



DBMS - Mini Project

Insurance Management System

Project Title:
INSURANCE MANAGEMENT SYSTEM

Submitted By:
Name : Chandan M S
SRN : PES2UG20CS094
SEC : 'B'

Short Description and Scope of the Project

The insurance company needs to keep track of details of its target companies, agents, policyholders, their premium payments and the various products that are available with it. Hence it is under tremendous pressure maintaining their day-to-day activities, which is currently being done manually.

Entire records have to be updated timely, even a slight mistake could complicate things. It is very difficult to handle bulk data since human memory is weaker than electronic counter part. It is time consuming to summarize these details to produce the reports.

The Insurance management system is a complete solution for organizations, which need to manage insurance for their vehicles, health , property, and other resources. Organizes and tracks insurance vendors and the policies provided under different coverage.

Insurance policy administration system consists of a mathematical notation that captures the relationship between policies and objects and the entities that manage policies for those objects. Hence there is need for an automated system, which can efficiently manage the company, records, provides instant access and one that improves the productivity.

As a result of this automated system, the activities of the company are performed with in the stipulated time and the reliable and efficient service is ensured to its users

MODULES

a) **Admin module:** This module allows only registered members to log in and manage the users

b) **users module:** This module allows to known the premium details of the users and product details.

PROGRAMMING ENVIRONMENT

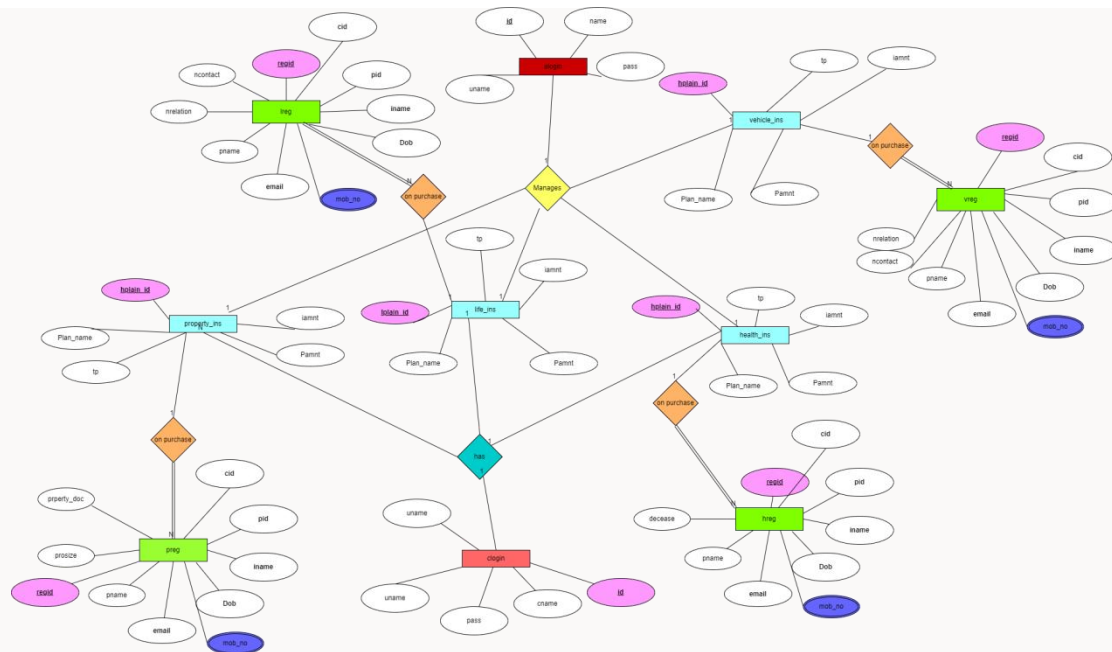
Operating System : Windows

Application Type : Web Application

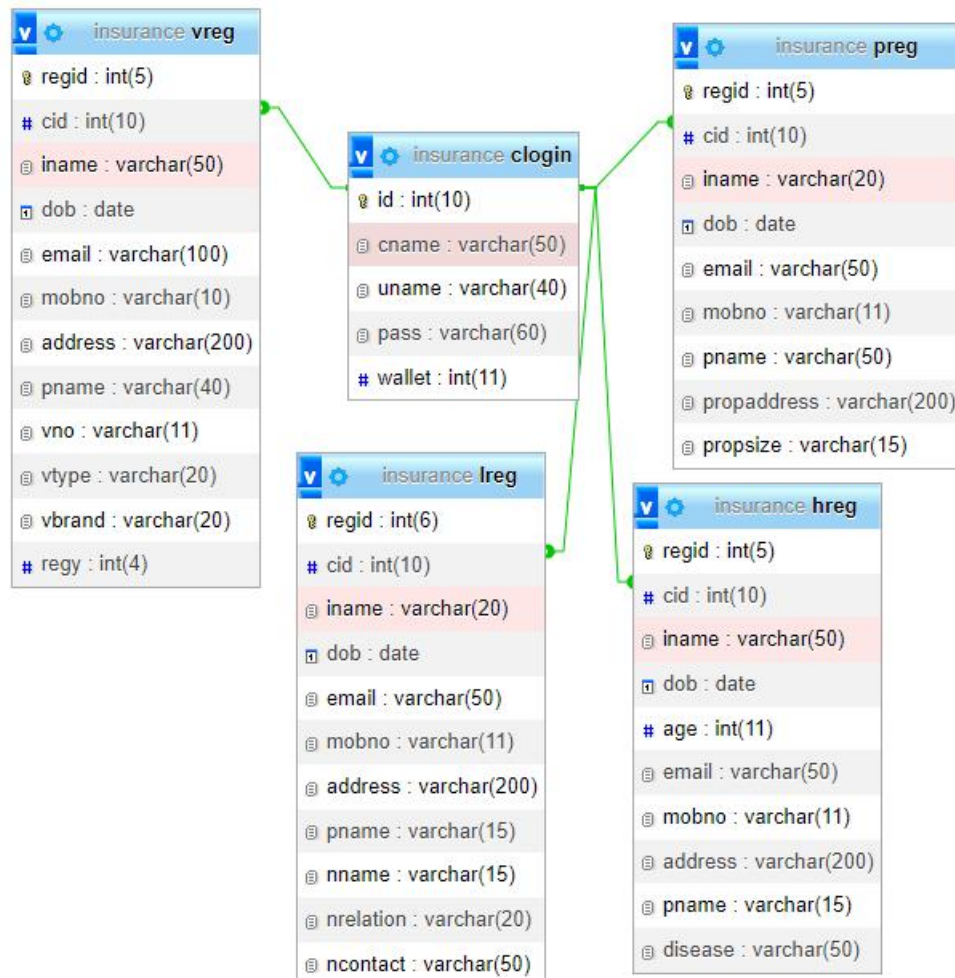
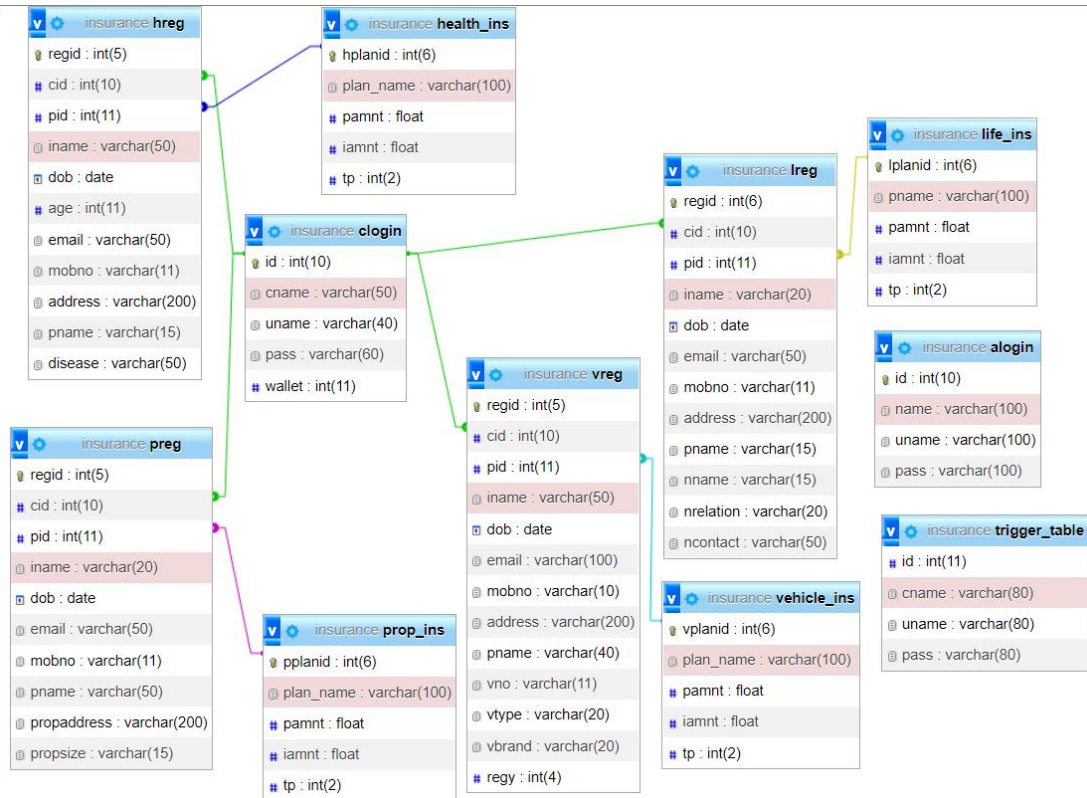
Front End : Php





Data Base : My sql





1












Relational Schema











  insurance health_ins
 hplanid : int(6)
 plan_name : varchar(100)
pamnt : float
iamnt : float
tp : int(2)

  insurance alogin
 id : int(10)
 name : varchar(100)
 uname : varchar(100)
 pass : varchar(100)

  insurance trigger_table
id : int(11)
 cname : varchar(80)
 uname : varchar(80)
 pass : varchar(80)

  insurance vehicle_ins
 vplanid : int(6)
 plan_name : varchar(100)
pamnt : float
iamnt : float
tp : int(2)

  insurance life_ins
 lplanid : int(6)
 pname : varchar(100)
pamnt : float
iamnt : float
tp : int(2)

  insurance prop_ins
 pplanid : int(6)
 plan_name : varchar(100)
pamnt : float
iamnt : float
tp : int(2)

