

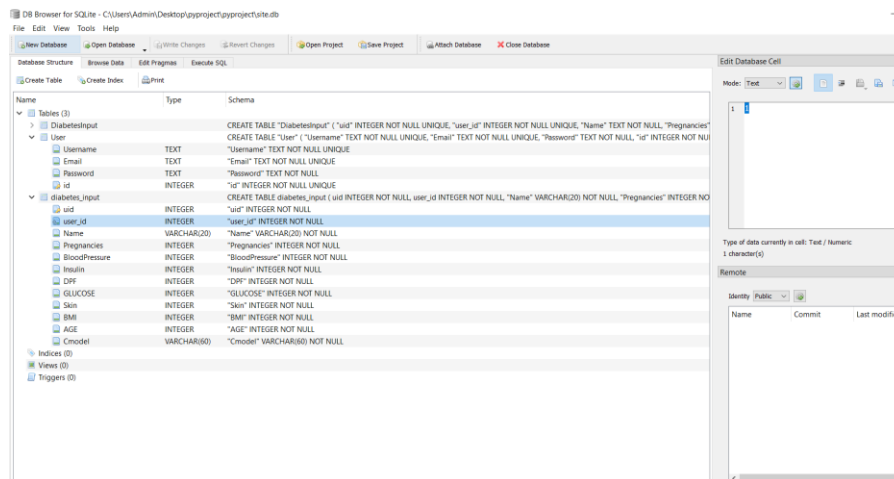
Internet and Web Programming Project Diabetes Prediction System Review-2

Name-Ritika Raj

Reg No- 18BCE2246

Query Document:

1.Create Table:



- User Table

```
CREATE TABLE "User" (  
    "id"    INTEGER NOT NULL UNIQUE,  
    "Username" TEXT NOT NULL UNIQUE,  
    "Email"    TEXT NOT NULL UNIQUE,  
    "Password" TEXT NOT NULL UNIQUE,
```

```
PRIMARY KEY("id" AUTOINCREMENT)  
);
```

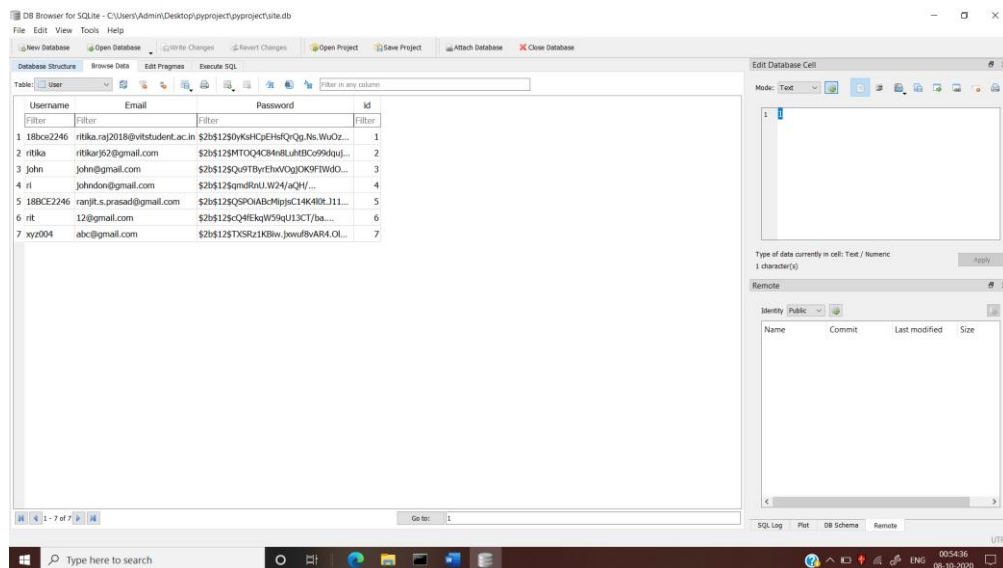
Flask Code:

```
class User(db.Model, UserMixin):  
    id = db.Column(db.Integer, primary_key=True)  
    username = db.Column(db.String(20), unique=True, nullable=False)  
    email = db.Column(db.String(120), unique=True, nullable=False)  
    password = db.Column(db.String(60), nullable=False)  
    diab = db.relationship('DiabetesInput', lazy=True)
```

2. Input Data:

- Inputting data from command prompt :

Original Database:



Username	Email	Password	Id
18bce2246	ritika.ra2018@vitsstudent.ac.in	\$2b\$12\$DyKsHqCjEHdQcQp.Ns.Wu.Qz...	1
ritika	ritika.r2@gmail.com	\$2b\$12\$MTQACR4n4LUhBC6994jd...	2
John	john@gmail.com	\$2b\$12\$Q9TByrEhvVOjOK9FTWdO...	3
ri	johnson@gmail.com	\$2b\$12\$Qnd8tUjW24/xCH/...	4
18bce2246	ranjit.s.prasad@gmail.com	\$2b\$12\$QSP0AB3MyjsC14K40L311...	5
rit	12@gmail.com	\$2b\$12\$QHEkqW59qU13CT/ba...	6
xyz004	abc@gmail.com	\$2b\$12\$TKSRz1K8Iw.Jeuf8vAR4.OL...	7

```

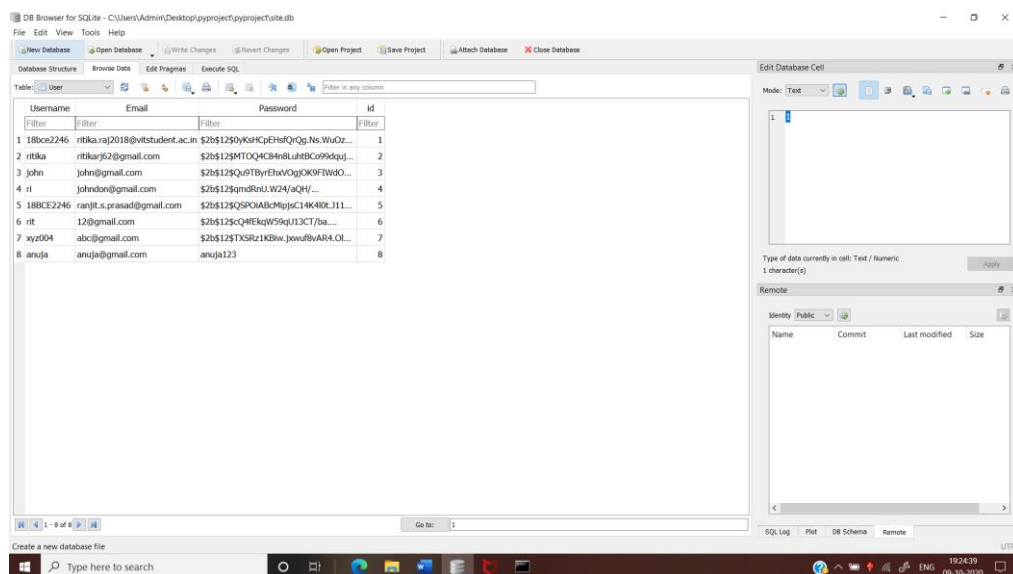
Command Prompt - python

C:\Users\Admin\Desktop>cd pyproject

C:\Users\Admin\Desktop\pyproject>python
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> from pyproject import db
C:\Users\Admin\AppData\Local\Programs\Python\Python37-32\lib\site-packages\flask_sqlalchemy\__init__.py:834: FSADeprecat
ionWarning: SQLALCHEMY_TRACK_MODIFICATIONS adds significant overhead and will be disabled by default in the future. Set
it to True or False to suppress this warning.
  'SQLALCHEMY_TRACK_MODIFICATIONS adds significant overhead and '
>>> from pyproject.models import User, DiabetesInput
>>> user=user.query.all()
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'user' is not defined
>>> user=User.query.all()
>>> user
[<User 1>, <User 2>, <User 3>, <User 4>, <User 5>, <User 6>, <User 7>]
>>> input1=User('anuja','anuja@gmail.com','anuja123')
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: __init__() takes 1 positional argument but 4 were given
>>> input1=User(username='anuja',email='anuja@gmail.com',password='anuja123',id='8')
>>> db.session.add(input1)
>>> db.session.commit()
>>> user=User.query.all()
>>> user
[<User 1>, <User 2>, <User 3>, <User 4>, <User 5>, <User 6>, <User 7>, <User 8>]
>>>

```

After running the command:



- Retrieving Data:

```

C:\Users\Admin\Desktop\pyproject>python
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> from pyproject import db
C:\Users\Admin\AppData\Local\Programs\Python\Python37-32\lib\site-packages\flask_sqlalchemy\__init__.py:834: FSADeprecat
ionWarning: SQLAlchemy_TRACK_MODIFICATIONS adds significant overhead and will be disabled by default in the future. Set
it to True or False to suppress this warning.
  'SQLALCHEMY_TRACK_MODIFICATIONS adds significant overhead and '
>>> from pyproject.models import User
>>> user=User.query.first()
>>> user
<User 1>
>>> user.password
'$2b$12$0ykSHCPeHsfQrQg.Ns.WuOztDa9U8p3YR6hMb01L1QJM0c3VZ7GWS'
>>> user.username
'18bce2246'
>>>

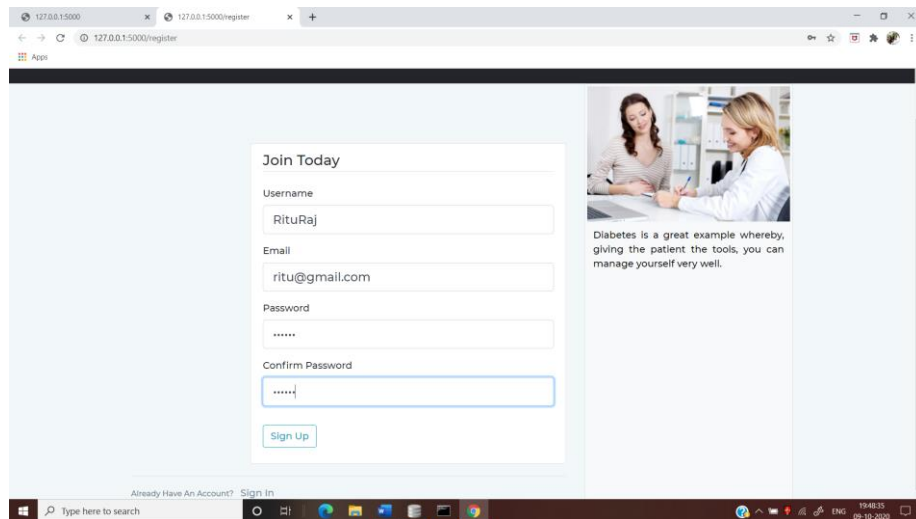
>>> user=User.query.first()
>>> user
<User 1>
>>> user.id
1
>>> user.username
'18bce2246'
>>> user.email
'ritika.raj2018@vitstudent.ac.in'
>>>

>>> user1=User.query.filter_by(username='anuja')
>>> user1
<flask_sqlalchemy.BaseQuery object at 0x0145BEF0>
>>> user1.all()
[<User 8>]

>>> data=DiabetesInput.query.filter_by(uid='2')
>>> data=data.all()
>>> data
[<DiabetesInput 2>]
>>> data=DiabetesInput.query.filter_by(Pregnancies='0')
>>> data=data.all()
>>> data
[<DiabetesInput 3>, <DiabetesInput 10>]
>>>

```

- Inputting data from the website:



Database:

DB Browser for SQLite - C:\Users\Admin\Desktop\project\project\project.db

File Edit View Tools Help

Database Structure Browse Data Edit Pragma Execute SQL

Table: user

Username	Email	Password	id
Filter	Filter	Filter	Filter
1 18bce2246	ritika.ra2018@vitstudent.ac.in	\$2b\$12\$0yKqCzEh6QpQp.Ns.WuOz...	1
2 ritika	ritika952@gmail.com	\$2b\$12\$MTOQH4n8uMBG9ndqj...	2
3 John	john@gmail.com	\$2b\$12\$Qu67BythvVogOK9fWdQ...	3
4 ri	johnson@gmail.com	\$2b\$12\$mdRnU.W24/aQh...	4
5 18bCE2246	rajilts.prasad@gmail.com	\$2b\$12\$9pOABcmj9sC14k40R.311...	5
6 rit	12@gmail.com	\$2b\$12\$Q4B4qW99gU13CT7m...	6
7 ay004	abc@gmail.com	\$2b\$12\$7S8z3K8W.jouu8vAR4.OL...	7
8 anuja	anuja@gmail.com	anuja123	8
9 RituRaj	ritu@gmail.com	\$2b\$12\$9c8rCW01ORqZLaGfR9f...	9

1 - 9 of 9

Create a new database file

Type here to search

SQL Log Plot DB Schema Remote

UTF-8

If similar username/email is given:

127.0.0.1:5000 127.0.0.1:5000/register 127.0.0.1:5000/login

127.0.0.1:5000/register

Join Today

Username

18bce2246

That username is taken. Please choose a different one.

