NULL and UNIQUE. Uniquely identifies each row in a table

```
CREATE TABLE department
(
    dept_id INT PRIMARY KEY,
    dept_name VARCHAR (20),
    branch VARCHAR (20)
);

INSERT INTO department VALUES (1,'Purchase','Delhi');
INSERT INTO department VALUES (2,'Legal','Mumbai');
INSERT INTO department VALUES (3,'Finance','Chennai');
INSERT INTO department VALUES (4,'HR','Kolkata');
```



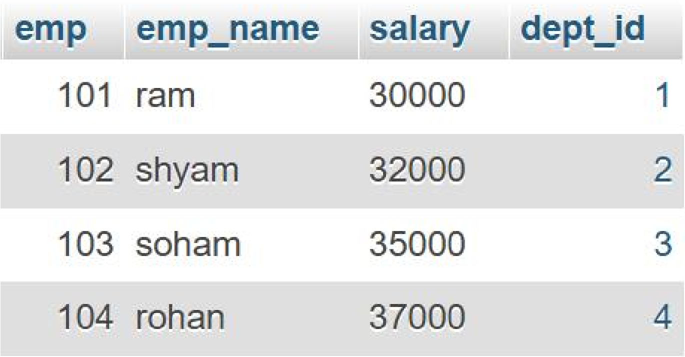
dept_id	dept_name	branch
1	Purchase	Delhi
2	Legal	Mumbai
3	Finance	Chennai
4	HR	Kolkata

FOREIGN KEY: Prevents actions that would destroy links between tables

```
CREATE TABLE employee
(
    emp INT,
    emp_name VARCHAR (20),
    salary VARCHAR (20),
    dept_id INT,

    PRIMARY KEY (emp),
    FOREIGN KEY (dept_id) references department(dept_id)
);

INSERT INTO employee VALUES (101,'ram',30000,1);
INSERT INTO employee VALUES (102,'shyam',32000,2);
INSERT INTO employee VALUES (103,'soham',35000,3);
INSERT INTO employee VALUES (104,'rohan',37000,4);
```

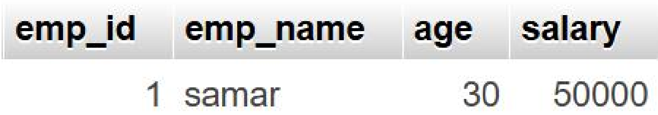


emp	emp_name	salary	dept_id
101	ram	30000	1
102	shyam	32000	2
103	soham	35000	3
104	rohan	37000	4

CHECK: Ensures that the values in a column satisfies a specific condition

```
CREATE TABLE employees
(
emp_id INT PRIMARY KEY,
emp_name VARCHAR (100),
age INT CHECK (age >= 18),
salary INT CHECK (salary > 0)
);

INSERT INTO employees (emp_id, emp_name, age,
salary) VALUES (1, 'samar', 30, 50000);
```

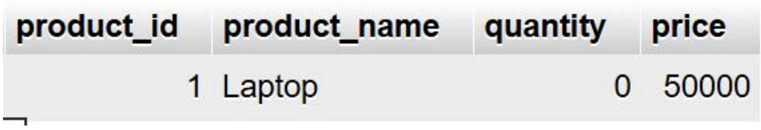


emp_id	emp_name	age	salary
1	samar	30	50000

Here, age column has a CHECK constraint to ensure that only values 18 or older can be inserted and salary column has a CHECK constraint to ensure that the salary is greater than 0