DDL Statements - Building The Database

```
CREATE TABLE `alogin` (`id` int(10) NOT NULL, `name` varchar(100) NOT NULL,
`uname` varchar(100) NOT NULL, `pass` varchar(100) NOT NULL)
```

```
CREATE TABLE `clogin` (`id` int(10) NOT NULL, `cname` varchar(50) NOT NULL,
`uname` varchar(40) NOT NULL, `pass` varchar(60) NOT NULL)
```

```
CREATE TABLE `health_ins` (`hplanid` int(6) NOT NULL, `plan_name` varchar(100)
NOT NULL, `pamnt` float NOT NULL, `iamnt` float NOT NULL, `tp` int(2) NOT NULL)
```

```
CREATE TABLE `hreg` (`regid` int(5) NOT NULL, `cid` int(10) NOT NULL, `iname`
varchar(50) NOT NULL, `dob` date NOT NULL, `email` varchar(50) NOT NULL, `mobno`
varchar(11) NOT NULL, `address` varchar(200) NOT NULL, `pname` varchar(15) NOT
NULL, `disease` varchar(50) NOT NULL)
```

```
CREATE TABLE `life_ins` (`lplanid` int(6) NOT NULL, `pname` varchar(100) NOT
NULL, `pamnt` float NOT NULL, `iamnt` float NOT NULL, `tp` int(2) NOT NULL)
```

```
CREATE TABLE `lreg` (`regid` int(6) NOT NULL, `cid` int(10) NOT NULL, `iname`
varchar(20) NOT NULL, `dob` date NOT NULL, `email` varchar(50) NOT NULL,
`mobno` varchar(11) NOT NULL, `address` varchar(200) NOT NULL, `pname` varchar(15)
NOT NULL, `nname` varchar(15) NOT NULL, `nrelation` varchar(20) NOT
NULL, `ncontact` varchar(50) NOT NULL)
```

```
CREATE TABLE `preg` (`regid` int(5) NOT NULL, `cid` int(10) NOT NULL, `iname`
varchar(20) NOT NULL, `dob` date NOT NULL, `email` varchar(50) NOT NULL, `mobno`
varchar(11) NOT NULL, `pname` varchar(50) NOT NULL, `propaddress` varchar(200) NOT
NULL, `proptype` varchar(15) NOT NULL)
```

```
CREATE TABLE `prop_ins` (`pplanid` int(6) NOT NULL, `plan_name` varchar(100) NOT
NULL, `pamnt` float NOT NULL, `iamnt` float NOT NULL, `tp` int(2) NOT NULL)
```

```
CREATE TABLE `vehicle_ins` (`vplanid` int(6) NOT NULL, `plan_name` varchar(100)
NOT NULL, `pamnt` float NOT NULL, `iamnt` float NOT NULL, `tp` int(2) NOT NULL)
```

```
CREATE TABLE `vreg` (`regid` int(5) NOT NULL, `cid` int(10) NOT NULL, `iname`
varchar(50) NOT NULL, `dob` date NOT NULL, `email` varchar(100) NOT NULL, `mobno`
varchar(10) NOT NULL, `address` varchar(200) NOT NULL, `pname` varchar(40) NOT
NULL, `vno` varchar(11) NOT NULL, `vtype` varchar(20) NOT NULL, `vbrand` varchar(20)
NOT NULL, `regy` int(4) NOT NULL)
```

Populating Database

```
INSERT INTO `alogin` (`id`, `name`, `uname`, `pass`)
VALUES
(1, 'admin', 'admin', '12345'),
(3, 'chandan', 'chandan', 'abcd');
```

```
INSERT INTO `clogin` (`id`, `cname`, `uname`, `pass`)
VALUES
(1, 'Sandhya', 'sandy', '12345'),
(2, 'lahari', 'laharisai', '12345');
```

```
INSERT INTO `health_ins` (`hplanid`, `plan_name`, `pamnt`, `iamnt`, `tp`)
VALUES
(3, 'Reliance Critical Illness', 1500, 50000, 4),
(4, 'SBI arogya premier', 1000, 12000, 1),
(5, 'Star Family health', 2000, 25000, 2),
(6, 'Digit Health', 200000, 2500000, 8);
```

```
INSERT INTO `hreg` (`regid`, `cid`, `iname`, `dob`, `email`, `mobno`, `address`, `pname`,
`disease`)
VALUES
(6, 1, 'Manasa', '1996-08-26', 'Man@gmail.com', '9312456807', 'RR nagar , Bangalore-60',
'Reliance Critic', 'NONE'),
(7, 1, 'YASH', '1979-12-25', 'Tabbu@gmail.com', '7123456890', 'Vijaynagar , Bagalore-40',
'Star Family hea', 'Covid'),
(8, 1, 'nidhi', '1983-06-09', 'n@gmail.com', '8312456790', 'Honnavaara-80', 'SBI arogya prem',
'NONE');
```

```
INSERT INTO `life_ins` (`lplanid`, `pname`, `pamnt`, `iamnt`, `tp`) VALUES(2, 'Term',
2500, 25000, 2),(3, 'Retirement plans', 3000, 35000, 5);
```

```
INSERT INTO `lreg` (`regid`, `cid`, `iname`, `dob`, `email`, `mobno`, `address`, `pname`,
`nname`, `nrelation`, `ncontact`)
VALUES
(6, 1, 'Venkat', '1991-12-16', 'Vc@gmail.com', '8724156728', 'Mysore-50', 'Term', 'Vaishnavi',
'wife', '9821023456'),
(7, 1, 'Hamsa', '1997-12-14', 'Ham@gmail.com', '7123456890', 'Girinagar, Bangalore-49',
'Retirement plan', 'Hitesh', 'Cousin', '9821023447');
```

```
INSERT INTO `preg` (`regid`, `cid`, `iname`, `dob`, `email`, `mobno`, `pname`,
`propaddress`, `propsize`)
VALUES
```



```
(4, 1, 'Rakesh', '1988-02-21', 'Rocky@gmail.com', '9312456234', 'Home Structure',  
'Harinagar, Bangalore-39', '50X60');
```

```
INSERT INTO `prop_ins` (`pplanid`, `plan_name`, `pamnt`, `iamnt`, `tp`)  
VALUES  
(2, 'Home Structure', 50000, 2000000, 30),  
(3, 'Contents', 60000, 12000000, 12);
```

```
INSERT INTO `vehicle_ins` (`vplanid`, `plan_name`, `pamnt`, `iamnt`, `tp`)  
VALUES  
(9047, 'Third-Party', 1500, 55000, 3);
```


Join Queries

Showcase at least 4 join queries

Write the query in English Language, Show the equivalent SQL statement and also a screenshot of the query and the results

1) INNER JOIN

Joining clogin and hreg on cid who has registered in “SBI arogya prem”

SQL query

```
SELECT * FROM clogin INNER JOIN hreg ON id=cid WHERE pname="SBI arogya prem"
```