Email

ritika.ra2018@vitstudent.ac.in

That email is taken. Please choose a different one.

Password

Confirm Password

Sign Up

Already Have An Account? Sign In

Diabetes is a great example whereby, giving the patient the tools, you can manage yourself very well.

If wrong login credentials are given:

127.0.0.1:5000 127.0.0.1:5000/register 127.0.0.1:5000/home 127.0.0.1:5000/login

127.0.0.1:5000/login

Medizin

Home Treatment Sources About

Helpline

Login Unsuccessful. Please check email and password

Log In

Email

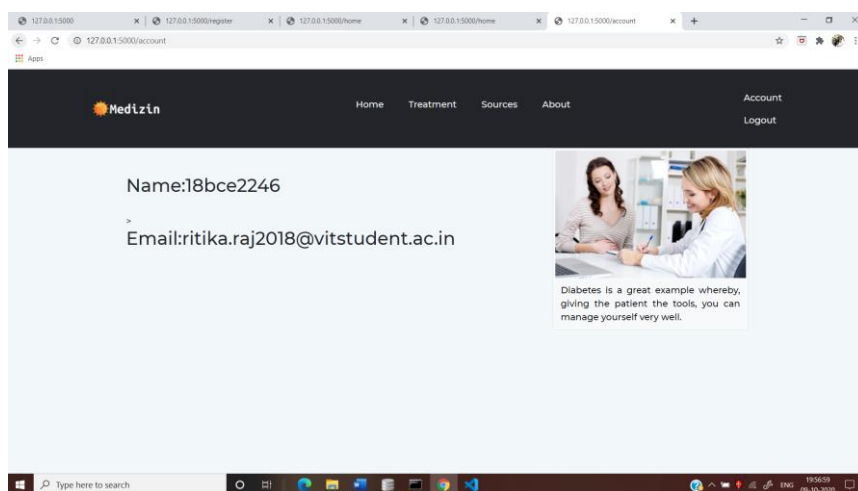
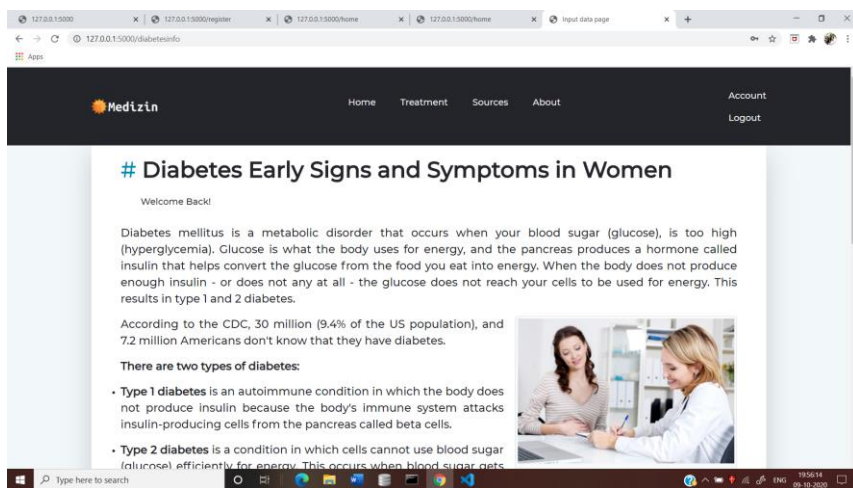
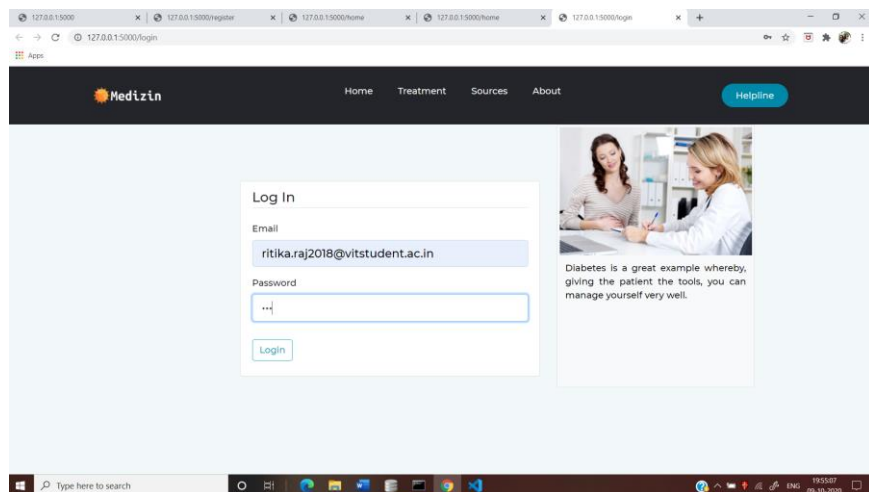
ritika.ra2018@vitstudent.ac.in