DEFAULT: Sets a default value for a column if no value is specified

```
CREATE TABLE products
(
   product_id INT PRIMARY KEY,
   product_name VARCHAR(255),
   quantity INT DEFAULT 0,
   price INT (10) DEFAULT 20
);
INSERT INTO products (product_id, product_name)
VALUES (1, 'Laptop');
```



product_id	product_name	quantity	price
1	Laptop	0	50000

Here, quantity and price are set default initially then added afterwards

7. Difference between RDBMS vs DBMS

RDBMS	DBMS
In RDBMS, data stored is in table format	In DBMS data stored is in the file format
In RDBMS, multiple data elements are accessible together	In DBMS, individual access of data elements
In RDBMS, data in the form of a table are linked together	In DBMS, there is no connection between data
RDBMS supports distributed database	In DBMS, there is no support for distributed database
In RDBMS, data is stored in a large amount	In DBMS, data stored is a small quantity
RDBMS supports multiple users	DBMS supports a single user
In RDBMS, the software and hardware requirements are higher	In DBMS, the software and hardware requirements are low
Example: Oracle, SQL Server	Example: XML, Microsoft Access

8. What is API Testing

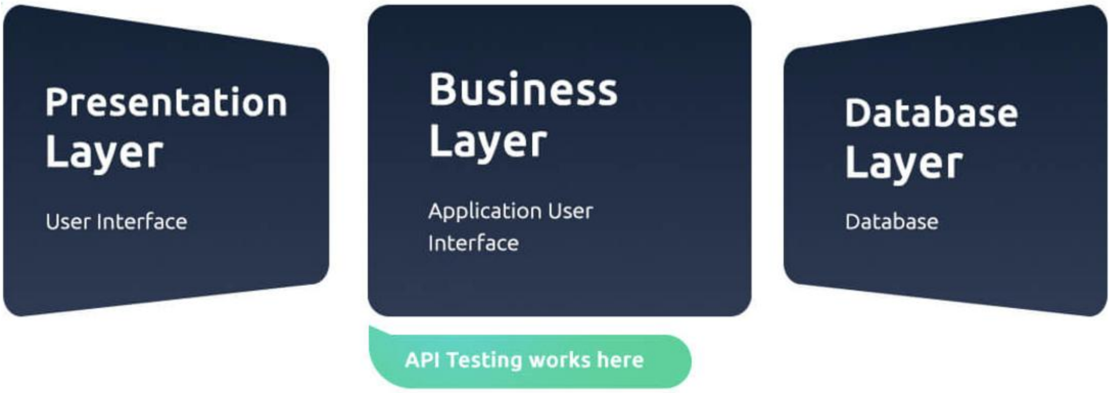
API testing is a software testing method that verifies the functionality, security, performance, and reliability of an application programming interface (API)

API is the mediator which helps to applications to communicate with each other. It is kind of logic written by developers using any programming language to perform something

API is a Software Interface that allows two applications to interact with each other without any user intervention

Testing the business logic of any application is called API, QA will test the same logic and called API testing

API testing is a part of back end testing like database



9. Types of API Testing

Open APIs: These types of APIs are publicly available to use. It has also not given any restriction to use them. So, they are also known as Public APIs. e.g. OAuth APIs from Google

Partner APIs: Specific rights or licenses needed to access this type of API because they are not available to the public

Internal APIs: Internal or private. These APIs are developed by companies to use in their internal systems It helps you to enhance the productivity of your teams

10. What is Responsive Testing?

To check the responsiveness of our website on multiple devices is simply called responsive testing

When user switches to one device to another device the contents of responsive websites adapt themselves according to the device UI, resolution etc factors

Responsive testing involves how a website or web application looks and behaves on different devices, screen sizes, and resolutions

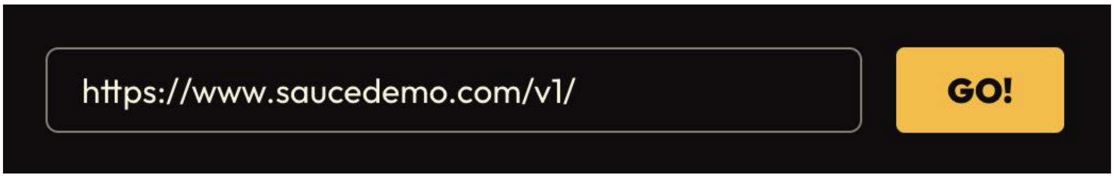
The goal of responsive testing is to ensure that the website or web application can be used effectively on various devices, including desktops, laptops, tablets, and smartphones