Submit

id	cname	uname	pass	wallet	regid	cid	pid	iname	dob	age	email	mobno	address	pname	disease
1	Sandhya	sandy	12345	50000	8	1	0	nidhi	1983-06-09	39	n@gmail.com	8312456790	Honnavara-80	SBI arogya prem	NONE
4	chandan	chandan@123	1233	50000	9	4	0	abc	2022-11-08	45	chandan@123	054054	54uyku,g	SBI arogya prem	yukgl
6	chand	chandan@123	987	50000	16	6	4	purchase	2002-06-07	20	chandan@123	6360238845	purchase	SBI arogya prem	purchase
6	chand	chandan@123	987	50000	17	6	7	new	2002-06-07	20	chandan@123	6360238845	csd	SBI arogya prem	na

✓ Showing rows 0 - 2 (3 total, Query took 0.0004 seconds.)

```
SELECT * FROM clogin INNER JOIN hreg ON id=cid WHERE pname="SBI arogya prem";
```

☐ Profiling [\[Edit Inline \]](#) [\[Edit \]](#) [\[Explain SQL \]](#) [\[Create PHP code \]](#) [\[Refresh \]](#)

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

id	cname	uname	pass	wallet	regid	cid	pid	iname	dob	age	email	mobno	address	pname	disease
1	Sandhya	sandy	12345	50000	8	1	0	nidhi	1983-06-09	39	n@gmail.com	8312456790	Honnavara-80	SBI arogya prem	NONE
4	chandan	chandan@123	1233	50000	9	4	0	abc	2022-11-08	0	chandan@123	054054	54uyku,g	SBI arogya prem	yukgl
6	chand	chandan@123	987	50000	16	6	4	purchase	2002-06-07	20	chandan@123	6360238845	purchase	SBI arogya prem	purchase

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

2) RIGHT JOIN

Jonning clogin and lreg on cid who has registered for the plan “Retirement plan”

Query Box

SQL query

```
SELECT * FROM clogin RIGHT JOIN lreg ON  
id=cid WHERE pname="Retirement plan";
```



Submit

id	cname	uname	pass	wallet	regid	cid	pid	iname	dob	email	mobno	address	pname	nname	nrelation	ncontact
1	Sandhya	sandy	12345	50000	7	1	0	Hamsa	1997-12-14	Ham@gmail.com	7123456890	Giriagar, Bangalore-49	Retirement plan	Hitesh	Cousin	9821023447
6	chand	chandan@123	987	50000	8	6	3	purchase	2002-06-07	chandan@123	6360238845	purchase	Retirement plan	chandan	ghgc	6360238845

✓ Showing rows 0 - 1 (2 total, Query took 0.0005 seconds.)

```
SELECT * FROM clogin RIGHT JOIN lreg ON id=cid WHERE pname="Retirement plan";
```

☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)] [[Create PHP code](#)] [[Refresh](#)]

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

id	cname	uname	pass	wallet	regid	cid	pid	iname	dob	email	mobno	address	pname	nname	nrelation	ncontact
1	Sandhya	sandy	12345	50000	7	1	0	Hamsa	1997-12-14	Ham@gmail.com	7123456890	Girinagar, Bangalore-49	Retirement plan	Hitesh	Cousin	9821023447
6	chand	chandan@123	987	50000	8	6	3	purchase	2002-06-07	chandan@123	6360238845	purchase	Retirement plan	chandan	ghgc	6360238845

⚠ Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available. ⓘ

✓ Showing rows 0 - 2 (3 total, Query took 0.0004 seconds.)

```
SELECT * FROM vehicle_ins RIGHT JOIN join_table ON vplanid=pid;
```

☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)] [[Create PHP code](#)] [[Refresh](#)]

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

vplanid	plan_name	pamnt	iamnt	tp	regid	cid	pid	iname	dob	email	mobno	address	pname	vno	vtype	vbrand	regy
NULL	NULL	NULL	NULL	NULL	98472	1	9561	Lahari	1990-04-06	lorry@gmail.com	9846127830	Konankunte , Bangalore-62	SBI vehicle	KA-07-SS-06	4-Wheeler	Honda-120	201
NULL	NULL	NULL	NULL	NULL	98473	2	9241	abc	2022-09-06	insaneopg@gmail.com	145786	sddsdsd	Third-Party	dsds	2-Wheeler	dsd	21
9051	xyz	500	50000	20	98475	6	9051	purchase	2002-06-07	chandan@123	6360238845	purchase	xyz	6556	2-Wheeler	ghgh	202

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Query Box

SQL query

```
SELECT * FROM clogin LEFT JOIN lreg ON  
id=cid WHERE pname="Retirement plan";
```

id	ename	uname	pass	walter	regid	cid	pid	iname	dob	email	mobno	address	pname	uname	nrelation	ncontact
1	Sandhya	sandy	12345	50000	7	1	0	Hamsa	1997-12-14	Ham@gmail.com	7123456890	Girinagar, Bangalore-49	Retirement plan	Hitesh	Cousin	9821023447
6	chand	chandan@123	987	50000	8	6	3	purchase	2002-06-07	chandan@123	6360238845	purchase	Retirement plan	chandan	ghgc	6360238845

Showing rows 0 - 1 (2 total, Query took 0.0012 seconds.)

```
SELECT * from clogin LEFT OUTER JOIN lreg ON id=cid WHERE pname="Retirement plan";
```

☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)] [[Create PHP code](#)] [[Refresh](#)]

☐ Show all | Number of rows: Filter rows: Sort by key:

Extra options

d	cname	uname	pass	wallet	regid	cid	pid	iname	dob	email	mobno	address	pname	nname	nrelation	ncontact
1	Sandhya	sandy	12345	50000	7	1	0	Hamsa	1997-12-14	Ham@gmail.com	7123456890	Girinagar, Bangalore-49	Retirement plan	Hitesh	Cousin	9821023447
6	chand	chandan@123	987	50000	8	6	3	purchase	2002-06-07	chandan@123	6360238845	purchase	Retirement plan	chandan	ghgc	6360238845

