DEVELOPMENT OF STUDENT WEB PORTAL USING PYTHON PROGRAMMING

Dr. K. Valarmathi
Professor,

Ms. V. Hemapriya
Assistant Professor,

Computer Science and Engineering,
Panimalar Engineering College,
Chennai, Tamilnadu, India

Mrs. D. Anuradha
Associate Professor,

Ms. S. Sathya
Assistant Professor,

Computer Science and Engineering,
Panimalar Engineering College,
Chennai, Tamilnadu, India

Abstract— Python is used for creating web sites, but it is a tedious task for creating dynamic web page securely. To make Python program more interactive, CGI Program act as a gateway interface for implementing Python. Many methods were used to create dynamic web page in server side scripting language. Python programming takes major role to create dynamic web page more securely using   MYSQL database connectivity. Now a day's maintaining Student database is a difficult process in many organizations. Hence the proposed work reduces the difficulties of maintaining the student database using Python.
Keywords— CGI (Common Gateway Interface) , MYSQL

I.   INTRODUCTION

Python is a dynamic, interpreted scripting language. Python Source code does not declare the types of variables or parameters or methods. This feature makes the code short and flexible. Python was developed by Guido van Rossum in the late eighties and early nineties at the National Research Institute for Mathematics and Computer Science in the Netherlands. Python is derived from many other languages like C, C++ and UNIX shell and other scripting languages.

In Python programming explicit declaration of variable is not necessary .Variables can be assigned and used directly by assigning the values. Python program can be typed using different editors namely **vi editor** or **gedit** facilities. Python programs save with the extension of **.py.**
Python features
1.1   Python's feature highlights include
Limited keywords: Python has limited keywords, follows simple structure, and simple syntax.

Easy readability: Python code provides clear readability

Easy-to-maintain: Maintaining Python's code is the key secret of python programming

A huge standard library: contains cross platform compatible libraries common to UNIX, Windows and Macintosh.

Portable: Python can run on a wide variety of hardware platforms and has the same interface on all platforms.

Extendable: Low-level modules can be added to Python interpreter. This facility helps the programmers to add or customize their

GUI Programming: Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh and the X Window system of Unix.

Scalable: Python provides a better structure and support for large programs than shell scripting.

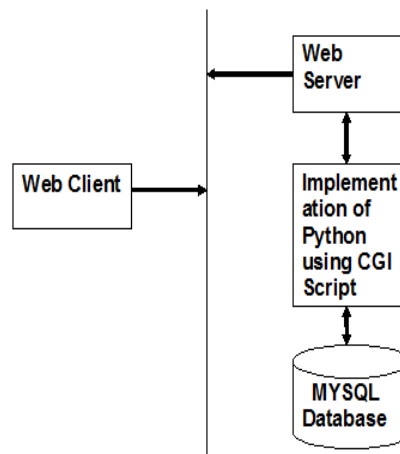*Databases:* Python provides interfaces to all major commercial databases.

Fig 1: Python Database

II.   HTTP PROTOCOL

2.1 Simple Python Code (For Printing Employee Details)

Code is implemented and tested in BOSS 5.0(Bharat Operating System Solutions) (ANOKHA 32 bit)

Click applications -> Accessories -> Terminal

vi employee.py (For typing the python program)

python employee.py (for Executing python program)
EMPLYOYEE.PY

```
# Reading Employee Details and Printing the same
class Employee:
pass
john = Employee() # Create an empty employee record
# Fill the fields of the record
john.prno = 20110001
john.name = 'John Doe'
john.dept = 'Computer Lab'
john.salary = 15000
print "------------------"
print "Employee Details"
print "-------------------"
print "Prno  = ",john.prno
print "Name  = ",john.name
print "Dept  = ",john.dept
print "Salary = ",john.salary
```

OUTPUT

```
[fosslab@bosslab]$ python employee.py
------------------
Employee Details
-------------------
Prno   =  20110001
Name   =  John Doe
Dept   =  Computer Lab
Salary =  15000
```

III. PYTHON CODING WITH MYSQL DATABASE
Create Database and Table using MYSQL

Step 1: mysql -u root -p   \\To connect with mysql
password:
Step 2: create database student;   \\ Create database
Step 3: use student;   \\Using database create a table
Step 4: create table details(regno varchar(10),
```
                name varchar(20),
                subcode1 varchar(5),
                mark1 int(3),
                subcode2 varchar(5),
                mark2 int(3),
                subcode3 varchar(5),
                mark3 int(3),
                subcode4 varchar(5),
                mark4 int(3),
                subcode5 varchar(5),
                mark5 int(3));
```

Step 5: desc details;   \\describe the table
details
Step6: insert into details
vales('2114110401','John','cs2351',80,'cs2352',85,'cs2353',90,'cs2354',100,'cs2355',96);
Step7: insert into details
vales('2114110402','Black','cs2351',70,'cs2352',75,'cs2353',80,'cs2354'

'cs2354',85,'cs2355',90);

Step8: insert into details
vales('2114110403','Patrieck','cs2351',90,'cs2352',85,'cs2353',80,'cs2354',75,'cs2355',79);

Step9: insert into details
vales('2114110406','Anu','cs2351',45,'cs2352',85,'cs2353',90,'cs2354',100,'cs2355',96);

Field	Type	Null	Key	Default	Extra
regno	varchar(10)	YES		NULL	
name	varchar(20)	YES		NULL	
subcode1	varchar(5)	YES		NULL	
mark1	int(3)	YES		NULL	
subcode2	varchar(5)	YES		NULL	
mark2	int(3)	YES		NULL	
subcode3	varchar(5)	YES		NULL	
mark3	int(3)	YES		NULL	
subcode4	varchar(5)	YES		NULL	
mark4	int(3)	YES		NULL	
subcode5	varchar(5)	YES		NULL	
mark5	int(3)	YES		NULL	