11. Which types of tools are available for Responsive Testing

Lembda Testing
Google Resizer
am I responsive
Pixel tuner

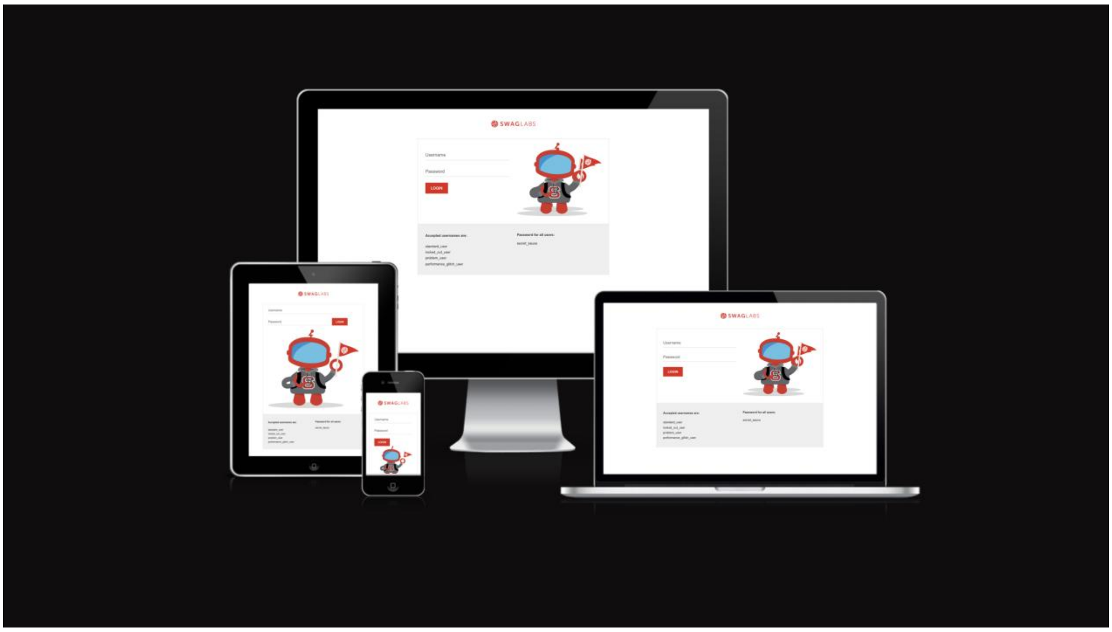
e..g., https://ui.dev/amiresponsive

Enter your URL in this box to search

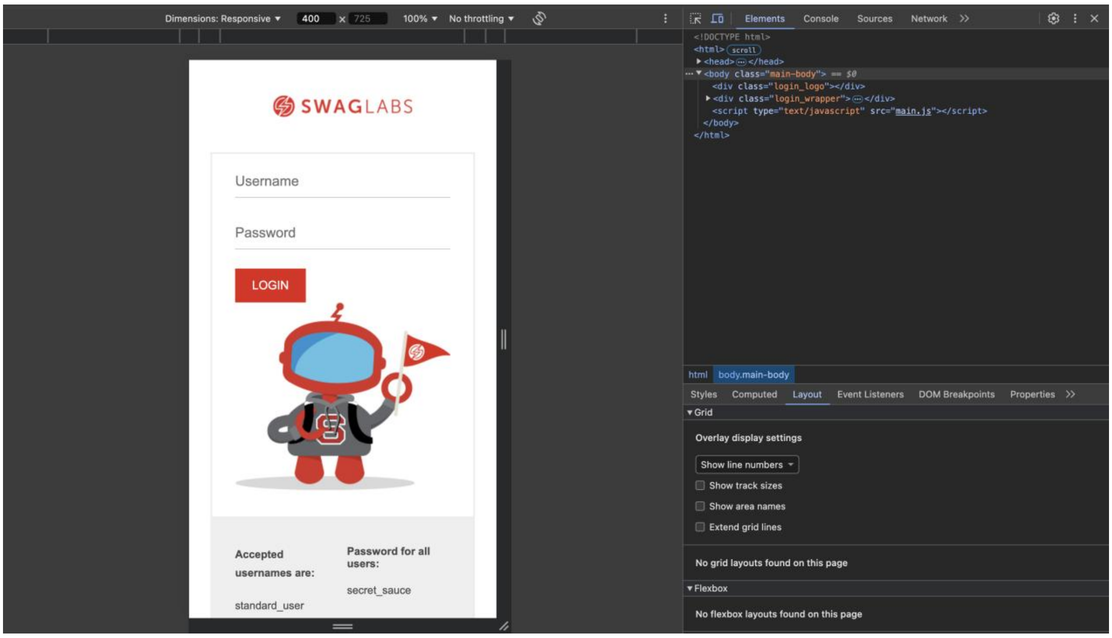


https://www.saucedemo.com/v1/

we get different response of the website on different devices



For Google chrome, you can right click in anywhere in the browser and select “Inspect”, you can check the screen where you can set the screen with multiple dimensions



Accepted usernames are:

standard_user

Password for all users:

secret_sauce

```
<!DOCTYPE html>
<html> scroll
 ▸ <head> ··· </head>
··· ▾ <body class="main-body"> == $0
     <div class="login_logo"></div>
   ▸ <div class="login_wrapper"> ··· </div>
     <script type="text/javascript" src="main.js"></script>
   </body>
</html>
```

Overlay display settings

Show line numbers ▾

□ Show track sizes

□ Show area names

□ Extend grid lines

No grid layouts found on this page

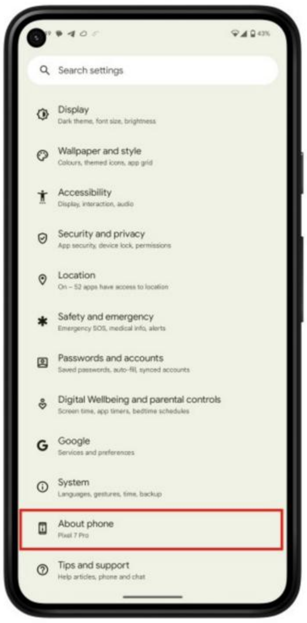
No flexbox layouts found on this page