[illegible]

4) FULL JOIN

Query Box

SQL query

```
SELECT * FROM vehicle_ins LEFT JOIN vreg
ON vplanid=pid UNION SELECT * FROM vehicle_ins RIGHT
JOIN vreg ON vplanid=pid;
```

Submit

vplanid	plan_name	pamnt	lamnt	tp	regid	cid	pid	iname	dob	email	mobno	address	pname	vno	vtype	vbrand	regy
9047	Third-Party	1500	55000	3													
9051	xyz	500	50000	20	98475	6	9051	purchase	2002-06-07	chandan@123	6360238845	purchase	xyz	6556	2-Wheeler	ghgh	2022
					98471	1	0	sandhya	1993-11-23	san@gmail.com	9123456780	Chandra Layout, Bangalore-72	Third-Party	KA-05-JJ-34	2-Wheeler	Honda-Activa	2021
					98472	1	0	Lahari	1990-04-06	lorry@gmail.com	9846127830	Konankunte , Bangalore-62	SBI vehicle	KA-07-SS-06	4-Wheeler	Honda-120	2019
					98473	2	0	abc	2022-09-06	insaneopg@gmail.com	145786	sddsdsd	Third-Party	dsds	2-Wheeler	dsd	214

Showing rows 0 - 4 (5 total, Query took 0.0034 seconds.)

```
SELECT * FROM vehicle_ins LEFT JOIN vreg ON vplanid=pid UNION SELECT * FROM vehicle_ins RIGHT JOIN vreg ON vplanid=pid;
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

vplanid	plan_name	pamnt	lamnt	tp	regid	cid	pid	iname	dob	email	mobno	address	pname	vno	vtype	vbrand	regy
9047	Third-Party	1500	55000	3	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NU
9051	xyz	500	50000	20	98475	6	9051	purchase	2002-06-07	chandan@123	6360238845	purchase	xyz	6556	2-Wheeler	ghgh	2022
NULL	NULL	NULL	NULL	NULL	98471	1	0	sandhya	1993-11-23	san@gmail.com	9123456780	Chandra Layout, Bangalore-72	Third-Party	KA-05-JJ-34	2-Wheeler	Honda-Activa	2021
NULL	NULL	NULL	NULL	NULL	98472	1	0	Lahari	1990-04-06	lorry@gmail.com	9846127830	Konankunte , Bangalore-62	SBI vehicle	KA-07-SS-06	4-Wheeler	Honda-120	2019
NULL	NULL	NULL	NULL	NULL	98473	2	0	abc	2022-09-06	insaneopg@gmail.com	145786	sddsdsd	Third-Party	dsds	2-Wheeler	dsd	214

Console | Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Aggregate Function

1) COUNT

Counting the number of people registered in particular plan

```
SELECT COUNT(regid),pname FROM hreg GROUP BY pname;
```

COUNT(regid)	pname
1	Reliance Critic
3	SBI arogya prem
4	Star Family hea

✓ Showing rows 0 - 2 (3 total, Query took 0.0007 seconds.)

```
SELECT COUNT(regid),pname FROM hreg GROUP BY pname;
```

☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)] [[Create PHP code](#)] [[Refresh](#)]

☐ Show all | Number of rows: 25 ▼ Filter rows:

Extra options

COUNT(regid)	pname
1	Reliance Critic
3	SBI arogya prem
4	Star Family hea

☐ Show all | Number of rows: 25 ▼ Filter rows:

2) MAX

Finding the max time period of the existing plan

```
SELECT max(tp) FROM health_ins;
```

max(tp)
3

✓ Showing rows 0 - 0 (1 total, Query took 0.0005 seconds.)

```
SELECT max(tp) FROM `health_ins`;
```

☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)] [[Create PHP code](#)]

☐ Show all | Number of rows: 25 Filter rows:

Extra options

max(tp)

8

3) MIN

Finding min age among the hreg

```
SELECT min(age) FROM hreg;
```

min(age)
13

✓ Showing rows 0 - 0 (1 total, Query took 0.0023 seconds.)

```
SELECT min(age) FROM `hreg`;
```

☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)] [[Create PHP code](#)] [[Refresh](#)]

☐ Show all | Number of rows: 25 ▼ Filter rows:

Extra options

min(age)

10

☐ Show all | Number of rows: 25 ▼ Filter rows:

4) AVG

```
SELECT avg(pamnt) FROM health_ins;
```

avg(pamnt)

1000

✓ Showing rows 0 - 0 (1 total, Query took 0.0011 seconds.)

```
SELECT avg(pamnt) FROM `health_ins`;
```

☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)] [[Create PHP code](#)] [[Refresh](#)]

☐ Show all | Number of rows: 25 ▼ Filter rows:

Extra options

avg(pamnt)

51125

Set Operations

Union:

`SELECT * FROM clogin UNION SELECT * FROM health_ins; SELECT * FROM clogin UNION SELECT * FROM health_ins;`

id	cname	uname	pass	wallet
1	Sandhya	sandy	12345	50000
2	lahari	laharisai	12345	50000
4	chandan	chandan@123	1233	50000
5	chandan	chandu	1245	2
6	chand	chandan@123	987	50000
7	chandan	chandan@12354	456	50000
4	SBI arogya premier	1000	12000	3
7	SBI arogya premier	1000	12000	3

id	cname	uname	pass	wallet
1	Sandhya	sandy	12345	50000
2	lahari	laharisai	12345	50000
4	chandan	chandan@123	1233	50000
5	chandan	chandu	1245	2
6	chand	chandan@123	987	50000
4	SBI arogya premier	1000	12000	3
6	SBI arogya premier	1000	12000	2
7	SBI arogya premier	1000	12000	3

NOT IN:

```
SELECT * FROM clogin WHERE id NOT IN( SELECT hplanid FROM health_ins);
```

id	cname	uname	pass	wallet
1	Sandhya	sandy	12345	50000
2	lahari	laharisai	12345	50000
5	chandan	chandu	1245	2
6	chand	chandan@123	987	50000

✓ Showing rows 0 - 2 (3 total, Query took 0.0026 seconds.)