Step 10: select * from details;

regno	name	subcode1	mark1	subcode2	mark2	subcode3	mark3	subcode4	mark4	subcode5	mark5
2114110401	John	cs2351	80	cs2352	85	cs2353	90	cs2354	100	cs2355	96
2114110402	Black	cs2351	70	cs2352	75	cs2353	80	cs2354	85	cs2355	90
2114110403	Patrieck	cs2351	90	cs2352	85	cs2353	80	cs2354	75	cs2355	79
2114110406	Anu	cs2351	45	cs2352	85	cs2353	90	cs2354	100	cs2355	96

Create a Web page Using HTML:

Step 1: su password:
Step 2: cd /var/www   \\ Location of html program
Step 3: vi simple.html
```
<html>
<head>
<title> Sample</title>
</head>
<form action= "simple3.cgi" method="post">
Register number: <input type="text" name="regno">
<input type="submit" name="submit" >
</form>
<body>
</body>
</html>
```

Create a Web page using python with CGI script:

Step 1: cd /var/www   \\ Location of CGI program
Step 2: vi simple3.cgi
```
#!/usr/bin/python
import cgi
form = cgi.FieldStorage()
regno=form.getvalue('regno')
print "Content-type: text/html\n\n"
print "<h1>Student Result</h1>"
```
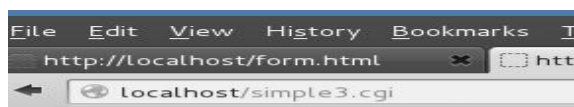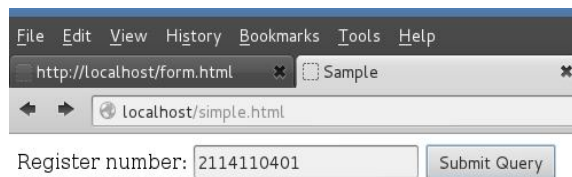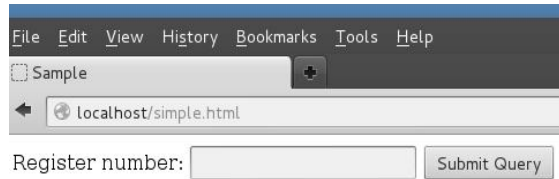
```
import MySQLdb
# Open database connection
db = MySQLdb.connect("localhost","root","root","student")
# prepare a cursor object using cursor() method
cursor = db.cursor()
cursor.execute("select
regno,name,subcode1,mark1,subcode2,mark2,subcode3,mark3
,subcode4,mark4,subcode5,mark5    from    details    where
regno='%s' "% regno)
data=cursor.fetchall()
if not data:
  print "<H1>!!!Invalid Register number</h1>"
for row in data:
  print "<table border=1>"
  print "<tr>"
  print "<td>"
  print "<b> Register number </b>"
  print "</td>"
  print "<td>"
  print row[0]
  print "</td>"
  print "</tr>"
  print "<tr>"
  print "<td>"
  print "<b> Name </b> "
  print "</td>"
  print "<td>"
  print row[1]
  print "</td>"
  print "</tr>"
  print "</table>"
  print "<br>"
  print "<br>"
  print "<br>"
  print "<table border=1>"
  print "<tr>"
  print "<th>"
  print "Subcode"
  print "</th>"
  print "<th>"
  print "Marks"
  print "</th>"
  print "<th>"
  print "Result"
  print "</th>"
  print "</tr>"
  print "<tr>"
  print "<td>"
  print row[2]
  print "</td>"
  print "<td>"
  print row[3]
  print "</td>"
  print "<td>"
if row[3]>=50:
  print "Pass"
else:
 print "Fail"
 print "</td>"
 print "</tr>"
 print "<tr>"
print "<td>"
print row[4]
print "</td>"
print "<td>"
print row[5]
print "</td>"
print "<td>"
if row[5]>=50:
print "Pass"
else:
print "Fail"
print "</td>"
print "</tr>"
print "<tr>"
print "<td>"
print row[6]
print "</td>"
print "<td>"
print row[7]
print "</td>"
print "<td>"
if row[7]>=50:
print "Pass"
else:
print "Fail"
print "</td>"
print "</tr>"
print "<tr>"
print "<td>"
print row[8]
print "</td>"
print "<td>"
print row[9]
print "</td>"
print "<td>"
if row[9]>=50:
print "Pass"
else:
print "Fail"
print "</td>"
print "</tr>"
print "<tr>"
print "<td>"
print row[10]
print "</td>"
print "<td>"
print row[11]
print "</td>"
print "<td>"
if row[11]>=50:
print "Pass"
else:
print "Fail"
print "</td>"
print "</tr>"
print "</table>"
cursor.close()
# disconnect from server
db.close()
```

Execution Steps:

Corresponding Author: Dr.K.Valarmathi, Panimalar Engineering College, Chennai, Tamilnadu, India.

Go to Application →   Internet →   Iceweasel   Web Browser

http://localhost/simple.html

OUTPUT SCREEN SHOTS

References

[1]   Introduction to Computation and Programming Using Python, Spring 2013 edition By John V. Guttag

[2]   CGI Programming On The World Wide Web Shishir Gundavaram

[3]   web technologies a computer science perspective   jeffrey c. Jackson