12. What is the full form of .ipa, .apk

.ipa stands for iOS package, App international phonetic alphabet
.apk stands for Android Application Package

13. How to create step for to open the developer option mode ON?

The following example uses a Google Pixel 7 Pro running Android 13 shows steps to open the developer option mode ON



Search settings

Display
Dark theme, font size, brightness

Wallpaper and style
Colours, themed icons, app grid

Accessibility
Display, interaction, audio

Security and privacy
App security, device lock, permissions

Location
On – 52 apps have access to location

Safety and emergency
Emergency SOS, medical info, alerts

Passwords and accounts
Saved passwords, auto-fill, synced accounts

Digital Wellbeing and parental controls
Screen time, app timers, bedtime schedules

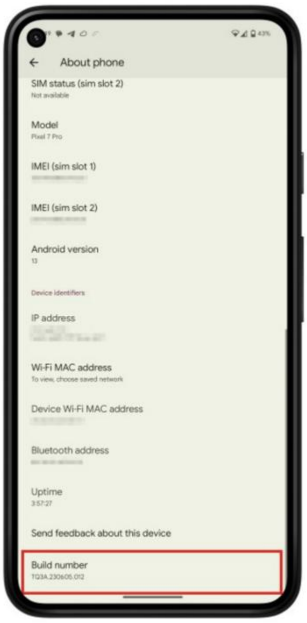
Google
Services and preferences

System
Languages, gestures, time, backup

About phone
Pixel 7 Pro

Tips and support
Help articles, phone and chat

Step: 1



About phone

SIM status (sim slot 2)
Not available

Model
Pixel 7 Pro

IMEI (sim slot 1)

IMEI (sim slot 2)