```
SELECT * FROM clogin WHERE id NOT IN( SELECT hplanid FROM health_ins);
```

☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)] [[Create PHP code](#)] [[Refresh](#)]

☐ Show all | Number of rows: Filter rows: Sort by ke

Extra options

					id	cname	uname	pass	wallet
<input type="checkbox"/>					1	Sandhya	sandy	12345	50000
<input type="checkbox"/>					2	lahari	laharisai	12345	50000
<input type="checkbox"/>					5	chandan	chandu	1245	2

☐ Check all With selected: Edit Copy Delete Export

Union All:

```
SELECT * FROM clogin UNION all select * from health_ins;
```

id	cname	uname	pass	wallet
1	Sandhya	sandy	12345	50000
2	lahari	laharisai	12345	50000
4	chandan	chandan@123	1233	50000
5	chandan	chandu	1245	2
6	chand	chandan@123	987	50000
7	chandan	chandan@12354	456	50000
4	SBI arogya premier	1000	12000	3
7	SBI arogya premier	1000	12000	3

✓ Showing rows 0 - 7 (8 total, Query took 0.0004 seconds.)

`SELECT * FROM clogin UNION ALL SELECT * FROM health_ins;`

☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)] [[Create PHP code](#)] [[Refresh](#)]


☐ Show all | Number of rows: Filter rows:

Extra options

id	cname	uname	pass	wallet
1	Sandhya	sandy	12345	50000
2	lahari	laharisai	12345	50000
4	chandan	chandan@123	1233	50000
5	chandan	chandu	1245	2
6	chand	chandan@123	987	50000
4	SBI arogya premier	1000	12000	3
6	SBI arogya premier	1000	12000	2
7	SBI arogya premier	1000	12000	3

Functions And Procedures

Function:

	Name	Type	Returns				
<input type="checkbox"/>	age_class	FUNCTION	varchar(20)	 Edit	 Execute	 Export	 Drop

```
DELIMITER $$
CREATE FUNCTION age_class(
  age int
)
RETURNS VARCHAR(20)
DETERMINISTIC
BEGIN
  DECLARE Age_group VARCHAR(20);
  IF age < 18 THEN
    SET Age_group = 'youth';
  ELSEIF (age <= 18 AND
    age >= 39) THEN
    SET Age_group = 'young adult';
  ELSEIF (age >= 40 AND
    age<=49)THEN
    SET Age_group = 'early middle-aged adults';
  ELSEIF (age >= 50 AND
    age<=59)THEN
    SET Age_group = 'late middle-aged adults';
  ELSEIF (age > 60)THEN
    SET Age_group = 'old peoples';
  END IF;
  -- return the customer occupation
  RETURN (Age_group);
END$$
```

```
SELECT age,age_class(age) AS age_group FROM hreg;
```

age	age_group
23	young adult
13	youth
39	young adult
45	early middle-aged ad
15	youth
55	late middle-aged adu
70	old peoples
20	young adult

✔ Showing rows 0 - 7 (8 total, Query took 0.0015 seconds.)

`SELECT age,age_class(age) AS age_Group FROM `hreg`;`

☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)] [[Create PHP code](#)] [[Refresh](#)]




☐ Show all | Number of rows: 25 Filter rows: Search this table

Extra options

				age	age_Group
<input type="checkbox"/>				23	young adult
<input type="checkbox"/>				13	youth
<input type="checkbox"/>				39	young adult
<input type="checkbox"/>				45	early middle-aged ad
<input type="checkbox"/>				15	youth
<input type="checkbox"/>				55	late middle-aged adu
<input type="checkbox"/>				70	old peoples
<input type="checkbox"/>				20	young adult

2) Procedure

3)

	Name	Type	Returns	
<input type="checkbox"/>	procedure_cursor	PROCEDURE		 Edit  Execute  Export  Drop

```
DELIMITER $$
CREATE PROCEDURE procedure_cursor()
BEGIN
DECLARE plan_n VARCHAR(20);
DECLARE planid,p INT;
DECLARE amnt,inamnt FLOAT;
DECLARE done INT DEFAULT 0;
declare cur_1 cursor for select * from health_ins WHERE hplanid=3;
declare continue handler for not found set done=1;
open cur_1;
get_data:loop
fetch cur_1 into planid,plan_n,amnt,inamnt,p;
select planid,plan_n,amnt,inamnt,p;
if done=1 then
leave get_data;
end if;
end loop get_data;
End
```

```
CALL `procedure_cursor`();
```

Execution results of routine `procedure_cursor`

planid	plan_n	amnt	inamnt	p
--------	--------	------	--------	---

4	SBI arogya premier	1000	12000	1
---	--------------------	------	-------	---

planid	plan_n	amnt	inamnt	p
--------	--------	------	--------	---

4	SBI arogya premier	1000	12000	1
---	--------------------	------	-------	---

Triggers And Cursors

Triggers:

I) Trigger is created on hreg to update the age from DOB before insert into table

```
CREATE TRIGGER age_update_hreg
BEFORE INSERT On hreg
FOR EACH ROW SET new.age=YEAR(CURRENT_TIMESTAMP)-YEAR(new.dob);
```

<input type="checkbox"/>	Edit	Copy	Delete	15	6	0	chandan	2022-11-08	70	chandan@gmail.com	6360238845	banglore	Star Family	hea	no
<input type="checkbox"/>	Edit	Copy	Delete	16	6	4	purchase	2002-06-07	20	chandan@123	6360238845	purchase	SBI arogya	prem	purchase

II) Trigger to Keep the backup clogin table so when ever new customer register the entry is made in backup table too