Password

Login

Diabetes is a great example whereby, giving the patient the tools, you can manage yourself very well.

After giving right credentials:



- Diabetes Input Table:

```
CREATE TABLE "DiabetesInput" (  
    "uid" INTEGER NOT NULL UNIQUE,  
    "user_id"    INTEGER NOT NULL UNIQUE,  
    "Name"      TEXT NOT NULL,  
    "Pregnancies"    INTEGER NOT NULL,  
    "BloodPressure"  INTEGER NOT NULL,  
    "Insulin"    INTEGER NOT NULL,  
    "DPF" INTEGER NOT NULL,  
    "GLUCOSE"  INTEGER NOT NULL,  
    "Skin" INTEGER NOT NULL,  
    "BMI" INTEGER NOT NULL,  
    "AGE" INTEGER NOT NULL,  
    "Cmodel"  TEXT NOT NULL,  
    PRIMARY KEY("uid")  
);
```

Flask code:

```
class DiabetesInput(db.Model):  
    uid = db.Column(db.Integer, primary_key=True)  
    user_id=db.Column(db.Integer, db.ForeignKey('user.id'), nullable=False)  
    Name =db.Column(db.String(20), nullable=False)  
    Pregnancies= db.Column(db.Integer, nullable=False)  
    BloodPressure = db.Column(db.Integer, nullable=False)  
    Insulin= db.Column(db.Integer, nullable=False)  
    DPF= db.Column(db.Integer, nullable=False)  
    GLUCOSE = db.Column(db.Integer, nullable=False)  
    Skin = db.Column(db.Integer, nullable=False)
```

```
BMI = db.Column(db.Integer, nullable=False)
AGE = db.Column(db.Integer, nullable=False)
Cmodel = db.Column(db.String(60), nullable=False)
```

Inputting from command prompt:

Command

The first screenshot shows a terminal window where a Python script is being executed. The script imports the necessary modules, creates a database session, and inserts data into the 'DiabetesInput' table. The second screenshot shows the 'DB Browser for SQLite' application displaying the 'DiabetesInput' table with 10 rows of data.

id	user_id	Name	Pregnancies	BloodPressure	Insulin	DPF	GLUCOSE	Skin	BMI	AGE	Cmodel
1	1	Ritika	1	120	120	120	120	120	120	20	k
2	2	anu	2	120	120	120	120	120	120	45	m
3	3	8 Ritika Raj	0	120	120	120	120	120	120	120	4
4	4	Anuja Prasad	2	140	120	120	120	120	120	45	2
5	5	Anuja	120	120	120	120	120	120	120	50	4
6	6	1 madhuri	1	120	120	120	120	120	120	50	2
7	8	1 E	1	120	120	2	120	120	120	45	4
8	9	1 qwert	1	120	120	4	120	120	120	45	4
9	10	10 Stacy	0	120	120	4	120	120	120	25	6

- Retrieving Data:

The screenshot shows a terminal window where a Python script is being executed. The script imports the necessary modules, creates a database session, and retrieves data from the 'DiabetesInput' table. The output shows a list of 10 'DiabetesInput' objects and the first object's name, 'Ritika'.

```
>>> d1=DiabetesInput.query.all()
>>> d1
[<DiabetesInput 1>, <DiabetesInput 2>, <DiabetesInput 3>, <DiabetesInput 4>, <DiabetesInput 5>, <DiabetesInput 6>, <DiabetesInput 7>, <DiabetesInput 8>, <DiabetesInput 9>, <DiabetesInput 10>, <DiabetesInput 11>]
>>> d1=DiabetesInput.query.first()
>>> d1.Name
'Ritika'
>>>
```



```
>>> from pyproject.models import DiabetesInput
>>> input=DiabetesInput.query.first()
>>> input
<DiabetesInput 1>
>>> input.uid
1
>>> input.name
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
AttributeError: 'DiabetesInput' object has no attribute 'name'
>>> input.Name
'Ritika'
>>> input.BloodPressure
120
>>> input.DPF
120
>>> input.AGE
30
>>>
```

```
>>> data=diabetesinput.query.filter_by(uid='2')
>>> data=data.all()
>>> data
[<DiabetesInput 2>]
>>> data=diabetesinput.query.filter_by(Pregnancies='0')
>>> data=data.all()
>>> data
[<DiabetesInput 3>, <DiabetesInput 10>]
>>>
```