Android version
13

Device identifiers

IP address

Wi-Fi MAC address
To view, choose saved network

Device Wi-Fi MAC address

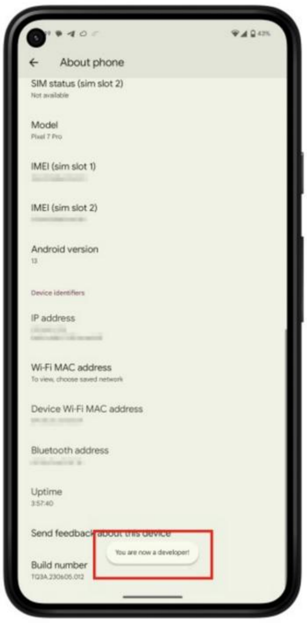
Bluetooth address

Uptime
3:57:27

Send feedback about this device

Build number
TQ3A.230605.012

Step: 2



About phone

SIM status (sim slot 2)
Not available

Model
Pixel 7 Pro

IMEI (sim slot 1)

IMEI (sim slot 2)

Android version
13

Device identifiers

IP address

Wi-Fi MAC address
To view, choose saved network

Device Wi-Fi MAC address

Bluetooth address

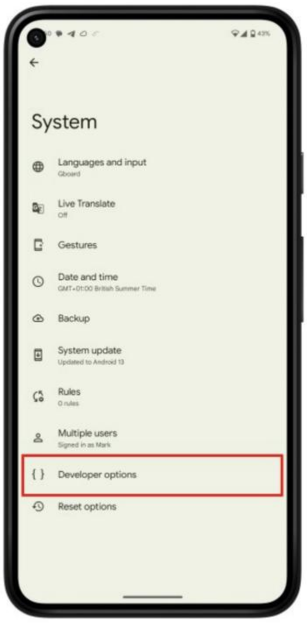
Uptime
3:57:40

Send feedback about this device

Build number
TQ3A.230605.012

You are now a developer!

Step: 3 & 4



System

Languages and input
Gboard

Live Translate
Off

Gestures

Date and time
GMT+01:00 British Summer Time

Backup

System update
Updated to Android 13

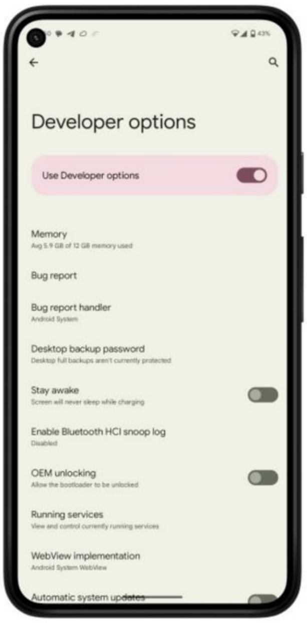
Rules
0 rules

Multiple users
Signed in as Mark

{ } Developer options

Reset options

Step: 5



Developer options

Use Developer options ●

Memory
Avg 5.9 GB of 12 GB memory used

Bug report

Bug report handler
Android System

Desktop backup password
Desktop full backups aren't currently protected

Stay awake ○
Screen will never sleep while charging

Enable Bluetooth HCI snoop log
Disabled

OEM unlocking ○
Allow the bootloader to be unlocked

Running services
View and control currently running services

WebView implementation
Android System WebView

Automatic system updates

Step: 6

Step 1: Go to *Settings* > *About phone*

Step 2: Scroll down to *Build number*

Step 3: Tap *Build number* seven times. After the first few taps, you should see the steps counting down until you unlock the developer options. You may also have to tap in your PIN for verification

Step 4: Once developer options are activated, you will see a message that reads, *You are now a developer*

Step 5: Go back to the *Settings* pane and head to *System*, where you will now find *Developer options* as an entry

Step 6: Tap it and toggle the switch on if it is not already, and from there, you can proceed to make adjustments to your phone

You can unlock the developer options on any Android smartphone or tablet by locating the build number in your *Settings* menu and tapping it multiple times. However, the exact location of the aforementioned build number may differ depending on your phone's manufacturer