```
CREATE TRIGGER after_insert
BEFORE INSERT On clogin
BEGIN
INSERT INTO trigger_table VALUES(new.id,new.cname,new.uname,new.pass);
END
```

id	cname	uname	pass
5	chandan	chandu	1245
6	chand	chandan@123	987

Triggers

☐ Check all

	Name	Table	Time	Event	
<input type="checkbox"/>	after_insert	clogin	AFTER	INSERT	<input type="button" value="Edit"/> <input type="button" value="Export"/> <input type="button" value="Drop"/>
<input type="checkbox"/>	age_update_hreg	hreg	BEFORE	INSERT	<input type="button" value="Edit"/> <input type="button" value="Export"/> <input type="button" value="Drop"/>

Cursor :

```
DELIMITER $$
CREATE PROCEDURE procedure_cursor()
BEGIN
DECLARE plan_n VARCHAR(20);
DECLARE planid,p INT;
DECLARE amnt,inamnt FLOAT;
DECLARE done INT DEFAULT 0;
declare cur_1 cursor for select * from health_ins WHERE hplanid=3;
declare continue handler for not found set done=1;
open cur_1;
get_data:loop
fetch cur_1 into planid,plan_n,amnt,inamnt,p;
select planid,plan_n,amnt,inamnt,p;
if done=1 then
leave get_data;
end if;
end loop get_data;
End
```