- Inputting Data from the website:

The screenshot shows a web browser window with a URL of 127.0.0.1:5000/diabetesinfo. The page contains a form with the following fields and values:

- UserID:** #DIA000
- Name:** NATASHA
- Pregnancies:** 1
- Glucose:** 120
- Blood Pressure:** 120
- Skin Thickness:** 120
- Insulin:** 120
- BMI:** 120
- Diabetes Pedigree Function:** 0
- Age:** 20
- Classification Model:** Kernel SVM

A blue 'Submit' button is located at the bottom of the form.

Database:

The screenshot shows a DB Browser for SQLite window with a table named 'diabetes_input'. The table has 11 rows of data. The columns are: uid, user_id, Name, Pregnancies, BloodPressure, Insulin, DPF, GLUCOSE, Skin, BMI, AGE, and Cmodel.

uid	user_id	Name	Pregnancies	BloodPressure	Insulin	DPF	GLUCOSE	Skin	BMI	AGE	Cmodel
1	1	Ritika	1	120	120	120	120	120	120	20	k
2	2	2.ane	2	120	120	120	120	120	120	45	m
3	3	8. Ritika Raj	0	120	120	120	120	120	120	120	4
4	4	4. Anuja Prasad	2	140	120	120	120	120	120	45	2
5	5	1. Anuja	120	120	120	120	120	120	120	50	4
6	6	1. madhuri	1	120	120	120	120	120	120	50	2
7	8	1. E	1	120	120	2	120	120	120	45	4
8	9	1. qwert	1	120	120	4	120	120	120	45	4
9	10	10. Stacy	0	120	120	4	120	120	120	25	6
10	11	1. NATASHA	1	120	120	0	120	120	120	20	4

- Retrieving Data:

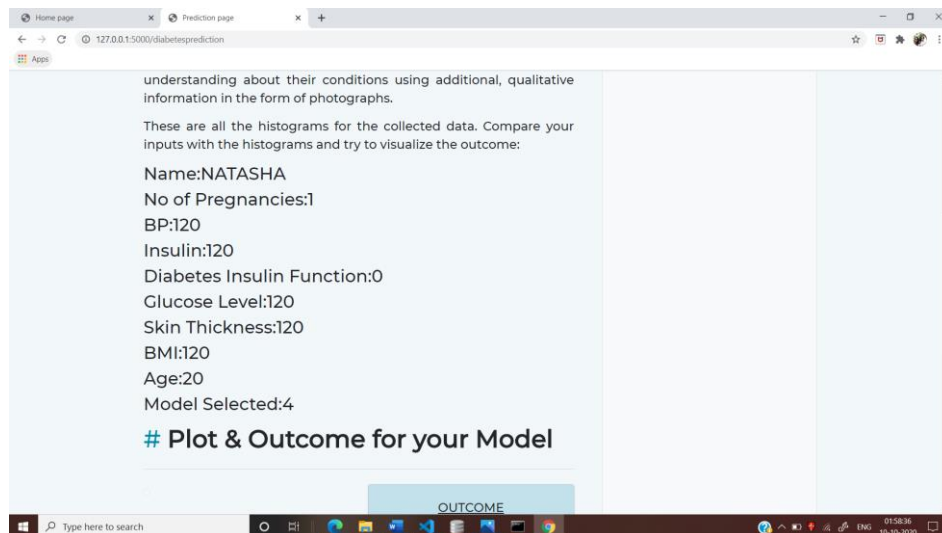
Query (To catch the input)-

```
Diabetes=db.session.query(DiabetesInput).order_by(DiabetesInput.uid.desc()).fi
rst()
return render_template('diabetesprediction.html',Diabetes=Diabetes)
```

Displaying in Prediction Page:

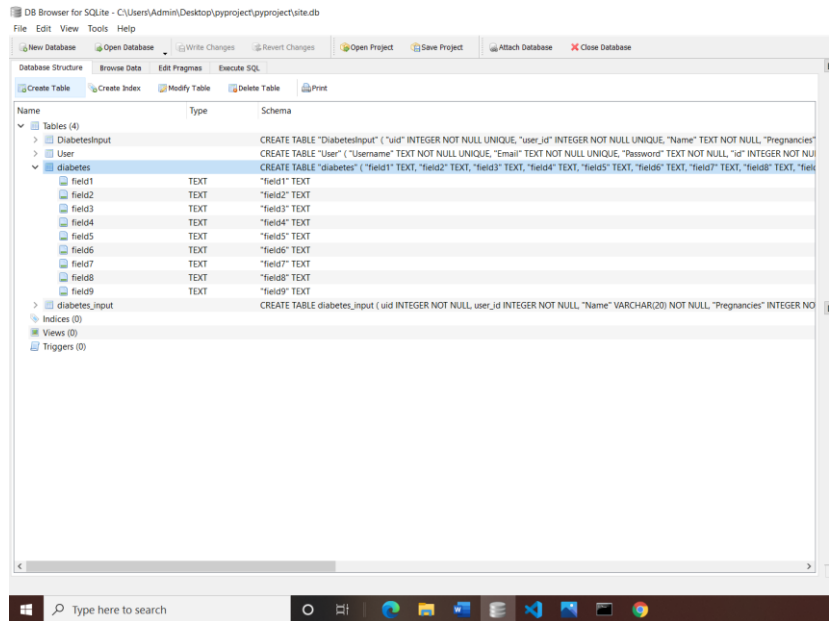
```
<h3>Name:{{ Diabetes.Name }}</h3>
<h3>No of Pregnancies:{{ Diabetes.Pregnancies}}</h3>
<h3>BP:{{ Diabetes.BloodPressure }}</h3>
<h3>Insulin:{{ Diabetes.Insulin }}</h3>
<h3>Diabetes Insulin Function:{{ Diabetes.DPF }}</h3>
<h3>Glucose Level:{{ Diabetes.GLUCOSE }}</h3>
<h3>Skin Thickness:{{ Diabetes.Skin }}</h3>
<h3>BMI:{{ Diabetes.BMI }}</h3>
<h3>Age:{{ Diabetes.AGE }}</h3>
<h3>Model Selected:{{ Diabetes.Cmodel }}</h3>
```

Output displayed:



Training Dataset (Imported from CSV file):

- For training the machine learning model:



DB Browser for SQLite - C:\Users\Admin\Desktop\pyproject\pyproject\site.db

File Edit View Tools Help

New Database Open Database Write Changes Revert Changes Open Project Save Project Attach Database Close Database

Database Structure Browse Data Edit Pragma Execute SQL

Table: diabetes

Filter in any column

	field1	field2	field3	field4	field5	field6	field7	field8	field9
	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
2	6	148	72	35	0	33.6	0.627	50	1
3	1	85	66	29	0	26.6	0.351	31	0
4	8	183	64	0	0	23.3	0.672	32	1
5	1	89	66	23	94	28.1	0.167	21	0
6	0	137	40	35	168	43.1	2.288	33	1
7	5	116	74	0	0	25.6	0.201	30	0
8	3	78	50	32	88	31	0.248	26	1
9	10	115	0	0	0	35.3	0.134	29	0
10	2	197	70	45	543	30.5	0.158	53	1
11	8	125	96	0	0	0	0.232	54	1
12	4	110	92	0	0	37.6	0.191	30	0
13	10	168	74	0	0	38	0.537	34	1
14	10	139	80	0	0	27.1	1.441	57	0
15	1	189	60	23	846	30.1	0.398	59	1
16	5	166	72	19	175	25.8	0.587	51	1
17	7	100	0	0	0	30	0.484	32	1
18	0	118	84	47	230	45.8	0.551	31	1
19	7	107	74	0	0	29.6	0.254	31	1
20	1	103	30	38	83	43.3	0.183	33	0
21	1	115	70	30	96	34.6	0.529	32	1
22	3	126	88	41	235	39.3	0.704	27	0
23	8	99	84	0	0	35.4	0.388	50	0

Go to: 1

Create a new database file