```
CALL `procedure_cursor`();
```

Execution results of routine `procedure_cursor`

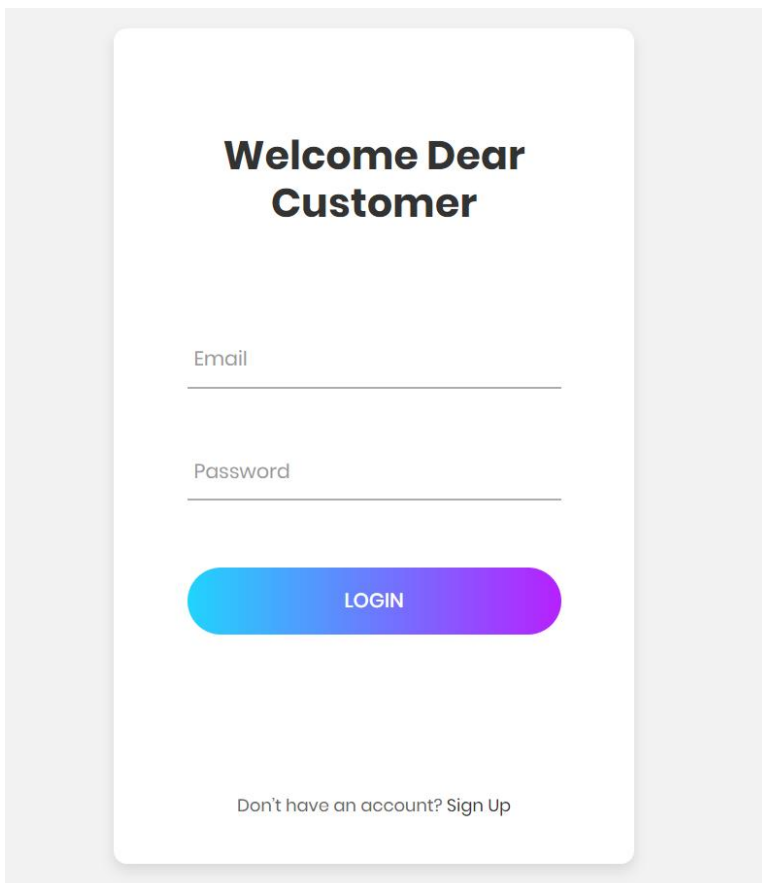
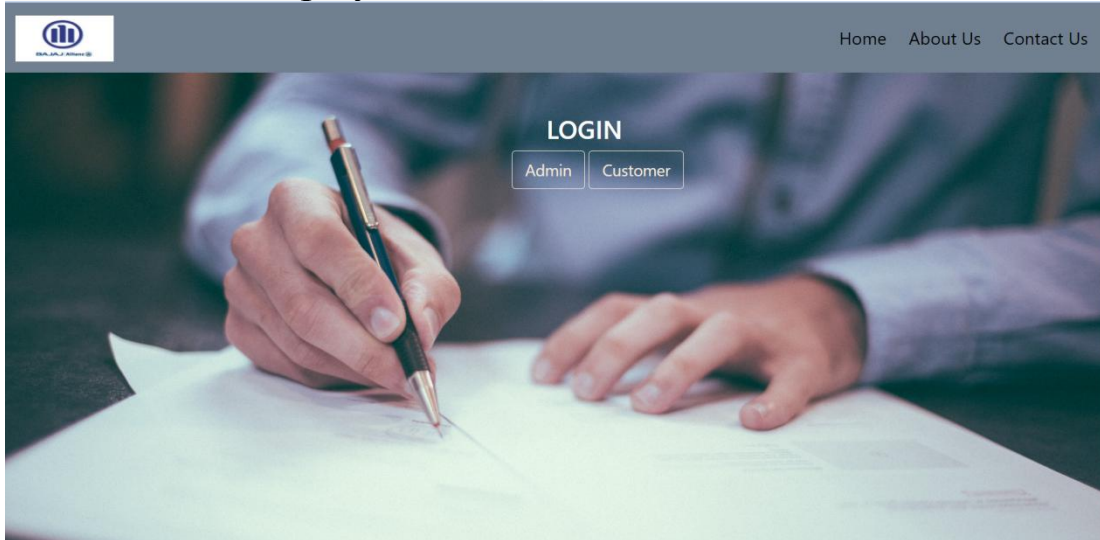
planid	plan_n	amnt	inamnt	p
4	SBI arogya premier	1000	12000	1

planid	plan_n	amnt	inamnt	p
4	SBI arogya premier	1000	12000	1


Developing Frontend

The frontend should support

- 1.Addition, Modification and Deletion of records from any chosen table
- 2.There should be an window to accept and run any SQL statement and display the result



CUSTOMER OPTIONS:

Plans Registrations Logout

[BACK](#)

SBI arogya premier

Pay **Rs.1000**

To get sum insured of **12000**

For **3** Years.

[PURCHASE](#)

SBI arogya premier

Pay **Rs.1000**

To get sum insured of **12000**

For **2** Years.

[PURCHASE](#)

SBI arogya premier

Pay **Rs.1000**

To get sum insured of **12000**

For **3** Years.

[PURCHASE](#)

NAME :

Date Of Birth :

E mail :

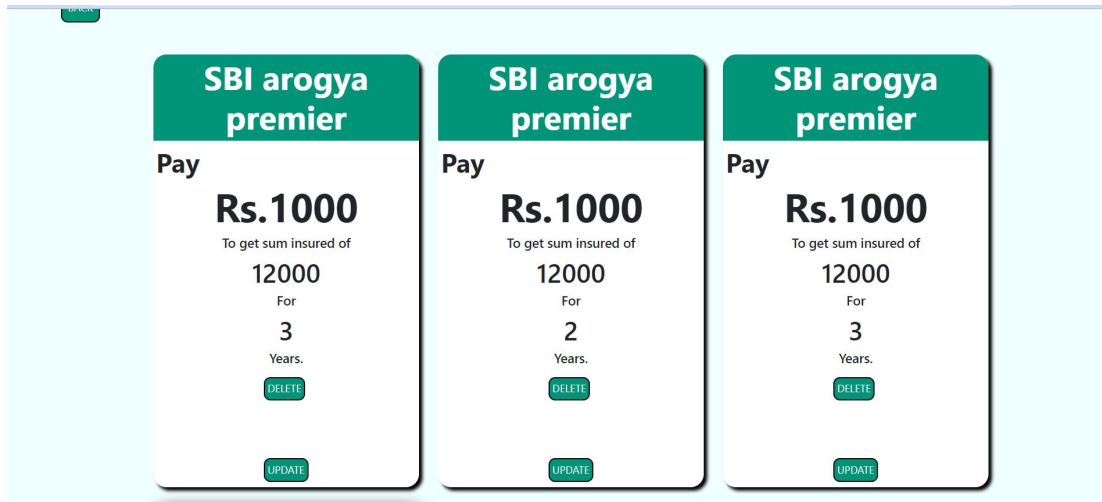
Mobile No :

Address :

Diseases :

[SUBMIT](#)

ADMIN OPTIONS:



A form for adding a new plan, set against a dark background. It contains four input fields: 'PLAN NAME:', 'PLAN PRICE:', 'SUM INSURED AMOUNT:', and 'TIME PERIOD:'. Below these fields is a 'SUBMIT' button.

UPDATE:

A form for updating a plan. It features a single input field containing the number '4'. Below the input field is an 'UPDATE' button.

INSERT



DELETE:

12000

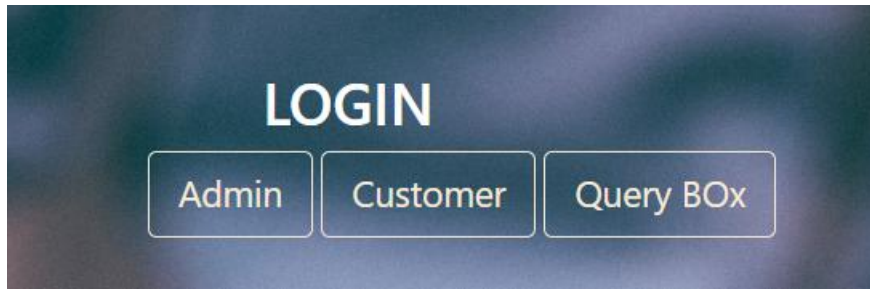
For

3

Years.

DELETE

window to accept and run any SQL statement and display the result



Query Box

SQL query

Submit

Query Box

SQL query

Submit

regid	cid	pid	iname	dob	age	email	mobno	address	puname	disease
6	1	0	Manasa	1996-08-26	23	Man@gmail.com	9312456807	RR nagar , Bangalore-60	Reliance Critic	NONE
7	1	0	YASH	1979-12-25	13	Tabbu@gmail.com	7123456890	Vijaynagar , Bagalore-40	Star Family hea	Covid
8	1	0	nidhi	1983-06-09	39	n@gmail.com	8312456790	Honnavaara-80	SBI arogya prem	NONE
9	4	0	abc	2022-11-08	45	chandan@123	054054	54uyku.g	SBI arogya prem	yukgl
10	4	0	kmuy,mi7	2022-11-16	15	chandan@123	6km67k	6jm67k	Star Family hea	6ykJ67
14	6	0	new	2002-06-07	55	chandan@123	6360238845	last	Star Family hea	no
15	6	0	chandan	2022-11-08	70	chandan@gmail.com	6360238845	banglore	Star Family hea	no
16	6	4	purchase	2002-06-07	20	chandan@123	6360238845	purchase	SBI arogya prem	purchase
17	6	7	new	2002-06-07	20	chandan@123	6360238845	csd	SBI arogya prem	na
18	6	7	new	2002-06-07	20	chandan@123	6360238845	csd	SBI arogya prem	na
19	6	7	chadan	2002-06-07	20	chandan@123		dd	SBI arogya prem	na
20	6	7	purchase	2002-06-07	20	chandan@123	6360238845	a	SBI arogya prem	no

Query :

SELECT cname,uname FROM clogin WHERE cname=”chandan”;

cname	uname
chandan	chandan@123
chandan	chandu
chandan	chandan@12354